

11th international meeting on visualizing biological data (VIZBI 2021)

Edited by

Sean O'Donoghue, Jim Procter, Lucy Collinson, Andrew David Yates,
Lydia Gregg, Bjorn Sommer, Sameer Velankar, Robert Beiko and Yann Ponty

Published in

Frontiers in Bioinformatics



FRONTIERS EBOOK COPYRIGHT STATEMENT

The copyright in the text of individual articles in this ebook is the property of their respective authors or their respective institutions or funders. The copyright in graphics and images within each article may be subject to copyright of other parties. In both cases this is subject to a license granted to Frontiers.

The compilation of articles constituting this ebook is the property of Frontiers.

Each article within this ebook, and the ebook itself, are published under the most recent version of the Creative Commons CC-BY licence. The version current at the date of publication of this ebook is CC-BY 4.0. If the CC-BY licence is updated, the licence granted by Frontiers is automatically updated to the new version.

When exercising any right under the CC-BY licence, Frontiers must be attributed as the original publisher of the article or ebook, as applicable.

Authors have the responsibility of ensuring that any graphics or other materials which are the property of others may be included in the CC-BY licence, but this should be checked before relying on the CC-BY licence to reproduce those materials. Any copyright notices relating to those materials must be complied with.

Copyright and source acknowledgement notices may not be removed and must be displayed in any copy, derivative work or partial copy which includes the elements in question.

All copyright, and all rights therein, are protected by national and international copyright laws. The above represents a summary only. For further information please read Frontiers' Conditions for Website Use and Copyright Statement, and the applicable CC-BY licence.

ISSN 1664-8714
ISBN 978-2-83250-637-0
DOI 10.3389/978-2-83250-637-0

About Frontiers

Frontiers is more than just an open access publisher of scholarly articles: it is a pioneering approach to the world of academia, radically improving the way scholarly research is managed. The grand vision of Frontiers is a world where all people have an equal opportunity to seek, share and generate knowledge. Frontiers provides immediate and permanent online open access to all its publications, but this alone is not enough to realize our grand goals.

Frontiers journal series

The Frontiers journal series is a multi-tier and interdisciplinary set of open-access, online journals, promising a paradigm shift from the current review, selection and dissemination processes in academic publishing. All Frontiers journals are driven by researchers for researchers; therefore, they constitute a service to the scholarly community. At the same time, the *Frontiers journal series* operates on a revolutionary invention, the tiered publishing system, initially addressing specific communities of scholars, and gradually climbing up to broader public understanding, thus serving the interests of the lay society, too.

Dedication to quality

Each Frontiers article is a landmark of the highest quality, thanks to genuinely collaborative interactions between authors and review editors, who include some of the world's best academicians. Research must be certified by peers before entering a stream of knowledge that may eventually reach the public - and shape society; therefore, Frontiers only applies the most rigorous and unbiased reviews. Frontiers revolutionizes research publishing by freely delivering the most outstanding research, evaluated with no bias from both the academic and social point of view. By applying the most advanced information technologies, Frontiers is catapulting scholarly publishing into a new generation.

What are Frontiers Research Topics?

Frontiers Research Topics are very popular trademarks of the *Frontiers journals series*: they are collections of at least ten articles, all centered on a particular subject. With their unique mix of varied contributions from Original Research to Review Articles, Frontiers Research Topics unify the most influential researchers, the latest key findings and historical advances in a hot research area.

Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers editorial office: frontiersin.org/about/contact

11th international meeting on visualizing biological data (VIZBI 2021)

Topic editors

Sean O'Donoghue — Garvan Institute of Medical Research, Australia

Jim Procter — University of Dundee, United Kingdom

Lucy Collinson — Francis Crick Institute, United Kingdom

Andrew David Yates — European Bioinformatics Institute (EMBL-EBI), United Kingdom

Lydia Gregg — Johns Hopkins University, United States

Bjorn Sommer — Royal College of Art, United Kingdom

Sameer Velankar — European Bioinformatics Institute (EMBL-EBI), United Kingdom

Robert Beiko — Dalhousie University, Canada

Yann Ponty — École Polytechnique, France

Citation

O'Donoghue, S., Procter, J., Collinson, L., Yates, A. D., Gregg, L., Sommer, B., Velankar, S., Beiko, R., Ponty, Y., eds. (2022). *11th international meeting on visualizing biological data (VIZBI 2021)*. Lausanne: Frontiers Media SA. doi: 10.3389/978-2-83250-637-0

Table of contents

- 05 **Exploring the Microbiome Analysis and Visualization Landscape**
Jannes Peeters, Olivier Thas, Ziv Shkedy, Leyla Kodalci, Connie Musisi, Olajumoke Evangelina Owokotomo, Aleksandra Dyczko, Ibrahim Hamad, Jaco Vangronsveld, Markus Kleinewietfeld, Sofie Thijs and Jan Aerts
- 20 **Visualizing Phytochemical-Protein Interaction Networks: *Momordica charantia* and Cancer**
Yumi L. Briones, Alexander T. Young, Fabian M. Dayrit, Armando Jerome De Jesus and Nina Rosario L. Rojas
- 31 **Extending Association Rule Mining to Microbiome Pattern Analysis: Tools and Guidelines to Support Real Applications**
Agostinetto Giulia, Sandionigi Anna, Bruno Antonia, Pescini Dario and Casiraghi Maurizio
- 45 **Interactive, Visual Simulation of a Spatio-Temporal Model of Gas Exchange in the Human Alveolus**
Kerstin Schmid, Andreas Knote, Alexander Mück, Keram Pfeiffer, Sebastian von Mammen and Sabine C. Fischer
- 59 **Naview: A d3.js Based JavaScript Library for Drawing and Annotating Voltage-Gated Sodium Channels Membrane Diagrams**
Marcelo Querino Lima Afonso, Néli José da Fonseca Júnior, Thainá Godinho Miranda and Lucas Bleicher
- 67 **Development and Application of Automatized Routines for Optical Analysis of Synaptic Activity Evoked by Chemical and Electrical Stimulation**
Debarpan Guhathakurta, Enes Yağız Akdaş, Anna Fejtová and Eva-Maria Weiss
- 77 **Uncertainty Visualization: Concepts, Methods, and Applications in Biological Data Visualization**
Daniel Weiskopf
- 94 **DJExpress: An Integrated Application for Differential Splicing Analysis and Visualization**
Lina Marcela Gallego-Paez and Jan Mauer
- 116 **Strategies for the Production of Molecular Animations**
Erik Werner
- 126 **SingleCANalyzer: Interactive Analysis of Single Cell RNA-Seq Data on the Cloud**
Carlos Prieto, David Barrios and Angela Villaverde

- 134 **BioViz *Connect*: Web Application Linking CyVerse Cloud Resources to Genomic Visualization in the Integrated Genome Browser**
Karthik Raveendran, Nowlan H. Freese, Chaitanya Kintali, Srishti Tiwari, Pawan Bole, Chester Dias and Ann E. Loraine
- 147 **ShapoGraphy: A User-Friendly Web Application for Creating Bespoke and Intuitive Visualisation of Biomedical Data**
Muhammed Khawatmi, Yoann Steux, Saddam Zourob and Heba Z. Sailem



Exploring the Microbiome Analysis and Visualization Landscape

Jannes Peeters^{1*}, Olivier Thas¹, Ziv Shkedy¹, Leyla Kodalcı¹, Connie Musisi¹, Olajumoke Evangelina Owokotomo¹, Aleksandra Dyczko^{2,3}, Ibrahim Hamad^{2,3}, Jaco Vangronsveld^{4,5}, Markus Kleinewietfeld^{2,3}, Sofie Thijs⁴ and Jan Aerts¹

¹CENSTAT, Data Science Institute (DSI), Hasselt University, Diepenbeek, Belgium, ²VIB Laboratory of Translational Immunomodulation, VIB Center for Inflammation Research (IRC), Hasselt University, Diepenbeek, Belgium, ³Department of Immunology and Infection, Biomedical Research Institute (BIOMED), Hasselt University, Diepenbeek, Belgium, ⁴Center for Environmental Sciences, Environmental Biology, Hasselt University, Diepenbeek, Belgium, ⁵Department of Plant Physiology and Biophysics, Faculty of Biology and Biotechnology, Maria Curie-Skłodowska University, Lublin, Poland

OPEN ACCESS

Edited by:

Robert Beiko,
Dalhousie University, Canada

Reviewed by:

Florian Ganglberger,
Center for Virtual Reality and
Visualization Research GmbH, Austria
Inimany Toby,
University of Dallas, United States

*Correspondence:

Jannes Peeters
jannes.peeters@uhasselt.be

Specialty section:

This article was submitted to
Data Visualization,
a section of the journal
Frontiers in Bioinformatics

Received: 12 September 2021

Accepted: 29 October 2021

Published: 02 December 2021

Citation:

Peeters J, Thas O, Shkedy Z, Kodalcı L, Musisi C, Owokotomo OE, Dyczko A, Hamad I, Vangronsveld J, Kleinewietfeld M, Thijs S and Aerts J (2021) Exploring the Microbiome Analysis and Visualization Landscape. *Front. Bioinform.* 1:774631. doi: 10.3389/fbinf.2021.774631

Research on the microbiome has boomed recently, which resulted in a wide range of tools, packages, and algorithms to analyze microbiome data. Here we investigate and map currently existing tools that can be used to perform visual analysis on the microbiome, and associate the including methods, visual representations and data features to the research objectives currently of interest in microbiome research. The analysis is based on a combination of a literature review and workshops including a group of domain experts. Both the reviewing process and workshops are based on domain characterization methods to facilitate communication and collaboration between researchers from different disciplines. We identify several research questions related to microbiomes, and describe how different analysis methods and visualizations help in tackling them.

Keywords: microbiome, visual analytics, data visualization, bioinformatics, data analysis, biostatistics

1 INTRODUCTION

The human gut microbiome has been the topic of many academical studies over the latest years, as several diseases like multiple sclerosis and inflammatory bowel disease, have been found to be connected to it (Wilck et al., 2017; Allaband et al., 2019). Studies even suggest that there is a link between the gut microbiome and depression (Dash et al., 2015; Winter et al., 2018). Tripathi et al. (2018) noted that although much progress has been made in this research field, a framework of aggregated scientific knowledge about the topic (one needs to pose meaningful hypotheses) is still lacking. The authors therefore advocate for more discovery-driven, and tool-driven research projects instead of traditional, hypothesis-driven studies conducted using hypotheses-driven statistical or mathematical models. The reasoning behind this inductive approach, from which we start with a hypothesis-free exploration of the data, is that it can lead to unanticipated interesting questions as well as deeper insights of understanding. A promising and by now well-established technique to support hypothesis-free data exploration, are interactive data visualization and Visual Analytics (VA) (Van Wijk, 2005; Keim et al., 2010). Visualization experts play an important role in this as they possess the knowledge and visual literacy to perform visual analysis, and develop meaningful interactive data visualizations. Data visualization projects, and the interplay between visualization experts and domain experts therefore becomes more prominent in different research fields; e.g., social sciences (Lamqaddam et al., 2020), archaeology (Panagiotidou et al., 2020), and microbiome research. To work closely with domain experts, and performing a good requirement analysis is key for the visualization experts to succeed in the development of meaningful visualization tools (Knoll

et al., 2020). This involves the visualization expert(s) to gain sufficient background knowledge in the research domain to understand expert's needs, and domain experts to express their domain tasks, data types and analysis (Sakai and Aerts, 2015).

In this paper, we provide a picture of how (interactive) data visualization and visual analytics are currently used in microbiome research. To do so, literature covering visual analysis pipelines, visualization methods and visual analytic tools designed for microbiome research were reviewed and discussed in interactive expert panel focus groups. These interactive workshops were organized based on the principles of Kerzner et al. (2019) and Gray et al. (2010), using an informal setting in which discussion was facilitated through brainstorming games (e.g., Post-up, Card sort).

2 MATERIALS AND METHODS

Data and material for the analysis was collected using a combination of literature review and collaborative workshops with a panel of experts related to microbiome research.

2.1 Literature Review

Literature was hand collected based on a google scholar search on “microbiome visualization,” “microbiome visual analysis,” and “microbiome studies interactive analysis.” To be as inclusive as possible, additional tools were added if referenced in one of the papers within this selection. Nevertheless, the final collection may not be exclusive. In total, 31 papers published between 2009 and 2021 were selected. This should give an accurate presentation of the analysis tools landscape. Note, that because of the special interest in the visual analytics aspect, a strong emphasis on visualization tools was laid in the search and collection process.

The review process was done manually. From each paper we extracted general information on the tool; such as the platform the tool is hosted on, the input formats of the data, and the aspects of the microbiome that could be revealed using the tool (e.g., diversity indices, differential relative abundances, etc.). In addition, we described which methods were used to extract information on the several microbiome aspects as well as the visualization method (if not overlapping) used for visual interpretation. Note that for the interest of this study, only analyses to perform on operational taxonomic unit (OTU) or amplicon sequence variant (ASV) tables were taken into account. This paper will not cover the process of transforming raw sequence data (.fastq files) into readable OTU/ASV tables.

2.2 Evaluation Methods

To analyze and draw conclusions of the observations, two techniques coming from the business environments were used to facilitate insight generation by revealing underlying patterns; being a *closed card sorting* game (Sakai and Aerts, 2015) and the use of a *history map* (Gray et al., 2010). Both were conducted individually prior to the expert panel focus group discussions.

In *card sorting*, the objective is domain characterization, which is crucial in visual design. As visualization experts might not have

sufficient background knowledge in the field of microbiome research, “*expert's need*” have to be extracted in more abstract low-level tasks (Munzner, 2014). In this card sorting game, these abstractions were made based on the literature. The rules of the game are simple, a set of cards need to be sorted into meaningful categories. Cards can represent items, objects, pictures, names or attributes. In this case a closed Card Sort was conducted, meaning a set of predetermined categories is used; each category representing a feature (aspect) of the microbiome that could be identified in the analysis tools. The cards to be sorted contained the statistical methods, visualization algorithms and visual designs that were found in the same analysis tools to compute and represent these aspects. The sort in this exercise was based on the frequency of occurrence in literature (i.e., if PCoA was used to visualize between sample diversity, the “PCoA” card was assigned to the “between sample diversity” class). An example of how this was done can be found in **Supplementary Figure S1** in the supplementary materials.

The *history map* (Gray et al., 2010) is used to familiarize new people with an organization's culture and history during periods of rapid growth. The idea is to ask employees share memories about certain topics (e.g., company successes, changes in leadership, culture shifts, etc.) on a continuous timeline, to later summarize and reflect on the findings, and look for emergent patterns. The same exercise can be done in academics however, shifting the focus from an “organisation's history” to a particular research field or research topic; being “microbiome research through visual analysis.” In the interest of this study, development of microbiome research through visual analysis was broken down in three separate questions: 1) How did the interest (coverage) of microbiome aspects develop over time in the collection of reviewed analysis tools?, 2) How did the methods used to capture these microbiome aspects develop or change over time?, 3) How did the use of platforms to host these visual analysis tools change over time? Like in the Card Sort game, the answers to these questions were provided based on frequency of occurrence in the literature (i.e., if a certain tool offers Shannon diversity to capture within sample diversity, it is listed on the timeline of methods used to capture within sample or alpha diversity). Hence, multiple timelines were created; one containing the aspect coverage, one representing the used platforms, and one for each aspect individually to show the methodological development and visual representations over time. An example of such an exercise can be found in the **Supplementary Figure S2**.

2.3 Workshops

To further explore and dive deeper into the results captured by the individual literature review analysis, similar exercises were done within a focus group of domain experts related to the microbiome. As experts in a complex research field may sometimes experience difficulties expressing their research objectives and needs due to the inherently exploratory nature of the analysis, data and its uncertainties, literature suggests the use of domain characterization exercises to facilitate communication and information sharing within interdisciplinary groups of experts (Munzner, 2009; Panagiotidou et al., 2020). The expert groups were drawn



FIGURE 1 | Phase two and three of the workshops; **(A)** a post up brainstorm sessions in which participants were asked to provide their knowledge on 5 microbiome analysis related questions, and **(B)** a closed card sorting to provide their experts opinion on currently used methods. The actual results of the post up session can be found in supplementary material (**Supplementary Tables S1, S5**).

from three different research domains (biologists, statisticians, and visualization experts), to obtain diverge insights coming from different perspectives. In total, 2 workshops were organized. The first workshop included 4 participants, among which 1 microbiologist, 2 bio-statisticians and 1 visualization expert. The second workshop included 1 microbiologist, 3 bio-statisticians and 1 visualization expert. The same visualization expert was present in both meetings, whereas all other participants within the focus group changed. Due to COVID-19, the second workshop had to be done virtually using the online collaborative whiteboard platform Miro (miro.com). The first meeting could be done in person. The meetings took between 1 h and 30 min and 2 h, using an informal “game” structured setting. An informal setting was chosen to create an open and friendly environment to establish collegiality and trust across participants (Knoll et al., 2020). The workshops were conducted in three phases; 1) introduction, 2) Post-Up, and 3) Card Sorting.

At the start of the workshop, goals and guidelines for the participants were communicated, followed by a short introduction round and warm up exercise. According to Kerzner et al. (2019), the latter encourages idea generation and self expression and consequently advances in agency.

The second phase of the workshop aimed at generating ideas. During this phase a *post-up* game (Gray et al., 2010) was played to support brainstorming. The idea of this game is to start with a question on which the group of participants will search answers to. The question should be written down somewhere (e.g., on a whiteboard) such that participants can consult it at any time. The brainstorm is done individually, and answers should be written down on separate sticky notes. Answers can then be shared and sorted underneath the question and briefly presented toward the group after a set amount of time; being 2 min within our setting. The intend of this game was to compare the experts’ knowledge and needs to what is currently available in the microbiome visualization tools. In this set-up, five questions were asked:

- Q1: Conceptually, what information/knowledge can we gain or would we like to obtain from doing microbiome research? For example: influence of food on obesity, how drugs change the gut microbiome, etc.
- Q2: Which data is required or relevant to obtain this knowledge? For example: location, time, etc.?
- Q3: To answer questions of Q1: which specific aspects can be retrieved from the OTU/ASV abundance table? e.g., taxonomic abundance, most present taxonomies in collected samples.
- Q4: Given the aspects you wrote down before, can you think about methods needed and or used (statistically, visually) to obtain this information.
- Q5: When you think about your own research, I’m interested in the platforms, tools, packages you have used, or are using currently to analyze the microbiome. Can you list these up?

An image of the workshop environment at the end of this phase is shown in **Figure 1A**, and the list of provided answers can be found in the supplementary materials (**Supplementary Tables S1, S5**).

Phase three of the workshop included the same closed card sort game as performed in the individual reviewing process. The same cards and categories were provided to the expert panel and the objective of the game was the same, only this time sorting was based on experts’ knowledge rather than frequency of occurrence in literature; allowing to easily identify discrepancies between experts opinions and literature. Therefore only one card was provided for each statistical method, visualization algorithm or visual design this time, regardless frequency of use. Still, participants were free to duplicate cards. All categories were briefly explained before the start of the game. Each card also contained concise description of the method. Based on this information, participants were asked to sort the card under the categories they believed it could be used for. Furthermore,

participants were also allowed to create additional cards and categories containing methods and aspects not covered in the tools. At the end, participants were asked to conduct a value mapping through dot voting (Gray et al., 2010) on the cards that had been sorted. Statistical methods, visualization algorithms and visual designs that experts believed were still informative and insightful obtained a dot, providing an indication of the ones that are still accurate and useful in microbiome research, which could result in interesting discussions. An image of the workshop environment at the end of this exercise is presented in **Figure 1B**.

Important with these type of exercises is to promote open communication among participants to obtain as much context and background knowledge as possible, and acknowledge expertise from all participants to gain as much input as possible (Kerzner et al., 2019). The workshops were recorded for later reference during analysis with permission of the participants.

3 RESULTS

3.1 Research Objectives

Based on the literature and the answers to Q1 of the post up game (i.e., Conceptually, what information/knowledge can we gain or would we like to obtain from doing microbiome research?), several objectives were identified in which microbiome research can play a role. The responses of the experts on the question “what information or knowledge can or could be obtained from microbiome research?” could be categorized in 5 major objectives. The first, and most prominent research objective listed by the experts is the association between the microbiome and diseases, among which obesity and multiple sclerosis. All experts believed there is a role to play for the microbiome in disease treatment. Currently, drugs are used for disease treatment, but more research is required on whether they directly affect the disease or whether the effect is mediated through the gut microbiome. If the latter is true, drug alternatives such as a specific diet or fecal therapy could play a prominent role. The second topic of interest that came forward during the discussions was the effect of environmental and personal conditions on microbiome composition. These include seasonal changes (e.g., sunlight), geographical location, past diseases, diet, etc. The third topic listed during the discussions was the role for the microbiome in agriculture, specifically its effect on plant growth/production. Next, psychological associations were listed as a topic of interest. Literature has shown that a link between the gut microbiome and psychological diseases (e.g., depression) exists (Dash et al., 2015; Winter et al., 2018), but does the gut microbiome composition also alter our mood? Lastly, the experts expressed interest in the role of the microbiome in areas such as crime investigation. This could be in revealing social contact patterns based on similar microbiome compositions, using the skin microbiome to see who had physical contact with whom, but also with certain objects or animals, etc. A commonality between all the topics listed above is that they all rely on finding the association between the microbiome (s) and other parameters,

and more interestingly (if possible) in revealing causal relationships.

3.2 Data Requirements

Qualitative data is needed to provide accurate answers to these research objectives. Based on the answers and discussion on Q2 of the post up game (i.e., Which data is required or relevant to obtain this knowledge?), a general outline of “qualitative data collection in microbiome research” could be established. Besides the need of qualitative genome sequencing, samples should be accompanied by a set of metadata containing additional information about the host and its environment, the (clinical) study, and the sample collection. Specifically, baseline characteristics of the host should be captured (e.g., if human: age, gender, geographic location, etc.); environment information from the host (e.g., exposure to certain chemicals, passive smoker, diet, etc.); clinical information from both the host and the clinical trial study; and information about sample collection (e.g., timestamp, sample location within the host). Furthermore, to obtain metabolic information, accurate databases are required for functional profiling. A full list of the answers provided to Q2 can be found in the supplementary material (**Supplementary Table S2**).

3.3 Methods and Algorithms in Microbiome Research

To analyse this data and investigate previously listed research objectives, an interplay between statistical methods, algorithmic visualizations and (interactive) visual representations are required. These allow us to reveal certain aspects of the microbiome which accordingly permit us to provide answers to these research objectives.

3.3.1 A Changing Research Landscape

The rapid development of these methods and algorithms in microbiome research is clearly visible in the literature. The first visualization oriented microbiome analysis tools only covered the visualization of taxonomic abundance and relationships (Ondov et al., 2011), and the exploration of within- and between-sample diversity (Schloss et al., 2009). Not many years later, tools started to implement methods to test for statistical differences between samples in terms of abundance (differential abundance analysis), and statistical differences between cohorts or populations that can be related to a particular (disease) condition (biomarker discovery) (McMurdie and Holmes, 2013; Robertson et al., 2013; Weiss et al., 2017). During the same period, the first tools allowing for visual exploration of microbial interactions and associations became available as well (Kuntal et al., 2013), used to get an idea about which microbes tend to co-occur with each other. Meta data also became more important in the analysis of diversity between microbiome samples. It is more and more explored together with the on taxonomic abundance based diversity scores (Vázquez-Baeza et al., 2013; Zakrzewski et al., 2017; Liao et al., 2019). In the latest years, major developments occurred; enrichment analysis found its way into the microbiome visual



FIGURE 2 | A matrix overview of the tools and algorithms included in the literature review, in which the tools and algorithms are represented in the columns, and the microbiome aspects they measure and present listed as rows. Cells indicate the coverage of an aspect by the corresponding tool, and are colored based on the platform they were hosted on.

analysis tools (Kuntal et al., 2016; Chong et al., 2020), researchers are now able to visualize and investigate taxon-function relationships (McNally et al., 2018), and tools were developed for longitudinal studies including feature volatility and time series analysis (Baksi et al., 2018; Bokulich et al., 2018). The latest development in the field was the introduction of machine learning (ML) classifiers (Chong et al., 2020; Shamsaddini et al., 2020). Regardless of the fast development and progression in microbiome research and its visual analysis tools, all types of analyses and aspects of the microbiome have remained relevant for exploration. This observation was made based on the fact that older methods (e.g., diversity indices) are still implemented in newer published tools (Carpenter et al., 2021), and confirmed by the expert panel focus group discussions. **Figure 2** provides an overview of which microbiome aspects are currently covered by which tool.

3.3.2 Aspects

In Q3 of the post up game, we asked our participants to list all aspects that could be extracted from an OTU/ASV abundance table in order to answer the research questions provided on Q1. A wide variety of features were provided and could be categorized into 4 major research interests: 1) exploratory analysis of baseline characteristics such as (relative) abundance, variability, diversity and richness, 2) statistical effect modelling to obtain effect sizes and p-values, and identify differences taxa abundance and discover biomarkers, 3) interaction models to reveal the interrelationship between taxa, and 4) functional analysis of taxa. In the following we discuss the aspects that were found

to be extracted in literature, supplemented with important findings that came up during the workshops (answers to Q4 and card sort) and review process.

(Relative) Abundance

Perhaps the most important thing in microbiome research is the ability to look into the (relative) abundance of taxa within and across samples. It provides a first impression of which taxa (functions) are most prominent within a sample, group or population, and can guide us into certain directions of interests. Due to the compositional structure of the data in microbiome research, one tends to prefer looking into relative abundances rather than absolute abundances. An exploration of the (relative) abundances involves no complex statistical modelling, and can be easily done by means of some descriptive statistics and a visual representation of the data.

Visualization—Stacked or regular bar-charts seem to be the most prevalent visual encodings to do so, although they are limited in the number of species (functions) they can visualize for the chart to still be readable (Knaflitz, 2015). Heatmaps are a frequently used alternative that allow us to visualize all species (functions) at once. The use of color intensity as a channel in heatmaps on the other hand makes the comparison in terms of relative abundance a bit harder than using length (bars) (Munzner, 2014). Nonetheless, does the use of color allows us to easily include (relative) abundance visualization in other microbiome aspect oriented visualizations [e.g., alongside taxonomic classification (Ondov et al., 2011)]. Other alternative visual encodings found in literature include the use

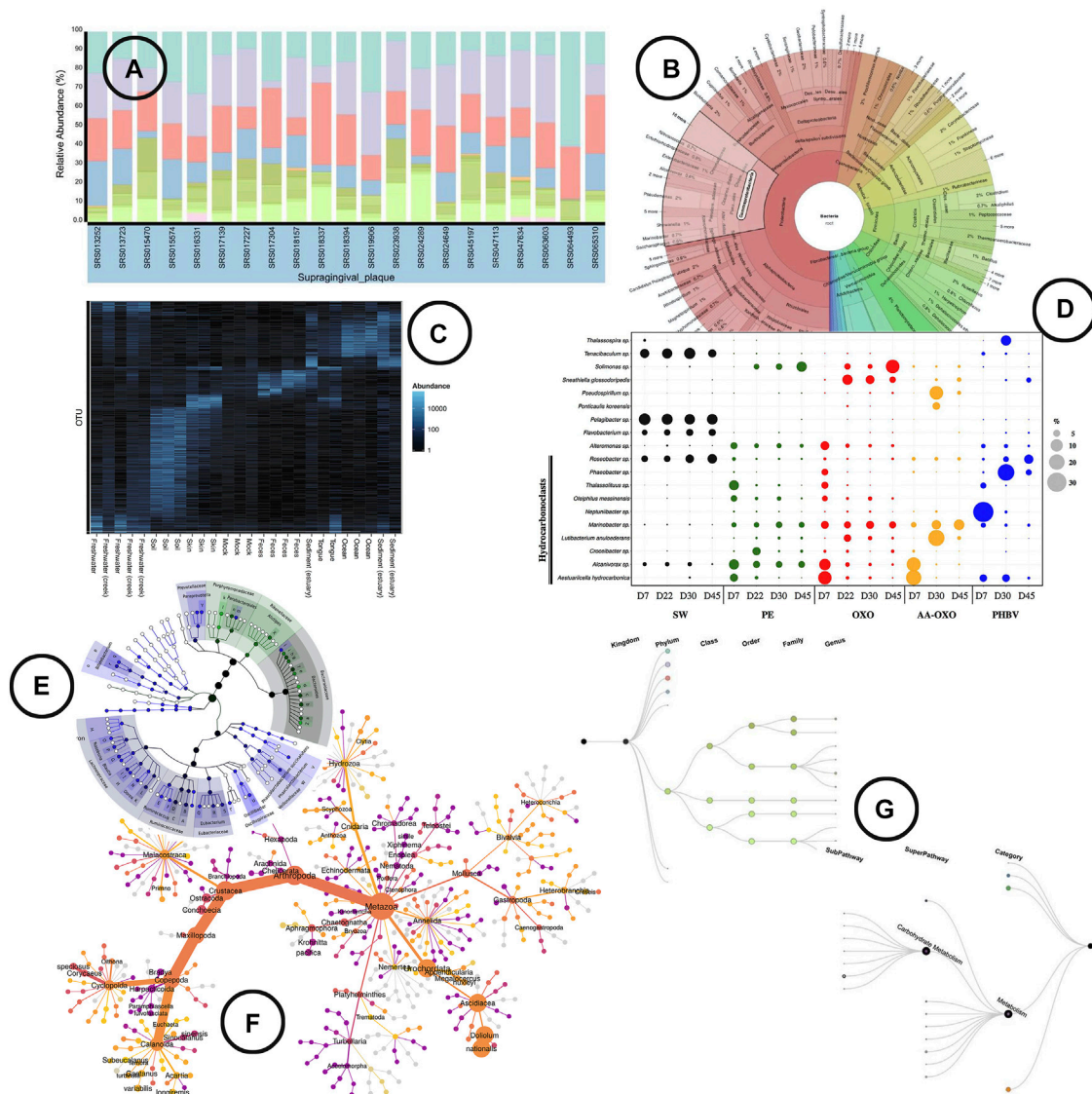


FIGURE 3 | An overview of the visual encodings used to display (relative) abundance and hierarchical/relational structures; **(A)** relative abundance displayed by means of a stacked bar chart in BURRITO (McNally et al., 2018), **(B)** a krona sunburst chart showing the taxonomic hierarchy of the observed bacteria and their relative abundance (Ondov et al., 2011), **(C)** OTU abundance visualized as a heatmap using Phyloseq (McMurdie and Holmes, 2013), **(D)** relative abundance of OTUs represented in a bubble plot (Dussud et al., 2018), **(E)** GraPhlAn, a tree based visualization tool that allows to add visual annotations (Asnicar et al., 2015), **(F)** a “heat tree” visualization showing the taxonomic hierarchy within its tree structure and OTU abundance using node width (Foster et al., 2017), **(G)** taxa and function hierarchy displayed within tree structures in BURRITO with node width representing abundance (McNally et al., 2018).

of angle [e.g., sunburst chart (Ondov et al., 2011)] and area [e.g., bubble plot (Dussud et al., 2018)] to display (relative) abundance. An overview of how visualization is been used to represent (relative) abundance in literature is shown in **Figure 3**.

Hierarchical/Relational Structures

Microbiome analysis can be done up to different levels depending on the interest of the study, and the sequencing process used to sample the data. In general, sequencing up to a deeper level provides more detailed information. On the other hand, does it bring more problems into the analysis due to sparseness. Most

statistical models are not suited to handle many zero counts in the data (Knight et al., 2018).

Visualization—In the analysis of microbiome samples, it can be interesting to visually represent the hierarchical level of the taxonomies (domain, kingdom, phylum, class, order, family, genus, species), hierarchical level of the functions (category e.g., metabolism, superpathway e.g., carbohydrate metabolism, subpathway e.g., glycolysis), or even the phylogenetic relationship of the species. Tree structures (including radial trees, cladograms, etc.) are the typical visual encodings used, and are basically the only visual encoding found in literature (**Figures 3B,E–G**).

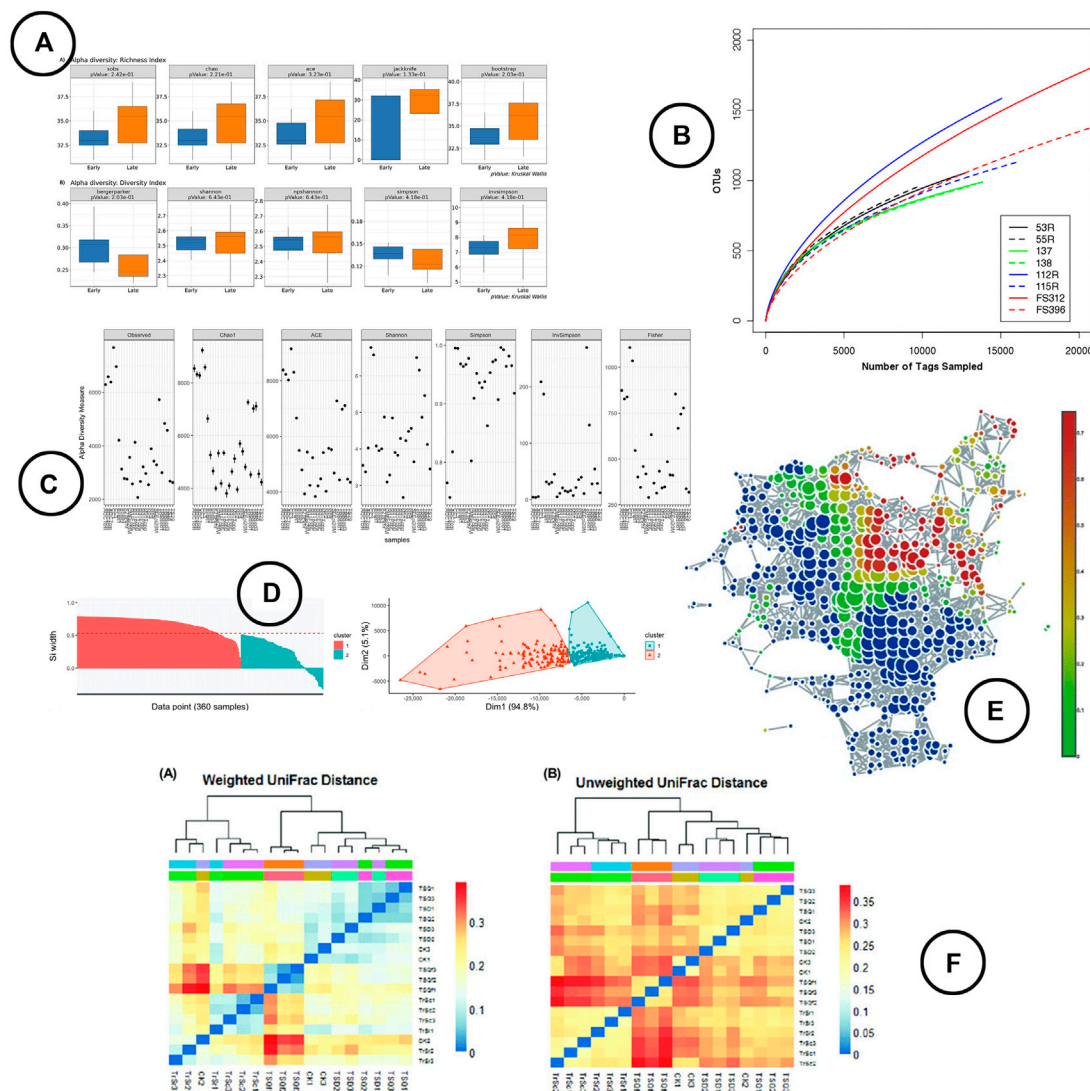


FIGURE 4 | An overview of the visual encodings used to display within (alpha) and between (beta) diversity; **(A)** alpha diversity metrics compared between groups by means of box-plots in BiomMiner (Shamsaddini et al., 2020), **(B)** rarefaction curve showing the number of OTUs by sequence size in Mothur (Schloss et al., 2009), **(C)** alpha diversity metrics visualized using scatter plots in Phyloseq (McMurdie and Holmes, 2013), **(D)** beta diversity visualized using ordination in IMAP (Buza et al., 2019), **(E)** a node-link diagram produced using TDA in TMAP to display beta diversity (Liao et al., 2019), **(F)** heatmap visualizations showing beta diversity distance matrices (Lei et al., 2017).

Within Sample (Alpha) Diversity

Alpha diversity provides an idea of the diversity of species within a particular sample. This metric is often used as a biomarker (Prehn-Kristensen et al., 2018) in disease association studies, but also as a check of sample quality (Schloss et al., 2009).

Analysis—Looking into alpha diversity calculations and visual representations, no clear evolution could be found. Many different options exist and are used, but no uniform standard has emerged yet. Typically, alpha diversity metrics can be distinguished into two types: richness- and evenness-measures; *Chao1* being the most used richness metric, and *Shannon* the most used evenness metric. A full list of alpha diversity measures is provided by Hagerty et al. (2020). The authors advocate for the

use of a composite metric based on exploratory factor analysis (EFA), taking into account both richness and evenness metrics unified in one.

Visualization—box-plots are widely used to display alpha diversity if the objective is to make a comparison between sample cohorts. Line-charts (rarefaction curves) and scatter-plots tend to be used more frequently when visualizing the metrics across samples; the rarefaction curve presenting the (predicted) sample richness by sequence size, often used for re-sampling. Venn diagrams are used to display which part of the microbial taxa are present in multiple samples in relation to the total diversity within those samples. An overview of the visuals used to represent the within sample diversity is given in **Figure 4A–C**.

Between Sample (Beta) Diversity

Beta diversity represents the diversity of species across samples, commonly used to find clusters of similar samples. Typically, this feature is calculated in the exploratory analysis, as it provides a first impression on which taxa are important to distinguish samples, but also on how microbial compositions are related to environmental and personal meta data. With regard to the research objectives listed above, social contact networks could for instance be revealed based on similar microbiome compositions of the skin.

Analysis—Beta diversity is expressed as a distance matrix calculation on relative OTU abundance, which serves as an input for visual exploration of sample divergence and similarity. Often occurring distance metrics are: (*weighted*) *UniFrac*, *Jaccard*, *Bray-Curtis* and *Jenson-Shannon* (Oliveira et al., 2018; Chong et al., 2020; Shamsaddini et al., 2020). An important note however is that none of these measures account for the compositionality of the data. Compositional replacements for these distance metrics have been developed; *phlir* (Silverman et al., 2017) as a replacement for (*weighted*) *UniFrac*, and *Aitchison distance* (Aitchison et al., 2000) for *Jensen-Shannon* divergence and the *Bray-Curtis* dissimilarity metrics. Nevertheless, implementation is lacking in the microbiome visual analysis tools.

From 2019 onward, a new trend seemed to develop, which is to test for statistical significance of the between-sample differences (ordination measures). Statistical tests used for this include AMOVA, HOMOVA, ANOSIM, PERMANOVA, PERMDISP, and LIBSSHUFF (Buza et al., 2019; Chong et al., 2020; Shamsaddini et al., 2020). One important recent development is that ordination analysis techniques can be performed on sample functional potentials rather than their taxonomic proportions (Nagpal et al., 2019).

Visualization—The visual representation of beta diversity can be either directly through heatmaps of the distance matrix (Lei et al., 2017), through ordination based methods (e.g., PCoA, NMDS) which present the samples in a 2 or 3 dimensional space using dimensionality reduction techniques (Vázquez-Baeza et al., 2013; Wang et al., 2016; Bolyen et al., 2019), or by means of network visualizations based on topological data analysis (TDA) (Liao et al., 2019) or cut-off based edges (McMurdie and Holmes, 2013). Note that because of the compositional ignorance in the commonly used distance metrics, samples will be almost exclusively discriminated based on the features that are most abundant relative to the others features and not on the most variable ones between samples. Therefore, sample location could vary a lot in ordination plots when different features are included or excluded (Gloor et al., 2017). An example of the visual encodings listed above is shown in **Figure 4D–F**.

Differential Abundance

With differential abundance analysis, OTUs that differ significantly between samples, cohorts or populations are identified using statistical hypothesis testing. In doing so, taxa can be related to a certain response (e.g., disease state, growth process).

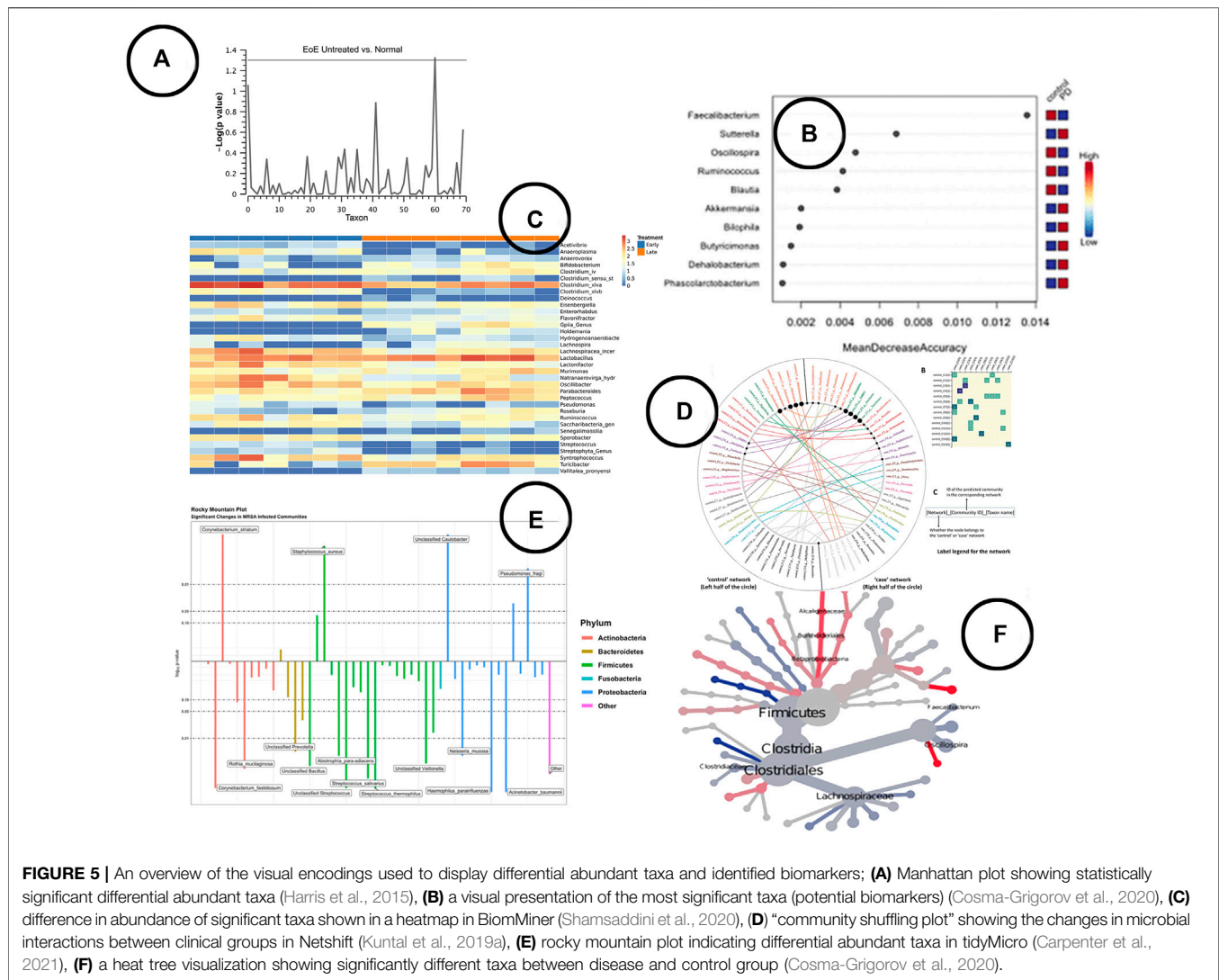
Analysis—The search for the ideal analysis method for differential abundance is still ongoing (Hawinkel et al., 2019). To date, it has been proven that distributional assumptions do not hold for the majority of the taxa, leading to poor performance of parametric models (Hawinkel et al., 2020). The problem with non parametric rank alternatives such as Wilcoxon is that they are typically less powerful in comparison to parametric tests due to their vulnerability to ties in the data (Jonsson et al., 2016). Custom methods have been developed to test on significant differences between microbiome data, taking the compositionality of the data into account (e.g., ANCOM, ALDEx2) (Gloor et al., 2017). In comparison to the complete lack of awareness in Beta diversity analyses, differential relative abundance analysis methods relying on these compositional assumptions are present in some visual analysis tools (Zakrzewski et al., 2017). Yet, another possible solution lies in semiparametric models, such as Probabilistic Index Models (PIM) (Thas et al., 2012). These are based on rank tests (non parametric), but allow for estimates of effect sizes and inclusion of continuous covariates. So far, they haven't been introduced in microbiome visual analysis tools in a significant way. An important note that came up during one of the workshops, is that the methods used in visual analysis tools are all limited to cross sectional analysis. To the awareness of the expert panel, methods that do allow differential abundance testing in longitudinal studies are sparse, and mostly parametric. Besides, with the currently offered methods, conclusions can only be drawn about associations between taxa and meta data identifying sample cohorts, whereas inference on causality would be of major interest. In recent years, several methods have been proposed relying on structural equation models to reveal the direct and mediation effect of the microbiome on a certain response (Sohn and Li, 2019; Wang et al., 2020). These however cannot be found in the current visual analysis tools. Nonetheless, these methods suffer from validity issues (Vanderweele and Vansteelandt, 2009).

Visualization—To visualize statistical significance, several visual encodings have been used; ranging from simple heatmaps and box-plots, to more complex visuals like the Manhattan plot (Harris et al., 2015), rocky mountain plot (Carpenter et al., 2021), volcano plot (Shamsaddini et al., 2020) or heat tree (Foster et al., 2017). An overview of some of the visualizations found in literature is given in **Figures 5A,C,E,F**.

3.3.2.5 Biomarker Discovery

Biomarker discovery focuses on finding specific parameters or indicators, called biomarkers, that can be related (assigned) to a particular condition (disease).

Analysis—When it comes to biomarker discovery, two schools of thought can be distinguished: one using predictive models such as machine learning classifiers, and the other based on hypothesis testing. Among the predictive models, LEfSE (Swenson and Swenson, 2014) is by far the most offered method in the visual analysis tools, followed by some other machine learning algorithms. Methods based on hypothesis testing include methods for statistical difference testing between groups (both parametric and non-parametric). Similar to differential abundance testing, models for clinical studies that take into



account the effect of an intervention on both the response (immune response) and biomarkers can be of interest as well. The primary difference however is that their focus is merely on association rather than causal relationships. To the best of our knowledge, there are only two tools that test for association between biomarkers (microbiome taxa compositions) and clinical response variables: NetShift using an algorithmic visualization (Kuntal et al., 2019a), and PhyloSeq using supervised methods (i.e., canonical correspondence analysis, discriminant correspondence analysis, sparse linear discriminant analysis, etc.) (McMurdie and Holmes, 2013). The authors of IVikodak listed the quantification of association between specific sets of bacteria with disease state as a planned future enhancement (Nagpal et al., 2019). None of them however allow for longitudinal analysis, taking into account the effect of an intervention on both the biomarkers and disease response.

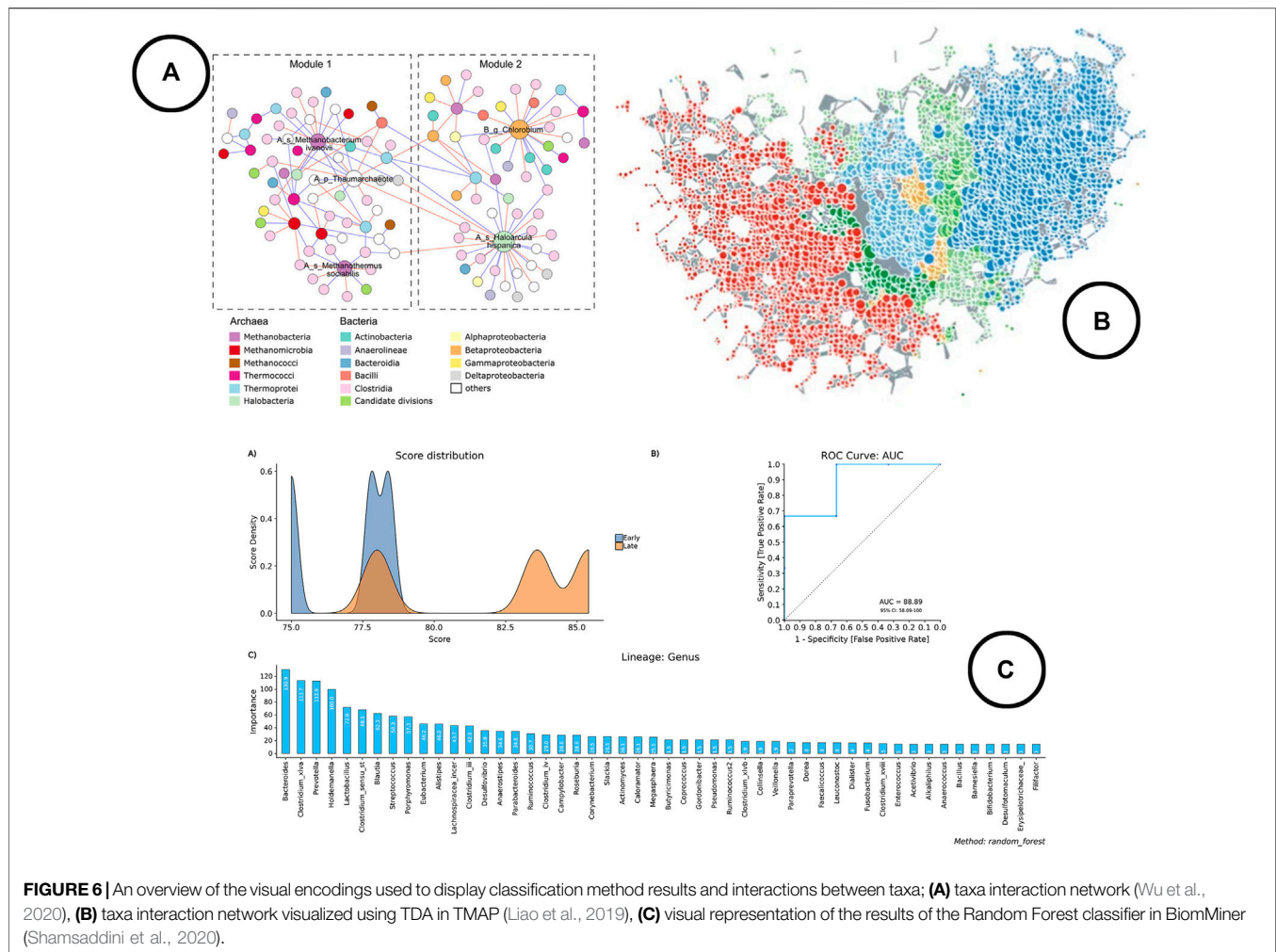
Visualization—A wide variety of visual encodings have been used to represent the result of biomarker discovery analysis;

ranging from simple heatmaps and bar charts, to more complex visuals like the volcano plot (Shamsaddini et al., 2020) and heat trees (Foster et al., 2017). An ongoing search noted by one of the experts in the focus group discussions is on how to visually represent the results of clinical longitudinal intervention studies: how do microbial composition and clinical response variables change over time given a particular intervention. In **Figures 5B,D,F**, some of the visualizations used in the visual analysis tools are shown.

Classification

Classification is used to classify samples in predefined groups based on their microbial composition. It provides information on the most important features (taxa) within sample cohorts, and is therefore often returning as a method for biomarker identification as well.

Analysis—Classification methods are fairly new in microbiome research, as only the more recently developed visual analysis tools cover these methods (Chong et al., 2020;



Shamsaddini et al., 2020). Machine learning algorithms such as random forest classifiers or support vector machines are typically used for this type of analysis.

Visualization—Line charts (expressed as ROC curves) are typically used to represent model performance, whereas bar charts are used to display the most important features. An example of how this is shown in literature is given in **Figure 6C**.

Microbial Interaction

The analysis of microbial interaction is focused on identifying the relationship between species. Different types of relations can exist between microbes: mutualistic, commensal, parasitic and competitive (Faust et al., 2012). The goal is to find a method that reveals all of them at once. Identifying these relationships is important for all research objectives listed above. It provides more context on why certain taxa abundances differ in certain situations, and guides us towards possible causal relationships (e.g., is the drug altering the relative OTU abundance or is it altering its relative abundance through another taxa that contains a specific relationship with the OTU of interest).

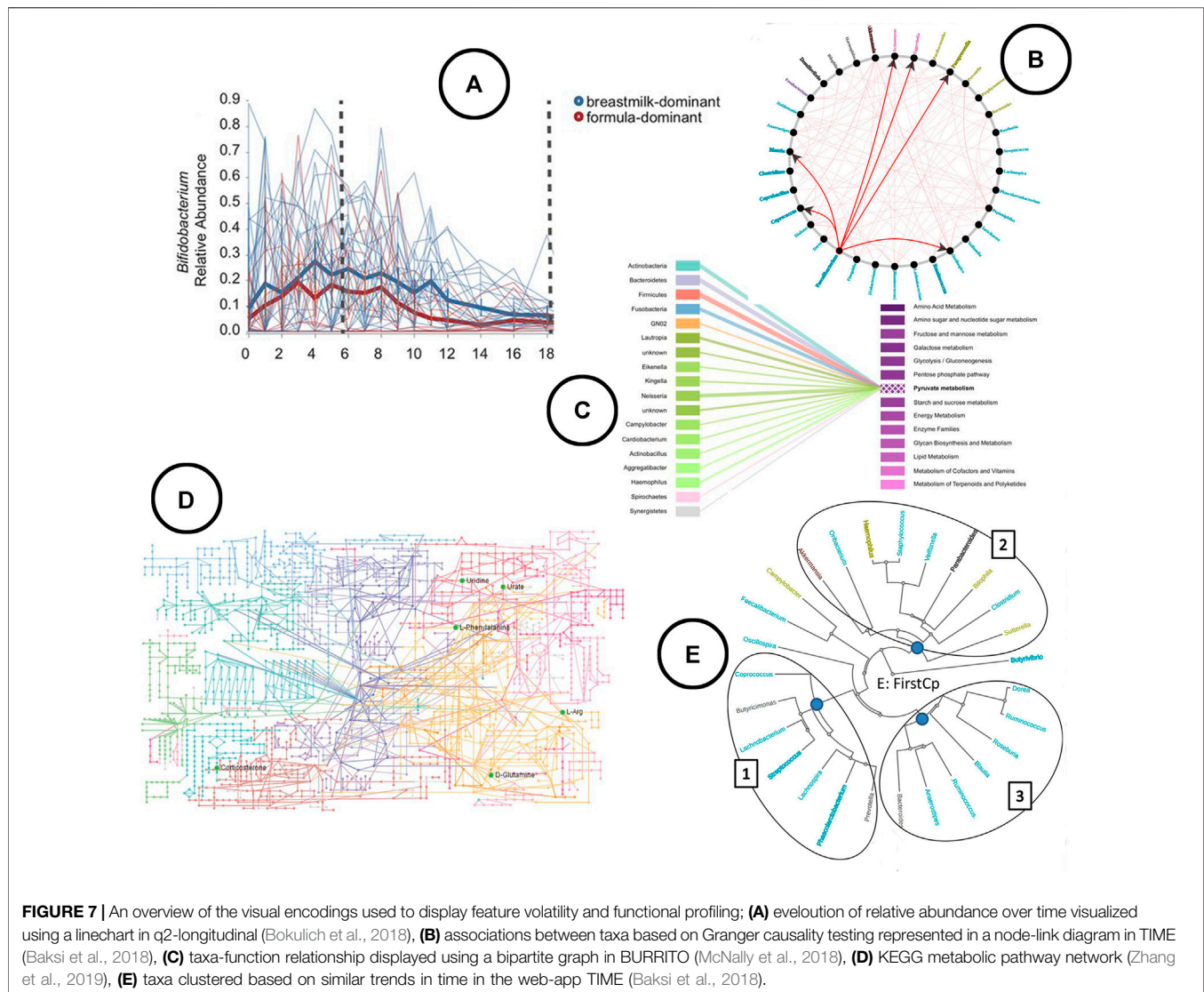
Analysis and Visualization—Looking at the development of microbial interaction analysis within the microbiome visual

analysis tools, new methods have been introduced during recent years, which gives an indication that the use of different methods is still further explored. At the moment, three schools of thought can be distinguished: 1) correlation based methods. Problem however with correlation is that it doesn't correct for the compositionality of the data, and thus leads to spurious correlations (Gloor et al., 2017). Therefore, methods like *SparCC*, *SPIEC-EASI* and *FastSpar* were developed which result in network visualizations based on cut-off values (Chong et al., 2020). 2) Predator-Prey based methods using (generalized) Lotka Volterra equations to model relationships (Shaw et al., 2016; Kuntal et al., 2019b). 3) Topology based methods using topological data analysis (TDA) to construct the networks (Liao et al., 2019). All of these methods result in a graph visualized as a node-link diagram. **Figures 6A,B** provides an overview of how networks are used to represent microbial interactions.

Functional Profiling

As mentioned above on (*relative*) abundance, one could also look into the metabolic functions of microbial populations.

Analysis—Depending on the type of sequencing, different programs and methods can be used for functional profiling.



Galloway-Peña and Hanson (2020) provide a nice overview including use cases and shortcomings. Using 16S rRNA sequencing, methods such as *PICRUSt* (Langille et al., 2013) and *Tax4Fun* (Aßhauer et al., 2015) allow to predict the gene content potential functionality based on a comparison between relative abundances and the reference genome of the taxa present. An important note of the authors that came up in the expert panel discussions as well is that these however are rough approximations, as they don't take into account actual protein expressions. Using shotgun and metatranscriptome sequencing approaches, tools such as *MetaGeneMark* (Zhu et al., 2010) and *Glimmer-MG* (Kelley et al., 2012) carry out protein sequence homology based searches against databases of orthologues, enzymes, or protein domains and families for gene identification and annotation. The results could then be used for pathway enrichment analysis.

Visualization—The link between taxa and functions can be visualized using bipartite graphs (Figure 7C) or interactive stacked bar charts using highlighting, as was done in *Burrito* (McNally et al.,

2018). The result of functional profiling are typically represented in a metabolic pathway network (Figure 7D) (Zhang et al., 2019).

Longitudinal Analysis

As mentioned before in the section on differential abundance and repeated in the section on biomarker discovery, to gain a deeper understanding of causal relationships between the microbiome and various sample cohorts (e.g., grouped by disease state), longitudinal studies are required (Secrier and Schneider, 2013). Given the literature reviewed in this study, two tools were found to allow for longitudinal microbiome time series analysis; *TIME* (Baksi et al., 2018), and *q2-longitudinal* (Bokulich et al., 2018), which is an extension on *QIIME2*.

Analysis—In *q2-longitudinal*, linear mixed effect models are used to test for differential abundance. Changes of microbial sample compositions are captured across time using unweighted UniFrac, whereas in *TIME* dynamic time warping distance is used to capture groups of taxa showing similar trends over time. *TIME* identifies

causal relationships among taxa using Granger Lasso causality. Stationary taxonomic groups (meaning no inter-microbial competition) are identified using an augmented dickey fuller test.

Visualization—Both tools allow for exploration of feature volatility using volatility plots (line charts) (**Figure 7A**). causal relationships between taxa are displayed using node-link diagrams (**Figure 7B**); clustering of taxa showing similar trends over time is visualized using a radial tree structure (**Figure 7E**).

Still, to the best of our knowledge no methods for longitudinal mediation analysis allowing for the identification of causal relationships between intervention, microbiome and response are incorporated yet.

3.4 Tools and Platforms

Situating all publications on a timeline (see **Figure 2**) it becomes clear that initially (2009–2014) tools were mainly made available as standalone downloadable software. Quickly, tools were made available as web applications as well. R and Python are often used to run the analyses on the server side of these web applications (Chong et al., 2020; Reeder et al., 2020), but packages and libraries do also exist to run analyses in the R studio or python programming environments (McMurdie and Holmes, 2013; Buza et al., 2019). The main reason to develop software or web-apps is to remove the constraint of coding, as not all biologist know how to code and learning R or Python might be a bit cumbersome (Huse et al., 2014; Chong et al., 2020). Hence they most often serve as complete analysis pipelines in which microbiome researchers upload their data and can perform different analyses through a point-and-click user interface (Huse et al., 2014). The major problem however with these applications is maintenance. Since standalone software is not open source, updates most often stop when funding stops, as there is nobody who can keep everything up to date besides the developers. A solution to partly alleviate this could be the use of R and Python based server apps like R Shiny (Chang et al., 2015), as was done in Microbiome Explorer (Reeder et al., 2020) or Microbiome Analyst (Chong et al., 2020). Looking into the R packages and Python libraries, three types of packages and libraries can be distinguished: the complete analysis pipeline packages which allow for a thorough and diverse analysis of the microbiome [e.g., Phyloseq (McMurdie and Holmes, 2013), MicrobiomeExplorer (Reeder et al., 2020), IMAP (Buza et al., 2019)], the extensions on these complete packages [e.g. phylogeo (Charlop-Powers and Brady, 2015)], and the computational- or visualization algorithms [e.g. SPIECE-EASI (Kurtz et al., 2015), TMAP (Liao et al., 2019)]. These extensions and algorithms both focus on revealing one particular aspect of the microbiome. During the expert panel group workshops, it became clear that R is primarily used among the participating bio-statisticians. For the creation of a custom visualization, visualization experts make use of web based environments and its according coding languages (HTML, CSS, and JS), and dedicated visualization libraries [D3 (Bostock et al., 2011), p5, etc.].

4 DISCUSSION

Based on the expert panel focus group workshops, the main interest in microbiome research is in the identification of associations

between the microbiome and host characteristics; be it environmental or health related factors within or among humans, or growth indicators in agriculture. Relevant analysis methods are mainly differential abundance analysis and biomarker discovery. Although these analyses often include metrics like alpha diversity as model parameters, or start from preliminary exploration of the data by looking at the taxonomic compositions and diversity between groups. These methods often include baseline characteristics (e.g., diversity metrics) as model parameters, and proceed from preliminary exploratory analysis of the data.

When it comes to revealing these aspects in the data, several approaches are available. For some aspects the same approach is used exclusively, whereas for others different schools of thought apply. Within sample (alpha) diversity is captured using either richness- or evenness-measures, but a uniform standard is missing (Hagerty et al., 2020). Between sample (beta) diversity is always measured using a distance metric on relative OTU abundance, and stored in a distance matrix. None of the currently implemented distance metrics however accounts for the compositional structure of the data. This compositionality is also one of the major problems for the reliability of statistical hypothesis testing models, which are central in differential abundance testing. Based on the card sorting within the focus group discussions, it became clear that biomarker discovery can rely either on statistical hypothesis testing or predictive modeling. Therefore, many of the methods used in differential abundance testing are found to be used for biomarker discovery as well. Consequently, the same overlap can be found in methods based on predictive modeling which are used for sample classification. A major interest expressed by the expert panel group is the ability to perform causal analysis, which is currently insufficiently developed in differential abundance analysis and biomarker discovery. To do so, the necessity of longitudinal studies and analysis was stressed.

A wide variety of visual encodings exists to represent the data aspects concealed in the OTU abundance tables. Some of these are more unconventional than others, but standard charts (e.g., bar chart, line chart) are most common. Some of them are unconditionally bound to a certain data aspect; hierarchical structures within the data (e.g., taxonomic level) are visualized exclusively using tree structures, connected components are typically used to express relationships (e.g., between taxa, or between functions and taxa), and line charts are most conventional to display evolution over time. Other data aspects on the contrary have been visually represented in many different ways. (Relative) abundance has been visually encoded using channels such as length (e.g., bar chart), color saturation (e.g., heatmap), angle (e.g., Krona), and area (e.g., bubble plot). Based on visualization theory, length would be the most effective channel to display quantitative information such as (relative) abundance (Munzner, 2014), but the use of bar charts however limits the amount of information that can be displayed for it to be still informative. Color saturation on the other hand would be the least effective channel from the ones listed, whereas heatmaps would be the only choice to visually represent the entire data on a static manner. For this reason, heatmaps are also used to visualize beta diversity. It provides a nice overview of the (dis) similarities between samples, although it can become a bit

cumbersome to read when the amount of samples is too large. Since the interest is often not limited to the discovery of (dis)similar samples but also in revealing the underlying patterns between samples, ordination based methods are most prevalent in literature. They allow additional data features to be included in the visualization for interpretation, which is not possible using standard heatmaps. The downside of ordination based methods however is that these are limited to a visual representation in a 2 or 3 dimensional space, which might not capture the entire variance to be explained. By displaying the samples using TDA (i.e., node-link diagram), distance between samples is expressed in the edges between the nodes (samples), and therefore no longer relies on the geometric space (Lum et al., 2013). The visualization of the outcomes of statistical models could be as simple as using bar charts and box plots, but have been conducted many times by means of custom visuals as well. In general, the choice depends on the information of interest. If the interest is a list of potential biomarkers (i.e., most important features), a simple bar chart will do and is highly effective according to visualization theory (Munzner, 2014). If the interest is on the effect sizes or any other parameters, more complex and custom visuals are needed.

Here, it is important to also address the issue of visual literacy. In general, the advantage that comes with using standard charts is that everyone can read them. The amount and richness of information that can be shared with them is however limited. On the other hand, custom representations can provide more information in a single graphic but can become hard to read. They should be used with care, by providing the right amount of context needed by the user to understand. An example that emerged during one of the workshops was the Rocky Mountain Plot (Figure 5E) used in tidyMicro (Carpenter et al., 2021) to highlight taxa counts correlated with subjects' age. One could draw conclusions based on the highly correlated taxa counts, but important additional information is missing to draw more accurate conclusions (e.g., variability). Hence, the custom visualization can provide the solution to bring more context to the data analysts, as multiple data aspects can be embedded in the same visual and no longer need to be looked at in isolation [e.g., GraPhlAn (Asnicar et al., 2015)]. In creating these custom visuals, it is imperative that a user-driven design process is used in which visualization expert and domain expert work closely together (Munzner, 2009). Yet, current papers on microbiome visualization and visual analysis mention nothing about the use of design process.

5 LIMITATIONS

It is sometimes hard to make a clear distinction between tools, as some of them are actually algorithms (e.g., SPIEC-EASI) or visual

encodings (e.g., Krona, GraPhlAn) that act and were specifically developed as microbiome visualization tools, but are also embedded as encodings in other tools.

Given the contact constraints added through the COVID-19 pandemic, one of the workshops had to be done virtually. As not all participants were familiar with the tools used during this session, additional time was required to familiarize. Nevertheless, both meetings provided a clear overview of some important research topics to cover in microbiome research. The workshop setting was found to be key in structuring discussions, from which interesting information could be obtained such as pointing out current problems and shortcomings. Due to the interdisciplinary composition of the workshops, an additional result was that participants could quickly familiarize themselves in other research domains. We understand that providing examples during the workshops could prime answers into a certain direction. However, due to the interdisciplinary setting of the workshops, we also believe that providing an example helps participants to come to a common understanding of the question asked.

AUTHOR CONTRIBUTIONS

The literature review was conducted by JP, under the supervision of JA. All other authors listed participated in the discussions and provided their intellectual input and feedback on the literature review.

FUNDING

This work is funded through Hasselt University BOF grant ADMIRE (BOF21GP17) and BOF grant (BOF20OWB33) and Flemish Government programme “Onderzoeksprogramma Artificiële Intelligentie (AI).” MK was supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (640116) and by a SALK-grant from the government of Flanders, Belgium and by the Research Foundation Flanders (FWO), Belgium (G0G1216N, G080121N). ST and JV are supported by the UHasselt Methusalem project 08M03VGRJ.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fbinf.2021.774631/full#supplementary-material>

REFERENCES

- Aitchison, J., Barceló-Vidal, C., Martín-Fernández, J. A., and Pawłowsky-Glahn, V. (2000). Logratio Analysis and Compositional Distance. *Math. Geology*. 32, 271–275. doi:10.1023/A:1007529726302
- Allaband, C., McDonald, D., Vázquez-Baeza, Y., Minich, J. J., Tripathi, A., Brenner, D. A., et al. (2019). Microbiome 101: Studying, Analyzing, and Interpreting Gut Microbiome Data for Clinicians. *Clin. Gastroenterol. Hepatol.* 17, 218–230. doi:10.1016/j.cgh.2018.09.017
- Asnicar, F., Weingart, G., Tickle, T. L., Huttenhower, C., and Segata, N. (2015). Compact Graphical Representation of Phylogenetic Data and Metadata with GraPhlAn. *PeerJ* 3, e1029–17. doi:10.7717/peerj.1029
- Aßhauer, K. P., Wemheuer, B., Daniel, R., and Meinicke, P. (2015). Tax4fun: Predicting Functional Profiles from Metagenomic 16s Rrna Data. *Bioinformatics* 31, 2882–2884. doi:10.1093/bioinformatics/btv287

- Baksi, K. D., Kuntal, B. K., and Mande, S. S. (2018). 'TIME': A Web Application for Obtaining Insights into Microbial Ecology Using Longitudinal Microbiome Data. *Front. Microbiol.* 9, 36–13. doi:10.3389/fmicb.2018.00036
- Bokulich, N. A., Dillon, M. R., Zhang, Y., Rideout, J. R., Bolyen, E., Li, H., et al. (2018). q2-longitudinal: Longitudinal and Paired-Sample Analyses of Microbiome Data. *mSystems* 3, 1–9. doi:10.1128/msystems.00219-18
- Bolyen, E., Rideout, J. R., Dillon, M. R., Bokulich, N. A., Abnet, C. C., Al-Ghalith, G. A., et al. (2019). Reproducible, Interactive, Scalable and Extensible Microbiome Data Science Using QIIME 2. *Nat. Biotechnol.* 37, 852–857. doi:10.1038/s41587-019-0209-9
- Bostock, M., Ogievetsky, V., and Heer, J. (2011). D³: Data-Driven Documents. *IEEE Trans. Vis. Comput. Graph.* 17, 2301–2309. doi:10.1109/TVCG.2011.185
- Buza, T. M., Tonui, T., Stomeo, F., Tiambo, C., Katani, R., Schilling, M., et al. (2019). IMAP: An Integrated Bioinformatics and Visualization Pipeline for Microbiome Data Analysis. *BMC Bioinformatics* 20, 1–18. doi:10.1186/s12859-019-2965-4
- Carpenter, C. M., Frank, D. N., Williamson, K., Arbet, J., Wagner, B. D., Kechris, K., et al. (2021). tidyMicro: a Pipeline for Microbiome Data Analysis and Visualization Using the Tidyverse in R. *BMC Bioinformatics* 22, 41–13. doi:10.1186/s12859-021-03967-2
- Chang, W., Cheng, J., Allaire, J., Xie, Y., and McPherson, J. (2015). *Package 'shiny'*. CRAN. Available at: <https://cran.microsoft.com/snapshot/2015-07-29/web/packages/shiny/shiny.pdf>.
- Charlop-Powers, Z., and Brady, S. F. (2015). Phylogeo: An R Package for Geographic Analysis and Visualization of Microbiome Data. *Bioinformatics* 31, 2909–2911. doi:10.1093/bioinformatics/btv269
- Cosma-Grigorov, A., Meixner, H., Mrochen, A., Wirtz, S., Winkler, J., and Marxreiter, F. (2020). Changes in Gastrointestinal Microbiome Composition in PD: A Pivotal Role of Covariates. *Front. Neurol.* 11, 1–13. doi:10.3389/fneur.2020.01041
- Dash, S., Clarke, G., Berk, M., and Jacka, F. N. (2015). The Gut Microbiome and Diet in Psychiatry: Focus on Depression. *Curr. Opin. Psychiatry* 28, 1–6. doi:10.1097/YCO.0000000000000117
- Dussud, C., Hudec, C., George, M., Fabre, P., Higgs, P., Bruzaud, S., et al. (2018). Colonization of Non-biodegradable and Biodegradable Plastics by marine Microorganisms. *Front. Microbiol.* 9, 1–13. doi:10.3389/fmicb.2018.01571
- Faust, K., Sathirapongsasuti, J. F., Izard, J., Segata, N., Gevers, D., Raes, J., et al. (2012). Microbial Co-occurrence Relationships in the Human Microbiome. *Plos Comput. Biol.* 8, e1002606. doi:10.1371/journal.pcbi.1002606
- Foster, Z. S., Sharpton, T. J., and Grünwald, N. J. (2017). Metacoder: An R Package for Visualization and Manipulation of Community Taxonomic Diversity Data. *Plos Comput. Biol.* 13, e1005404–15. doi:10.1371/journal.pcbi.1005404
- Galloway-Peña, J., and Hanson, B. (2020). Tools for Analysis of the Microbiome. *Dig. Dis. Sci.* 65, 674–685. doi:10.1007/s10620-020-06091-y
- Gloor, G. B., Macklaim, J. M., Pawlowsky-Glahn, V., and Egozcue, J. J. (2017). Microbiome Datasets Are Compositional: and This Is Not Optional. *Front. Microbiol.* 8, 2224. doi:10.3389/fmicb.2017.02224
- Gray, D., Brown, S., and Macanufo, J. (2010). *Gamestorming: A Playbook for Innovators, Rulebreakers, and Changemakers*. Sebastopol, CA, USA: O'Reilly Media, Inc.
- Hagerty, S. L., Hutchison, K. E., Lowry, C. A., and Bryan, A. D. (2020). An Empirically Derived Method for Measuring Human Gut Microbiome Alpha Diversity: Demonstrated Utility in Predicting Health-Related Outcomes Among a Human Clinical Sample. *PLoS ONE* 15, e0229204–21. doi:10.1371/journal.pone.0229204
- Harris, J. K., Fang, R., Wagner, B. D., Choe, H. N., Kelly, C. J., Schroeder, S., et al. (2015). Esophageal Microbiome in Eosinophilic Esophagitis. *PLoS ONE* 10 (5), e0128346. doi:10.1371/journal.pone.0128346
- Hawinkel, S., Mattiello, F., Bijmens, L., and Thas, O. (2019). A Broken Promise: Microbiome Differential Abundance Methods Do Not Control the False Discovery Rate. *Brief Bioinform* 20, 210–221. doi:10.1093/bib/bbx104
- Hawinkel, S., Rayner, J. C. W., Bijmens, L., and Thas, O. (2020). Sequence Count Data Are Poorly Fit by the Negative Binomial Distribution. *PLoS one* 15, e0224909. doi:10.1371/journal.pone.0224909
- Huse, S. M., Mark Welch, D. B., Voorhis, A., Shipunova, A., Morrison, H. G., Eren, A. M., et al. (2014). VAMPS: A Website for Visualization and Analysis of Microbial Population Structures. *BMC Bioinformatics* 15, 41. doi:10.1186/1471-2105-15-41
- Jonsson, V., Österlund, T., Nerman, O., and Kristiansson, E. (2016). Statistical Evaluation of Methods for Identification of Differentially Abundant Genes in Comparative Metagenomics. *BMC genomics* 17, 78–14. doi:10.1186/s12864-016-2386-y
- Keim, D., Kohlhammer, J., Ellis, G., and Mansmann, F. (2010). *Mastering the Information Age: Solving Problems with Visual Analytics*. Goslar, Germany: Eurographics Association.
- Kelley, D. R., Liu, B., Delcher, A. L., Pop, M., and Salzberg, S. L. (2012). Gene Prediction with Glimmer for Metagenomic Sequences Augmented by Classification and Clustering. *Nucleic Acids Res.* 40, e9–12. doi:10.1093/nar/gkr1067
- Kerzner, E., Goodwin, S., Dykes, J., Jones, S., and Meyer, M. (2019). A Framework for Creative Visualization-Opportunities Workshops. *IEEE Trans. Vis. Comput. Graphics* 25, 748–758. doi:10.1109/TVCG.2018.2865241
- Knaflitz, C. N. (2015). *Storytelling with Data: A Data Visualization Guide for Business Professionals*. John Wiley & Sons.
- Knight, R., Vrbanac, A., Taylor, B. C., Aksenov, A., Callewaert, C., Debelius, J., et al. (2018). Best Practices for Analysing Microbiomes. *Nat. Rev. Microbiol.* 16, 410–422. doi:10.1038/s41579-018-0029-9
- Knoll, C., Cetin, A., Moller, T., and Meyer, M. (2020). "Extending Recommendations for Creative Visualization-Opportunities Workshops," in Proceedings - 8th Evaluation and beyond: Methodological Approaches for Visualization, BELIV 2020 (IEEE), 81–88. doi:10.1109/BELIV51497.2020.00017
- Kuntal, B. K., Chandrakar, P., Sadhu, S., and Mande, S. S. (2019a). 'NetShift': a Methodology for Understanding 'driver Microbes' from Healthy and Disease Microbiome Datasets. *ISME J.* 13, 442–454. doi:10.1038/s41396-018-0291-x
- Kuntal, B. K., Dutta, A., and Mande, S. S. (2016). CompNet: A GUI Based Tool for Comparison of Multiple Biological Interaction Networks. *BMC Bioinformatics* 17, 1–11. doi:10.1186/s12859-016-1013-x
- Kuntal, B. K., Gadgil, C., and Mande, S. S. (2019b). Web-gLV: A Web Based Platform for Lotka-Volterra Based Modeling and Simulation of Microbial Populations. *Front. Microbiol.* 10, 288–8. doi:10.3389/fmicb.2019.00288
- Kuntal, B. K., Ghosh, T. S., and Mande, S. S. (2013). Community-Analyzer: A Platform for Visualizing and Comparing Microbial Community Structure across Microbiomes. *Genomics* 102, 409–418. doi:10.1016/j.ygeno.2013.08.004
- Kurtz, Z. D., Müller, C. L., Miraldi, E. R., Littman, D. R., Blaser, M. J., and Bonneau, R. A. (2015). Sparse and Compositionally Robust Inference of Microbial Ecological Networks. *Plos Comput. Biol.* 11, e1004226–25. doi:10.1371/journal.pcbi.1004226
- Lamqaddam, H., Moore, A. V., Abeele, V. V., Brosens, K., and Verbert, K. (2020). Introducing Layers of Meaning (LoM): A Framework to Reduce Semantic Distance of Visualization in Humanistic Research. *IEEE Trans. Vis. Comput. Graph* PP, 1. doi:10.1109/tvcg.2020.3030426
- Langille, M. G., Zaneveld, J., Caporaso, J. G., McDonald, D., Knights, D., Reyes, J. A., et al. (2013). Predictive Functional Profiling of Microbial Communities Using 16s Rrna Marker Gene Sequences. *Nat. Biotechnol.* 31, 814–821. doi:10.1038/nbt.2676
- Lei, Y., Xiao, Y., Li, L., Jiang, C., Zu, C., Li, T., et al. (2017). Impact of Tillage Practices on Soil Bacterial Diversity and Composition Under the Tobacco-Rice Rotation in China. *J. Microbiol.* 55, 349–356. doi:10.1007/s12275-017-6242-9
- Liao, T., Wei, Y., Luo, M., Zhao, G. P., and Zhou, H. (2019). Tmap: An Integrative Framework Based on Topological Data Analysis for Population-Scale Microbiome Stratification and Association Studies. *Genome Biol.* 20, 293. doi:10.1186/s13059-019-1871-4
- Lum, P. Y., Singh, G., Lehman, A., Ishkanov, T., Vajdemo-Johansson, M., Alagappan, M., et al. (2013). Extracting Insights from the Shape of Complex Data Using Topology. *Sci. Rep.* 3, 1236. doi:10.1038/srep01236
- McMurdie, P. J., and Holmes, S. (2013). Phyloseq: An R Package for Reproducible Interactive Analysis and Graphics of Microbiome Census Data. *PLoS ONE* 8, e61217. doi:10.1371/journal.pone.0061217
- McNally, C. P., Eng, A., Noecker, C., Gagne-Maynard, W. C., and Borenstein, E. (2018). BURRITO: An Interactive Multi-Omic Tool for Visualizing Taxa-Function Relationships in Microbiome Data. *Front. Microbiol.* 9, 1–11. doi:10.3389/fmicb.2018.00365

- Munzner, T. (2009). A Nested Model for Visualization Design and Validation. *IEEE Trans. Vis. Comput. Graph* 15, 921–928. doi:10.1109/TVCG.2009.111
- Munzner, T. (2014). *Visualization Analysis and Design*. Boca Raton, FL, USA: CRC Press.
- Nagpal, S., Haque, M. M., Singh, R., and Mande, S. S. (2019). IVikodak-A Platform and Standard Workflow for Inferring, Analyzing, Comparing, and Visualizing the Functional Potential of Microbial Communities. *Front. Microbiol.* 9, 1–15. doi:10.3389/fmicb.2018.03336
- Oliveira, F. S., Brestelli, J., Cade, S., Zheng, J., Iodice, J., Fischer, S., et al. (2018). Microbiomedb: a Systems Biology Platform for Integrating, Mining and Analyzing Microbiome Experiments. *Nucleic Acids Res.* 46, D684–D691. doi:10.1093/nar/gkx1027
- Ondov, B. D., Bergman, N. H., and Phillippy, A. M. (2011). Interactive Metagenomic Visualization in a Web Browser. *BMC Bioinformatics* 12, 385. doi:10.1186/1471-2105-12-385
- Panagiotidou, G., Aerts, J., and Vande Moere, A. (2020). “Goco: A Gamified Activity for Winnowing Visualization Projects with Interdisciplinary Experts,” in IEEE VIS Workshop on Data Vis Activities to Facilitate Learning, Reflecting, Discussing, and Designing, Held in Conjunction with IEEE VIS 2020 (IEEE).
- Prehn-Kristensen, A., Zimmermann, A., Tittmann, L., Lieb, W., Schreiber, S., Baving, L., et al. (2018). Reduced Microbiome Alpha Diversity in Young Patients with Adhd. *PLoS One* 13, e0200728. doi:10.1371/journal.pone.0200728
- Reeder, J., Huang, M., Kaminker, J. S., and Paulson, J. N. (2020). MicrobiomeExplorer: an R Package for the Analysis and Visualization of Microbial Communities. *Bioinformatics* 1, 1–2. doi:10.1093/bioinformatics/btaa838
- Sakai, R., and Aerts, J. (2015). “Card Sorting Techniques for Domain Characterization in Problem-Driven Visualization Research,” in Eurographics Conference on Visualization (EuroVis) (Geneva, Switzerland: Eurographics Association). doi:10.2312/eurovisshort.20151136
- Schloss, P. D., Westcott, S. L., Ryabin, T., Hall, J. R., Hartmann, M., Hollister, E. B., et al. (2009). Introducing Mothur: Open-Source, Platform-independent, Community-Supported Software for Describing and Comparing Microbial Communities. *Appl. Environ. Microbiol.* 75, 7537–7541. doi:10.1128/AEM.01541-09
- Secrier, M., and Schneider, R. (2013). Visualizing Time-Related Data in Biology, a Review. *Brief Bioinform* 15, 771–782. doi:10.1093/bib/bbt021
- Shamsaddini, A., Dadkhah, K., and Gillevet, P. M. (2020). BiomMiner: An Advanced Exploratory Microbiome Analysis and Visualization Pipeline. *PLoS ONE* 15, e0234860–13. doi:10.1371/journal.pone.0234860
- Shaw, G. T., Pao, Y. Y., and Wang, D. (2016). MetaMIS: A Metagenomic Microbial Interaction Simulator Based on Microbial Community Profiles. *BMC Bioinformatics* 17, 1–12. doi:10.1186/s12859-016-1359-0
- Silverman, J. D., Washburne, A. D., Mukherjee, S., and David, L. A. (2017). A Phylogenetic Transform Enhances Analysis of Compositional Microbiota Data. *eLife* 6, 1–20. doi:10.7554/eLife.21887
- Sohn, M. B., and Li, H. (2019). Compositional Mediation Analysis for Microbiome Studies. *Ann. Appl. Stat.* 13, 661–681. doi:10.1214/18-AOAS1210
- Swenson, N. G., and Swenson, M. N. G. (2014). *Package ‘lefse’*. CRAN. Available at: <https://mran.microsoft.com/snapshot/2014-10-25/web/packages/lefse/lefse.pdf>.
- Thas, O., Neve, J. D., Clement, L., and Ottoy, J.-P. (2012). Probabilistic Index Models. *J. R. Stat. Soc. Ser. B: Stat. Methodol.* 74, 623–671. doi:10.1111/j.1467-9868.2011.01020.x
- Tripathi, A., Marotz, C., Gonzalez, A., Vázquez-Baeza, Y., Song, S. J., Bouslimani, A., et al. (2018). Are Microbiome Studies Ready for Hypothesis-Driven Research? *Curr. Opin. Microbiol.* 44, 61–69. doi:10.1016/j.mib.2018.07.002
- Van Wijk, J. J. (2005). “The Value of Visualization,” in *VIS 05. IEEE Visualization, 2005 (IEEE)*, 79–86.
- Vanderweele, T. J., and Vansteelandt, S. (2009). Conceptual Issues Concerning Mediation, Interventions and Composition. *Stat. Its Interf.* 2, 457–468. doi:10.4310/sii.2009.v2.n4.a7
- Vázquez-Baeza, Y., Pirrung, M., Gonzalez, A., and Knight, R. (2013). EMPeror: A Tool for Visualizing High-Throughput Microbial Community Data. *GigaScience* 2, 16. doi:10.1186/2047-217X-2-16
- Wang, C., Hu, J., Blaser, M. J., and Li, H. (2020). Estimating and Testing the Microbial Causal Mediation Effect with High-Dimensional and Compositional Microbiome Data. *Bioinformatics* 36, 347–355. doi:10.1093/bioinformatics/btz565
- Wang, Y., Xu, L., Gu, Y. Q., and Coleman-Derr, D. (2016). MetaCoMET: A Web Platform for Discovery and Visualization of the Core Microbiome. *Bioinformatics* 32, 3469–3470. doi:10.1093/bioinformatics/btw507
- Weiss, S., Xu, Z. Z., Peddada, S., Amir, A., Bittinger, K., Gonzalez, A., et al. (2017). Normalization and Microbial Differential Abundance Strategies Depend upon Data Characteristics. *Microbiome* 5, 27–18. doi:10.1186/s40168-017-0237-y
- Wilck, N., Matus, M. G., Kearney, S. M., Olesen, S. W., Forslund, K., Bartolomeaus, H., et al. (2017). Salt-responsive Gut Commensal Modulates TH17 axis and Disease. *Nature* 551, 585–589. doi:10.1038/nature24628
- Winter, G., Hart, R. A., Charlesworth, R. P. G., and Sharpley, C. F. (2018). Gut Microbiome and Depression: what We Know and what We Need to Know. *Rev. Neurosci.* 29, 629–643. doi:10.1515/revneuro-2017-0072
- Wu, L., Shan, X., Chen, S., Zhang, Q., Qi, Q., Qin, Z., et al. (2020). Progressive Microbial Community Networks With Incremental Organic Loading Rates underlie Higher Anaerobic Digestion Performance. *mSystems* 5, e00357–19. doi:10.1128/mSystems.00357-19
- Zakrzewski, M., Proietti, C., Ellis, J. J., Hasan, S., Brion, M. J., Berger, B., et al. (2017). Calypso: A User-Friendly Web-Server for Mining and Visualizing Microbiome-Environment Interactions. *Bioinformatics* 33, 782–783. doi:10.1093/bioinformatics/btw725
- Zhang, Y. L., Yu, P. C., and Liu, P. (2019). Using High-Throughput Metabolomics to Discover Perturbed Metabolic Pathways and Biomarkers of Allergic Rhinitis as Potential Targets to Reveal the Effects and Mechanism of Geniposide. *RSC advances* 9 (30), 17490–17500. doi:10.1039/C9RA02166C
- Zhu, W., Lomsadze, A., and Borodovsky, M. (2010). Ab Initio gene Identification in Metagenomic Sequences. *Nucleic Acids Res.* 38, e132. doi:10.1093/nar/gkq275

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher’s Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2021 Peeters, Thas, Shkedy, Kodalcı, Musisi, Owokotomo, Dyczko, Hamaad, Vangronsveld, Kleinewietfeld, Thijs and Aerts. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Visualizing Phytochemical-Protein Interaction Networks: *Momordica charantia* and Cancer

Yumi L. Briones^{1*}, Alexander T. Young², Fabian M. Dayrit¹, Armando Jerome De Jesus¹ and Nina Rosario L. Rojas^{1*}

¹Department of Chemistry, Ateneo de Manila University, Quezon City, Philippines, ²Institute of Environmental Science & Meteorology, College of Science, University of the Philippines Diliman, Quezon City, Philippines

OPEN ACCESS

Edited by:

Sean O'Donoghue,
Garvan Institute of Medical Research,
Australia

Reviewed by:

Cagatay Turkay,
University of Warwick,
United Kingdom
William C. Ray,
Nationwide Children's Hospital,
United States

*Correspondence:

Yumi L. Briones
yumi.briones@obf.ateneo.edu
Nina Rosario L. Rojas
nrojas@ateneo.edu

Specialty section:

This article was submitted to
Data Visualization,
a section of the journal
Frontiers in Bioinformatics

Received: 01 September 2021

Accepted: 16 November 2021

Published: 13 December 2021

Citation:

Briones YL, Young AT, Dayrit FM,
De Jesus AJ and Rojas NRL (2021)
Visualizing Phytochemical-Protein
Interaction Networks: *Momordica*
charantia and Cancer.
Front. Bioinform. 1:768886.
doi: 10.3389/fbinf.2021.768886

The *in silico* study of medicinal plants is a rapidly growing field. Techniques such as reverse screening and network pharmacology are used to study the complex cellular action of medicinal plants against disease. However, it is difficult to produce a meaningful visualization of phytochemical-protein interactions (PCPIs) in the cell. This study introduces a novel workflow combining various tools to visualize a PCPI network for a medicinal plant against a disease. The five steps are 1) phytochemical compilation, 2) reverse screening, 3) network building, 4) network visualization, and 5) evaluation. The output is a PCPI network that encodes multiple dimensions of information, including subcellular location, phytochemical class, pharmacokinetic data, and prediction probability. As a proof of concept, we built a PCPI network for bitter melon (*Momordica charantia* L.) against colorectal cancer. The network and workflow are available at <https://yumibriones.github.io/network/>. The PCPI network highlights high-confidence interactions for further *in vitro* or *in vivo* study. The overall workflow is broadly transferable and can be used to visualize the action of other medicinal plants or small molecules against other diseases.

Keywords: network visualization, network pharmacology, reverse screening, medicinal plants, phytochemicals, *Momordica charantia* (bitter melon), colorectal cancer

1 INTRODUCTION

Medicinal plants have been consumed to fight disease since ancient times (Petrovska, 2012). However, even in the modern age, their complex cellular action is not fully understood. Unlike magic bullets that selectively target a given protein, phytochemicals in medicinal plants act on multiple protein targets to restore the overall equilibrium of the cell (Ding et al., 2009). While *in vitro* and *in vivo* methods are often used to study the therapeutic effects of medicinal plants, there is limited experimental data on phytochemical-protein interactions (PCPIs) (Huang et al., 2018). Recently there has been increasing use of *in silico* methods such as reverse screening and network pharmacology in natural products research, as these are well-suited for studying the multi-targeted action of medicinal plants (Chandran et al., 2017).

Reverse screening uses experimentally validated PCPIs to make novel predictions. While conventional screening starts with a target protein and searches for compounds targeting it, reverse screening starts with the compounds (e.g. phytochemicals) and looks for proteins targeted by these compounds (Huang et al., 2018). The ability of reverse screening to predict PCPIs makes it useful for a network pharmacology approach where phytochemicals and proteins are

analyzed as nodes in an interaction network. Of all existing reverse screening tools we are aware of, only one provides a network visualization: Bioinformatics Analysis Tool for Molecular mechANism of Traditional Chinese Medicine (BATMAN-TCM) (Liu et al., 2016). The network shows predicted interactions between phytochemicals, protein targets, and enriched pathways and diseases. However, this does not provide a complete picture of the action of a medicinal plant against a specific disease, which is often the goal of natural products research. It would be useful to see protein-protein interactions (PPIs) between targets to evaluate downstream effects. The network can be better organized by sorting nodes into subcellular compartments. To assess whether phytochemicals can reach these compartments, pharmacokinetic properties are needed. There are existing tools for each of these purposes, but they are all separately found.

Natural products research would greatly benefit from a streamlined workflow that results in a strong (PCPI) network visualization. Thus, we developed a novel workflow combining existing tools to predict and visualize the cellular action of a medicinal plant against a disease. The five-step pipeline consists of 1) phytochemical compilation, 2) reverse screening, 3) network building, 4) network visualization, and 5) evaluation. This outputs a PCPI network that encodes multiple dimensions of information including PPIs, subcellular location, phytochemical class, and pharmacokinetic properties. This makes it easier to determine which predicted PCPIs merit further *in vitro* and *in vivo* study.

As a proof of concept, we applied the workflow to *Momordica charantia* L. (bitter melon) against colorectal cancer. Bitter melon has shown anticancer activity *in vitro* and *in vivo* but has not been thoroughly investigated *in silico* (Raina et al., 2016). Meanwhile, colorectal cancer is a disease known to be highly influenced by diet (Dray et al., 2003). We evaluated select PCPIs by molecular docking and identified high-confidence predictions for further study. Our website (<https://yumibriones.github.io/network/>) contains the PCPI network we generated and a diagram of the workflow with links to all resources used. With this study, we aim to improve the efficiency of natural products research by using readily available tools to produce insightful network visualizations.

2 METHODS

2.1 General Workflow

The general workflow consists of five main steps:

- 1) **Phytochemical compilation:** A medicinal plant is chosen and searched in a phytochemical database and literature to obtain a “Phytochemical list.”
- 2) **Reverse screening:** The “Phytochemical list” is entered in a reverse screening program to obtain a “Complete PCPIs” list.
- 3) **Network building:** Protein targets from the “Complete PCPIs” list are run through pathway enrichment after which a disease is chosen. The “Disease-specific PCPIs” are merged with the existing PPI network Signaling Network Open Resource (SIGNOR) 2.0 to output a “PCPI-SIGNOR disease network.” Information on phytochemical class,

pharmacokinetic properties, subcellular location and protein function are added using various resources.

- 4) **Network visualization:** The “Annotated PCPI-SIGNOR disease network” is visualized using Cytoscape and arranged by subcellular location using the plug-in boundaryLayout. Phytochemical and protein attributes are visualized.
- 5) **Evaluation:** The “PCPI-SIGNOR disease network visualization” is analyzed and notable PCPIs are evaluated *in silico*, *in vitro*, or *in vivo*.

Figure 1 is a detailed diagram of the workflow showing inputs and outputs of each step and all resources used in the study.

The following sections provide more detail for each step, including brief backgrounds on each resource used.

2.2 Phytochemical Compilation

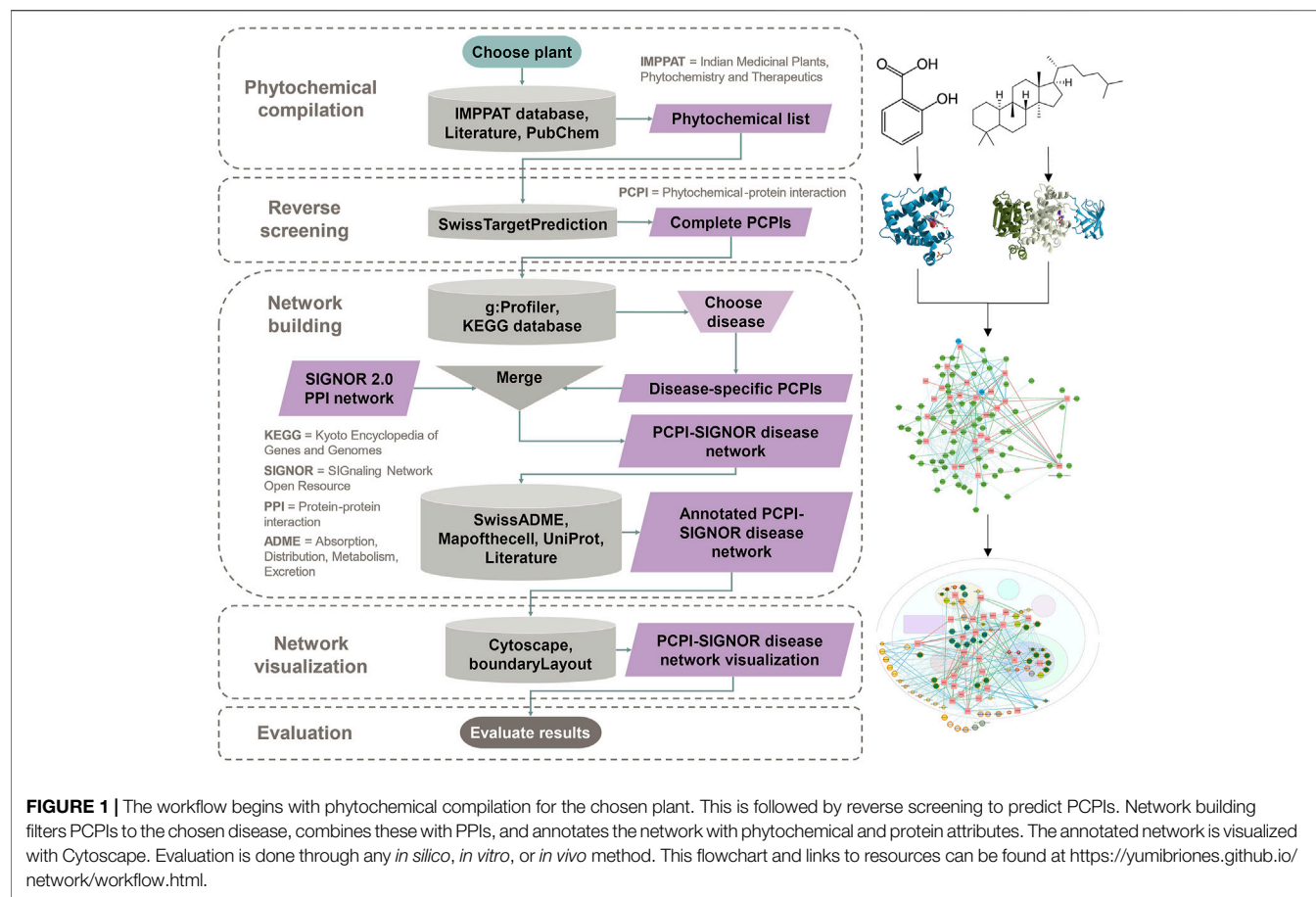
After choosing a medicinal plant to investigate, the plant is entered into the Indian Medicinal Plants, Phytochemistry And Therapeutics (IMPPAT) (<https://cb.imsc.res.in/imppat>) database. IMPPAT contains phytochemical-plant associations mined from medicinal plant books, phytochemical databases, and PubMed abstracts (Mohanraj et al., 2018). Phytochemicals may also be determined from the literature. Positive and negative control molecules may be selected. If drugs are selected as controls, their interactions are referred to as drug-protein interactions (DPIs). The Simplified Molecular Input Line Entry System (SMILES) of all molecules are obtained from PubChem (Kim et al., 2019). Phytochemicals are sorted by class according to Medical Subject Headings (MeSH) Tree (U. S. National Library of Medicine, 2021) or Chemical Entities of Biological Interest (ChEBI) ontology (Hastings et al., 2016).

To simulate metabolism, glycosides (molecules bonded to sugar units) are manually hydrolyzed with molecular editing software such as ChemSketch, developed by Advanced Chemistry Development, Inc. (ACD/Labs). Both glycosides and aglycones (the non-sugar unit) are kept in the list of phytochemicals, combining any duplicate structures into a single entry. The complete resulting list is the “Phytochemical list” from **Figure 1**.

2.3 Reverse Screening

Reverse screening is done with SwissTargetPrediction (<http://www.swisstargetprediction.ch>), a shape screening software that uses ligand-protein binding data from ChEMBL version 23 (Mendez et al., 2019). When a query molecule is entered, SwissTargetPrediction calculates 2D and 3D similarity scores with ligands in the database. Both scores are combined to obtain the probability that the query molecule shares the same protein target as the matched ligands (Daina et al., 2019). If the query molecule is already listed in the ChEMBL database, SwissTargetPrediction assigns a prediction probability of 1.

Molecules in the “Phytochemical list” from the previous step are entered into SwissTargetPrediction using the SMILES, with *Homo sapiens* as the selected organism. The output is a list of predicted protein targets and probability scores for the query molecule which can be downloaded as a CSV file. Only results



with probabilities greater than zero are considered. The combined list of predictions for all molecules in the “Phytochemical list” is the “Complete PCPIs” output.

2.4 Network Building

Network building consists of four steps: 1) pathway enrichment, 2) addition of PPIs and glycoside-aglycone relationships, 3) assignment of subcellular locations, and 4) pharmacokinetic analysis of phytochemicals.

2.4.1 Pathway Enrichment

The program g:Profiler (<https://biit.cs.ut.ee/gprofiler/gost>) (Reimand et al., 2007) is used to identify statistically overrepresented pathways in the set of predicted protein targets from the “Complete PCPIs” list. The protein names are entered as a query, and the search is carried out with *Homo sapiens* as the selected organism, a 0.05 significance threshold, and Kyoto Encyclopedia of Genes and Genomes (KEGG) (Kanehisa and Goto, 2000) as the reference database. Results are downloaded as a CSV file which lists all enriched pathways and intersected proteins per pathway. From the file, the disease of interest is located. The intersected proteins under the disease are used to filter the “Complete PCPIs” list to only the “Disease-specific PCPIs.”

2.4.2 Addition of PPIs and Glycoside-aglycone Relationships

The SIGNOR 2.0 database is used as a source of PPIs. SIGNOR 2.0 is a biological network of literature-based causal interactions between proteins. The entire network is directed from source to target node (Licata et al., 2020). The full *Homo sapiens* database was downloaded on September 28, 2020. Tableau Prep is used to combine the “Disease-specific PCPIs” with PPIs from SIGNOR 2.0 using the disease-specific protein targets as a join clause. Glycoside-aglycone relationships are added to the network as interactions directed from the parent glycoside to child aglycone. The resulting file is the “PCPI-SIGNOR disease network” (Figure 1). In this file, all source nodes are labelled “Entity A” while all target nodes are labelled “Entity B.”

2.4.3 Assignment of Subcellular Locations

All proteins in the “PCPI-SIGNOR disease network” are assigned a subcellular location using an interactive database of the HeLa spatial proteome developed by Itzhak et al. (2016) (<http://mapofthecell.biochem.mpg.de/>). The database is a downloadable Excel file that reports the most probable cellular location of a protein based on fractionation and mass spectrometry experiments. When protein names are entered into the file, the corresponding subcellular locations will appear.

UniProt is used for proteins not in the HeLa database. Protein entries in UniProt contain a “Subcellular location” section based on expert annotations (The UniProt Consortium, 2021). The Gene Ontology (GO) tool is not chosen for this step, as it often outputs a long list of all recorded links between a protein and cellular component with no way to narrow down options (Hill et al., 2008).

Subcellular locations of phytochemicals and controls are assigned in this order of priority:

- 1) ligands with protein targets in the nucleus were placed in the nucleus;
- 2) ligands with protein targets in the mitochondrion were placed in the mitochondrion;
- 3) ligands with protein targets in the plasma membrane were placed in the plasma membrane; and
- 4) ligands with protein targets in the cytoplasm were placed in the cytoplasm.

In the “PCPI-SIGNOR disease network” file, the subcellular locations of “Entity A” and “Entity B” are entered into separate columns labelled “Location A” and “Location B” respectively. This results in an “Annotated PCPI-SIGNOR disease network” file.

2.4.4 Pharmacokinetic Analysis of Phytochemicals

SwissADME (<http://www.swissadme.ch/>) assesses physicochemical and pharmacokinetic parameters of input molecules (Daina et al., 2017). All phytochemicals and controls included in the “Disease-specific PCPIs” are entered into SwissADME using their name and SMILES. The results are a list of pharmacokinetic data for each molecule. The results are downloaded as a CSV file and the following attributes are noted: Abbott bioavailability score, gastrointestinal (GI) absorption (for orally ingested medicinal plants), and lipophilicity using the partition coefficient $\log p$ (Eq. 1). A more lipophilic compound would have a higher $\log p$ value.

$$\log P = \log_{10} \frac{[\text{concentration of solute in octanol}]}{[\text{concentration of solute in water}]} \quad (1)$$

Each pharmacokinetic parameter is entered as its own column in the “Annotated PCPI-SIGNOR disease network” file. Columns modifying “Entity A” or “Entity B” are ended with “A” or “B” respectively (e.g. “Bioavailability A”).

2.5 Network Visualization

The “Annotated PCPI-SIGNOR disease network” Excel file is loaded into Cytoscape 3.6.0 (Shannon et al., 2003). All duplicate edges and self-loops are removed.

For edges, these parameters are followed:

- 1) edge thickness is mapped to the SwissTargetPrediction probability score (thicker edges = more probable); and
- 2) edge color is mapped to interaction type (predicted PCPI or DPI = blue, PPI upregulation = green, PPI downregulation = red, glycoside-aglycone relation = dark green dashed line).

For ligand nodes, these parameters are followed:

- 1) node shape is set to circle;
- 2) node transparency is mapped to $\log P$ value (lower $\log P$ = more transparent, higher $\log P$ = more opaque);
- 3) node size is mapped to GI absorption (high absorption = large, low absorption = small);
- 4) node border color is mapped to Abbott bioavailability score (lowest scores in red, highest scores in green); and
- 5) node color was mapped to ligand class.

For protein nodes, these parameters are followed:

- 1) node shape is set to square;
- 2) node color is set to pink; and
- 3) label color is mapped to protein function (red = oncogene protein, green = tumor suppressor, black = other protein).

Nodes are automatically organized into a cell template based on the assigned cellular location using the Cytoscape plug-in boundaryLayout, developed by University of California San Francisco’s Resource for Biocomputing, Visualization, and Informatics (UCSF RBVI).

Supplementary Figure S1 shows the evolution of the network visualization in graphical form. The complete PCPI network and detailed legend are shown in Figure 2 in the Results section. We visualized the network in two ways: with a white background (Figure 2) and a dark background (Supplementary Figure S2).

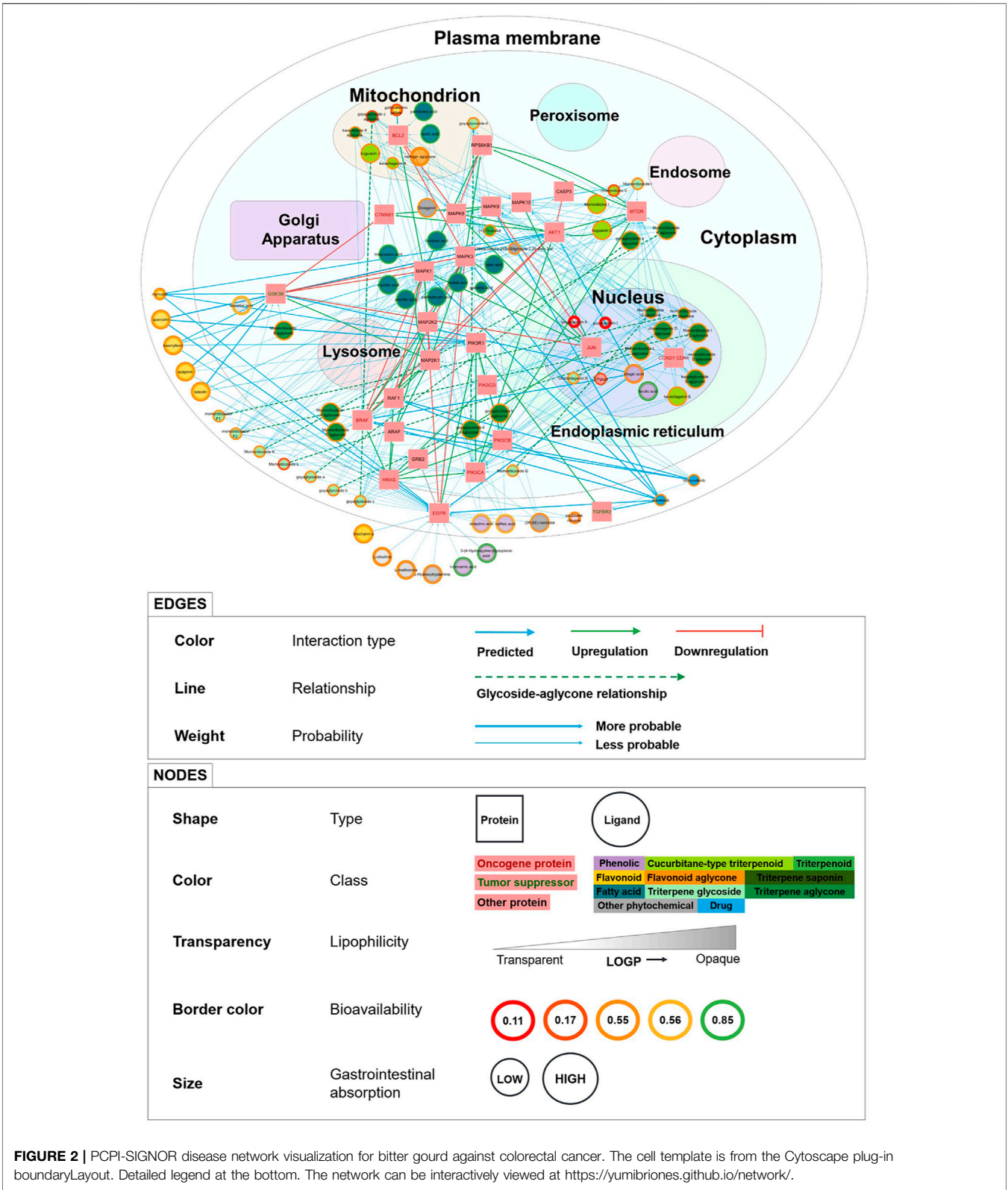
2.6 Evaluation Through Docking

Evaluation of interactions in the “PCPI-SIGNOR disease network visualization” may be done *in silico*, *in vitro*, or *in vivo*. We chose to evaluate PCPIs *in silico* through molecular docking with Autodock Vina, which outperforms its predecessor AutoDock 4 in speed and accuracy (Trott and Olson, 2010).

Protein structures were downloaded from the Protein Data Bank (PDB) (Berman et al., 2000). We used the Auto in silico Consensus Inverse Docking (ACID) server to guide our PDB structure selection (Wang et al., 2019). ACID contains a curated set of protein targets according to the following restrictions:

- 1) no structures with resolution larger than 3.0 Å;
- 2) no structures solved by Nuclear Magnetic Resonance (NMR) (structures are all solved by X-ray diffraction for uniformity);
- 3) no structures with ligands containing nonstandard atoms (e.g. Si, Be); and
- 4) structures must have only one drug-like ligand bound in the active site.

Structures of the bound inhibitors were obtained from PDB while phytochemical structures were obtained from PubChem. Protein and ligand structures were prepared for docking with Autodock Tools. We manually calculated grid boxes using Autodock Tools, centering the box on the bound ligand in the active site. In the absence of a bound inhibitor, protein structure



was analyzed with Aquaria (<http://aquaria.ws/>), which aligns UniProt sequence with a chosen PDB structure and highlights features such as binding site (O’Donoghue et al., 2015).

For each protein, we docked the inhibitor bound to the original PDB structure as a positive control before docking phytochemicals. Results were visualized in 3D with ChimeraX

(Pettersen et al., 2021) and in 2D with LigPlot+ (Laskowski and Swindells, 2011).

3 RESULTS

This section details results for our proof-of-concept study, where we applied the workflow to visualize a PCPI network for bitter gourd against colorectal cancer.

3.1 Phytochemical Compilation

We compiled 169 phytochemicals found in the fruit, seeds, and leaves of bitter gourd. These were taken from IMPPAT and reviews by Raina et al. (2016), Jia et al. (2017), and Mozaniel et al. (2018). Most were phenolic acids, triterpene glycosides, and aglycones. For positive controls, we selected the chemotherapy drugs vemurafenib (a selective B-raf inhibitor) and sorafenib (a multi-kinase inhibitor). Meanwhile for negative controls, we chose alprazolam (a benzodiazepine), tolinaftate (an antifungal), and tigecycline (a tetracycline antibiotic), all of which have similar structures to phytochemicals but are not expected to act on colorectal cancer signaling. In total, 174 ligands were compiled for screening. The phytochemical list is shown in **Supplementary Table S1** and summarized in **Supplementary Figure S3**.

3.2 Reverse Screening

SwissTargetPrediction predicted 6937 PCPIs with nonzero probability between 166 phytochemicals and 772 protein targets. No matches were found for (+)-catechin, (-)-epicatechin, and the *cis*-zeatin riboside aglycone.

For negative controls, SwissTargetPrediction predicted 52 DPIs for alprazolam, 7 DPIs for tolinaftate, and 17 DPIs for tigecycline with nonzero probability. The top predicted targets for alprazolam were GABA receptors, consistent with experimental knowledge. For tolinaftate and tigecycline, human targets were identified because of structural similarity to other molecules. SwissTargetPrediction may identify false positives, highlighting the need for an evaluation step.

For positive controls, SwissTargetPrediction predicted 100 DPIs for vemurafenib and 100 DPIs for sorafenib, all with nonzero probability. For sorafenib, all results had probability = 1 with targets being mostly protein kinases, consistent with experimental knowledge. For vemurafenib, there were only four results with probability = 1 including the experimentally known target B-Raf proto-oncogene, serine/threonine kinase (B-raf). This demonstrates the reliability of SwissTargetPrediction as a reverse screening tool. The “Complete PCPIs” list is shown in **Supplementary Table S2**.

3.3 Network Building

All g:Profiler results are listed in **Supplementary Table S3** with the top ten results shown in **Supplementary Figure S4**. Pathway enrichment of phytochemical targets identified 23 protein targets in the KEGG colorectal cancer entry. These proteins were involved in the epidermal growth factor receptor (EGFR)/mitogen-activated protein kinase (MAPK) and

phosphatidylinositol-4,5-bisphosphate 3-kinase (PI3K)/protein kinase B (Akt) pathways, Wnt-related integration site (Wnt) signaling, apoptosis and cell cycle regulation. The disease-specific protein targets included oncogene proteins like catenin beta 1 (CTNNB1) and B-raf and the tumor suppressor glycogen synthase kinase 3 beta (GSK3b). Protein classifications are listed in **Supplementary Table S4**.

A separate g:Profiler analysis for the negative controls alprazolam, tolinaftate, and tigecycline found no protein targets involved in the KEGG colorectal cancer pathway. Meanwhile, pathway enrichment of positive controls vemurafenib and sorafenib identified four additional protein targets involved in KEGG colorectal cancer: A-Raf proto-oncogene, serine/threonine kinase (A-Raf), Raf-1 proto-oncogene, serine/threonine kinase (Raf-1), mitogen-activated protein kinase kinase 2 (MAP2K2), and transforming growth factor beta receptor 2 (TGFB2) (**Supplementary Table S3**).

In total, the KEGG colorectal cancer PCPI-SIGNOR network contained 98 nodes (69 phytochemicals, 2 drugs, and 27 proteins) and 331 interactions (251 PCPIs, 60 PPIs, 10 DPIs, and 10 glycoside-aglycone relationships). The PCPI network and legend are shown in **Figure 2**. The dark version of the network can be viewed at <https://yumibriones.github.io/network/> (**Supplementary Figure S2**). **Supplementary Table S5** contains the data used to build the “Annotated disease-specific PCPI-SIGNOR network.”

3.4 Network Visualization

Figure 2 shows the “PCPI-SIGNOR disease network visualization” for bitter gourd against colorectal cancer.

Our PCPI network has a number of advantages over other visualization methods for medicinal plant interactions. The reverse screening tool BATMAN-TCM represents phytochemicals, proteins, pathways and diseases as nodes in a simple network. Yi et al. (2018) have also documented a workflow resulting in a visualization similar to BATMAN-TCM. However, natural products research often aims to study the action of a medicinal plant against a specific disease. These simple visualizations lack the information needed to address this problem, and additional information is presented in other diagrams or in the text of the paper. Meanwhile, our PCPI network presents plenty of information in a single diagram designed to be intuitively understood by biologists.

One clear advantage of our visualization is that nodes are sorted by subcellular compartment, highlighting which phytochemicals have targets in specific organelles (**Figure 2**). For instance, phytochemicals in the mitochondrion must target B-cell lymphoma 2 (Bcl-2). Seeing subcellular location makes it easier for biologists to identify the roles of proteins in the network.

Another major advantage is the display of pharmacokinetic properties to help assess whether phytochemicals are able to reach protein targets in the cell. High investigation priority may be given to phytochemicals with larger nodes (high GI absorption) and green or orange borders (high or medium bioavailability). Seeing phytochemical classifications is also

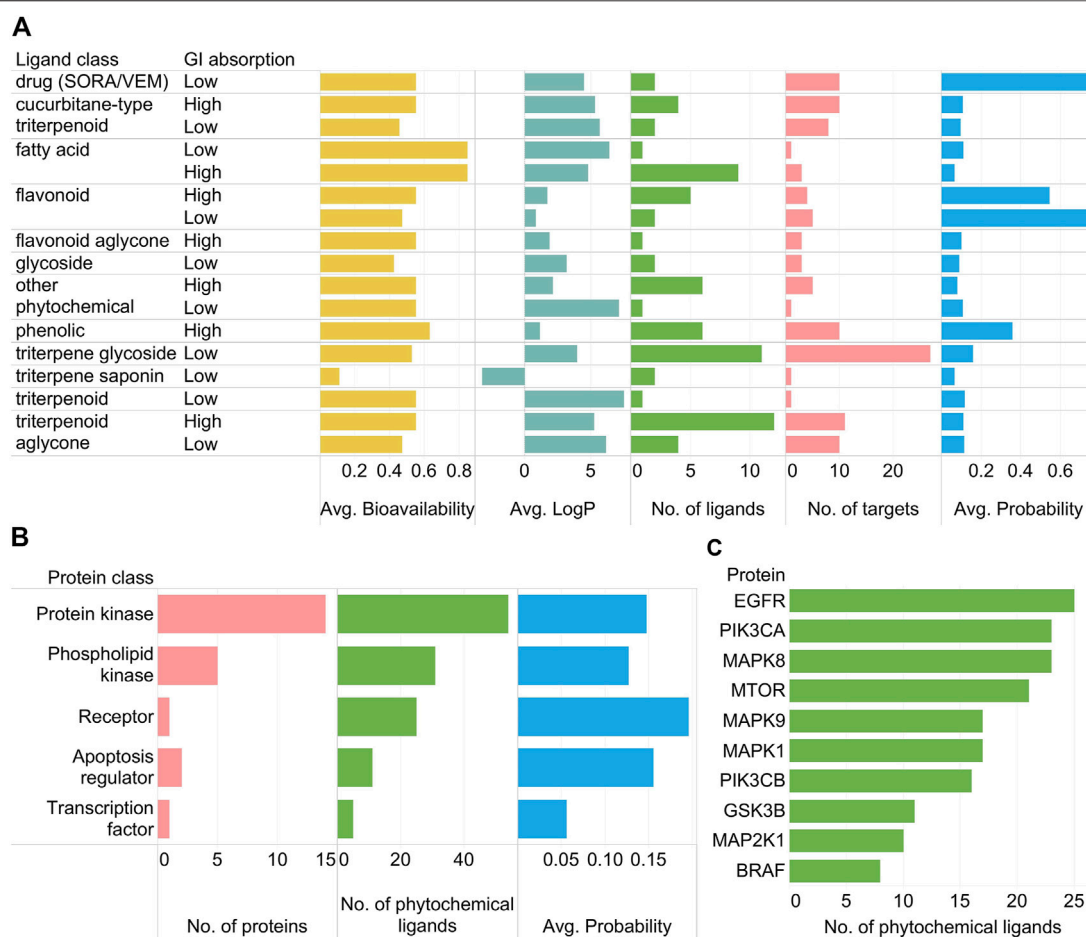


FIGURE 3 | (A) Relationships of SwissADME parameters (high/low GI absorption, average bioavailability, average log *p*) and SwissTargetPrediction results (number of ligands, number of protein targets, average probability) per phytochemical class as well as the two positive control drugs vemurafenib and sorafenib (SORA/VEM) in the KEGG colorectal cancer PCPI-SIGNOR network. **(B)** Number of proteins, number of phytochemical ligands, and average probability scores per protein class. Proteins targeted only by SORA/VEM are not included in the subfigure. **(C)** Top ten most targeted proteins by phytochemical ligands (excluding SORA/VEM).

helpful, as priority can be given to classes such as triterpenoids and flavonoids which are more unique to bitter gourd.

Our visualization also conveys information through edges. The thickest edges (SwissTargetPrediction probability = 1) represent interactions already recorded in ChEMBL. Novel predictions would have thinner edges. We can also see relationships between phytochemicals and their metabolism products by following the dashed arrows. Interactions between proteins are represented with green or red arrows for up or downregulation, revealing the downstream effects of a phytochemical beyond its direct protein target.

To illustrate how these advantages come together, here is an important insight we can get from **Figure 2**. Triterpene glycosides (light green) are all small nodes mostly in the plasma membrane. However, following the dashed arrows reveals that many aglycone products (dark green) have large nodes and are in the nucleus and cytoplasm. This tells us that aglycones generally have higher GI absorption than glycosides with targets deeper in the cell. This supports

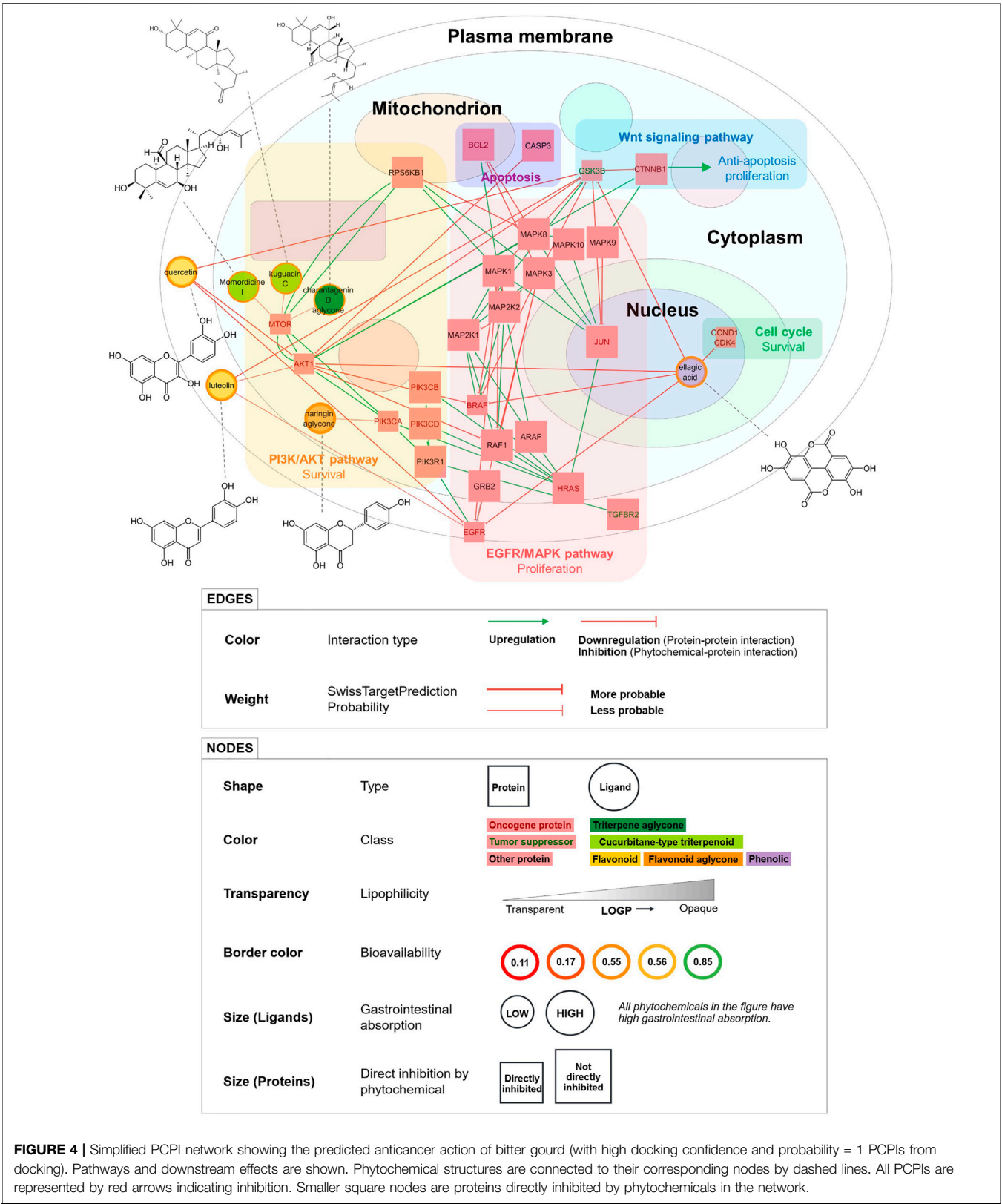
experimental knowledge that aglycones are better absorbed than their glycoside parents (Bhattacharya, 2019).

Important trends in the PCPI network can be summarized using standard bar graphs as in **Figure 3**.

As observed in **Figure 2**, triterpene glycosides were highly abundant but had low GI absorption while aglycones had high GI absorption (**Figure 3A**). Protein kinases were abundant and highly targeted by phytochemicals (**Figure 3B**). Highly targeted proteins include EGFR and the mechanistic target of rapamycin kinase (mTOR) (**Figure 3C**), though this is already apparent from **Figure 2**. While bar graphs can reveal general trends in the data, the network visualization shows these trends while also showing specific interactions. **Figure 2** alone can already highlight PCPIs to evaluate further *in vitro*, *in vivo*, or *in silico*.

3.5 Evaluation by Molecular Docking

We used Autodock Vina (Vina hereafter) to dock 28 PCPIs and 6 DPIs in the KEGG colorectal cancer PCPI-SIGNOR network. We



chose phytochemicals with high GI absorption from various classes including phenolic acids, triterpenoids, flavonoids, fatty acids, and aglycones. Proteins were selected from the EGFR/MAPK and PI3K/Akt pathways, Wnt signaling, apoptosis, and the cell cycle. Only the top pose from Vina was considered. Detailed docking information is listed in **Supplementary Table S6**.

For positive docking controls, we docked each protein to its bound inhibitor from the PDB structure. We found that predicted poses from Vina were visually similar to experimental poses. Docking interaction energies were generally more negative for bound inhibitor-protein pairs versus phytochemical-protein pairs (**Supplementary Figure S5**). The positive controls vemurafenib and sorafenib docked with highly negative energies comparable to the bound inhibitors. We concluded that Vina predicted binding poses with fairly high accuracy.

Flavonoids and phenolics docked to the adenosine triphosphate (ATP)-binding sites of protein kinases with highly negative docking interaction energies, suggesting competitive inhibition of kinase activity. On the other hand, triterpenoids generally had less negative docking interaction energies when docked to the ATP-binding site. This suggests that flavonoids and phenolics have a high potential for *in vitro* or *in vivo* activity.

To quantify this, we assigned confidence levels to PCPIs based on docking interaction energy (“docking confidence” hereafter) (**Supplementary Figure S6**). Among the PCPIs with probability = 1, we set the most negative docking interaction energy as the “soft cutoff” (−7.8 kcal/mol). The upper bound of the 99.7% confidence interval (CI) (−6.4 kcal/mol) was set as the “hard cutoff.” Interactions were classified as follows:

- 1) High docking confidence: *docking interaction energy*, $E < -7.8 \text{ kcal/mol}$ (soft cutoff);
- 2) Medium docking confidence: $-7.8 < E < -6.4 \text{ kcal/mol}$ (hard cutoff);
- 3) Low docking confidence: $E > -6.4 \text{ kcal/mol}$.

Most flavonoid-protein interactions had high docking confidence while triterpenoid-protein interactions had low docking confidence (**Supplementary Figure S7**). Interestingly however, all interactions between triterpenoids and mTOR had high docking confidence.

We then used docking confidence to calculate “probability confidence” regions based on SwissTargetPrediction probability. We took the mean probability values of each docking confidence level and calculated the 68% CI (equivalent to 1 standard deviation) for each mean (**Supplementary Figure S8**). Detailed calculations are shown in **Supplementary Table S6**. Probability confidence regions were assigned as follows:

- 1) High probability confidence: *probability*, $P > 0.1263$ (upper bound of the mean probability of low docking confidence interactions);
- 2) Uncertain probability confidence: $0.1263 > P > 0.0774$ (lower bound of the mean probability of medium docking confidence interactions);

- 3) Low probability confidence: $P < 0.0774$.

We then sorted each interaction in the KEGG colorectal cancer PCPI-SIGNOR network according to probability confidence regions (**Supplementary Table S5**). Flavonoids were most abundant in the high probability confidence region, triterpenoid aglycones were abundant in the uncertain region, and triterpene glycosides were abundant in the low probability confidence region. Protein kinases were highly targeted in all probability confidence regions (**Supplementary Figure S9**).

Docking results are color-coded according to the legend in **Supplementary Figure 10**, and all visualizations are shown in **Supplementary Figures S11–27**.

3.6 Simplified PCPI Network for Anticancer Action of Bitter Gourd

We visualized a smaller PCPI network including only high docking confidence and probability = 1 interactions (**Figure 4**). This is a simplified model of the predicted anticancer action of bitter gourd. Phytochemicals in this diagram are strong candidates for *in vitro* and *in vivo* activity.

Ellagic acid, a phenolic compound, was predicted to inhibit the most proteins and pathways including the cell cycle, EGFR/MAPK pathway, and PI3K/Akt pathway. Ellagic acid was also predicted to inhibit the tumor suppressor GSK3b, but interestingly, experiments show that inhibition of GSK3b may in fact decrease cancer cell proliferation (Marchand et al., 2012). Meanwhile, the flavonoids quercetin and luteolin were predicted to inhibit the same proteins and pathways including PI3K/Akt and EGFR/MAPK, thereby inhibiting cell survival and proliferation. The triterpenoids momordicine I, kuguacin C, and the charantagenin D aglycone were all predicted to inhibit the PI3K/Akt pathway via mTOR. We highly recommend that these predicted interactions be studied further through *in vitro* and *in vivo* experiments. The phytochemicals in **Figure 4** may also be used as marker compounds for medicinal formulations of bitter gourd.

This figure demonstrates the ability of our workflow to visualize high-confidence PCPI predictions as a detailed yet intuitive network. The workflow can be used to create PCPI networks for other medicinal plants and diseases. If small molecule drugs are searched together with medicinal plants, the PCPI network can even identify shared protein targets and potential interaction effects. Unlike the integrated tool BATMAN-TCM, our modular workflow allows researchers to use other tools at any step. However, we recommend using the tools presented in this study as these were carefully selected. The workflow and links to all resources are available at <https://yumibriones.github.io/network/workflow.html>.

4 CONCLUSION

We developed a novel workflow to visualize the predicted cellular action of a medicinal plant against a disease. We combined select tools into a five-step pipeline: phytochemical compilation, reverse

screening, network building, network visualization, and evaluation. The resulting phytochemical-protein interaction (PCPI) network visually reflects protein-protein interactions, subcellular location, phytochemical class, pharmacokinetic data, and other attributes in a single figure. By clearly communicating all these attributes visually, the network helps users identify interactions worth evaluating further. Our proof-of-concept study on bitter melon against colorectal cancer identified triterpenoid aglycones and flavonoids as key players in the network. The PCPI network and workflow are available at <https://yumibriones.github.io/network/>. We evaluated select PCPIs through docking to produce a smaller network of high-confidence interactions that can be validated *in vitro* and *in vivo*. Overall, this workflow streamlines natural products research by using readily available tools to visualize a rich, intuitive PCPI network.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/**Supplementary Material**, further inquiries can be directed to the corresponding authors.

REFERENCES

- Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., et al. (2000). The Protein Data Bank. *Nucleic Acids Res.* 28, 235–242. doi:10.1093/nar/28.1.235
- Bhattacharya, A. (2019). “Chapter 5 - High-Temperature Stress and Metabolism of Secondary Metabolites in Plants,” in *Effect of High Temperature on Crop Productivity and Metabolism of Macro Molecules*. Editor A. Bhattacharya (Academic Press), 391–484. doi:10.1016/B978-0-12-817562-0.00005-7
- Chandran, U., Mehendale, N., Patil, S., Chaguturu, R., and Patwardhan, B. (2017). Network Pharmacology. *Innovative Approaches Drug Discov.*, 127. doi:10.1016/B978-0-12-801814-9.00005-2
- Daina, A., Michielin, O., and Zoete, V. (2017). SwissADME: a Free Web Tool to Evaluate Pharmacokinetics, Drug-Likeness and Medicinal Chemistry Friendliness of Small Molecules. *Scientific Rep.* 7, 42717. doi:10.1038/srep42717
- Daina, A., Michielin, O., and Zoete, V. (2019). SwissTargetPrediction: Updated Data and New Features for Efficient Prediction of Protein Targets of Small Molecules. *Nucleic Acids Res.* 47, W357–W364. doi:10.1093/nar/gkz382
- Ding, H., Tauzin, S., and Hoessli, D. C. (2009). Phytochemicals as Modulators of Neoplastic Phenotypes. *Pathobiology: J. Immunopathology, Mol. Cell Biol.* 76, 55–63. doi:10.1159/000201674
- Dray, X., Boutron-Ruault, M.-C., Bertrais, S., Sapinho, D., Benhamiche-Bouvier, A.-M., and Faivre, J. (2003). Influence of Dietary Factors on Colorectal Cancer Survival. *Gut* 52, 868–873. doi:10.1136/gut.52.6.868
- Hastings, J., Owen, G., Dekker, A., Ennis, M., Kale, N., Muthukrishnan, V., et al. (2016). ChEBI in 2016: Improved Services and an Expanding Collection of Metabolites. *Nucleic Acids Res.* 44, D1214–D1219. doi:10.1093/nar/gkv1031
- Hill, D. P., Smith, B., McAndrews-Hill, M. S., and Blake, J. A. (2008). Gene Ontology Annotations: what They Mean and where They Come from. *BMC Bioinformatics* 9, S2. doi:10.1186/1471-2105-9-S2
- Huang, H., Zhang, G., Zhou, Y., Lin, C., Chen, S., Lin, Y., et al. (2018). Reverse Screening Methods to Search for the Protein Targets of Chemopreventive Compounds. *Front. Chem.* 6, 1–28. doi:10.3389/fchem.2018.00138
- Itzhak, D. N., Tyanova, S., Cox, J., and Borner, G. H. (2016). Global, Quantitative and Dynamic Mapping of Protein Subcellular Localization. *eLife* 5, e16950. doi:10.7554/eLife.16950
- Jia, S., Shen, M., Zhang, F., and Xie, J. (2017). Recent Advances in Momordica Charantia: Functional Components and Biological Activities. *Int. J. Mol. Sci.* 18, 1–25. doi:10.3390/ijms18122555
- Kanehisa, M., and Goto, S. (2000). KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Res.* 28, 27–30.
- Kim, S., Chen, J., Cheng, T., Gindulyte, A., He, J., He, S., et al. (2019). PubChem 2019 Update: Improved Access to Chemical Data. *Nucleic Acids Res.* 47, D1102–D1109. doi:10.1093/nar/gky1033
- Laskowski, R. A., and Swindells, M. B. (2011). LigPlot+: Multiple Ligand-Protein Interaction Diagrams for Drug Discovery. *J. Chem. Inf. Model.* 51, 2778–2786. doi:10.1021/ci200227u
- Licata, L., Lo Surdo, P., Iannuccelli, M., Palma, A., Micarelli, E., Perfetto, L., et al. (2020). SIGNOR 2.0, the SIGNaling Network Open Resource 2.0: 2019 Update. *Nucleic Acids Res.* 48, D504–D510. doi:10.1093/nar/gkz949
- Liu, Z., Guo, F., Wang, Y., Li, C., Zhang, X., Li, H., et al. (2016). BATMAN-TCM: a Bioinformatics Analysis Tool for Molecular Mechanism of Traditional Chinese Medicine. *Scientific Rep.* 6, 21146. doi:10.1038/srep21146
- Marchand, B., Tremblay, I., Cagnol, S., and Boucher, M.-J. (2012). Inhibition of Glycogen Synthase Kinase-3 Activity Triggers an Apoptotic Response in Pancreatic Cancer Cells through JNK-dependent Mechanisms. *Carcinogenesis* 33, 529–537. doi:10.1093/carcin/bgr309
- Mendez, D., Gaulton, A., Bento, A. P., Chambers, J., De Veij, M., Félix, E., et al. (2019). ChEMBL: towards Direct Deposition of Bioassay Data. *Nucleic Acids Res.* 47, D930–D940. doi:10.1093/nar/gky1075
- Mohanraj, K., Karthikeyan, B. S., Vivek-Ananth, R. P., Chand, R. P. B., Aparna, S. R., Mangalapandi, P., et al. (2018). IMPPAT: A Curated Database of Indian Medicinal Plants, Phytochemistry and Therapeutics. *Scientific Rep.* 8, 4329. doi:10.1038/s41598-018-22631-z
- Mozaniel, S. d. O., Wanessa, A. d. C., Fernanda, W. F. B., Marilena, E. A., Gracilda, C. F., and Raul, N. d. C. J. (2018). Phytochemical Profile and Biological Activities of Momordica Charantia L. (Cucurbitaceae): A Review. *Afr. J. Biotechnol.* 17, 829–846. doi:10.5897/AJB2017.16374
- O'Donoghue, S. I., Sabir, K. S., Kalemanov, M., Stolte, C., Wellmann, B., Ho, V., et al. (2015). Aquaria: Simplifying Discovery and Insight from Protein Structures. *Nat. Methods* 12, 98–99. doi:10.1038/nmeth.3258

AUTHOR CONTRIBUTIONS

YB wrote the manuscript, carried out the experiments, and visualized the network. AY guided the reverse screening aspect of the study. NR, FD, and AD guided the biochemical, natural products, and molecular docking aspects of the work, respectively. All authors read and approved the manuscript.

ACKNOWLEDGMENTS

We are deeply grateful to the Ateneo de Manila University for supporting this study. We also thank the organizers and participants of the Visualizing Biological Data (VIZBI) 2021 conference whose valuable suggestions have helped improve our work.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fbinf.2021.768886/full#supplementary-material>

- Petrovska, B. B. (2012). Historical Review of Medicinal Plants' Usage. *Pharmacognosy Rev.* 6, 1–5. doi:10.4103/0973-7847.95849
- Pettersen, E. F., Goddard, T. D., Huang, C. C., Meng, E. C., Couch, G. S., Croll, T. I., et al. (2021). UCSF ChimeraX: Structure Visualization for Researchers, Educators, and Developers. *Protein Sci.* 30, 70–82. doi:10.1002/pro.3943
- Raina, K., Kumar, D., and Agarwal, R. (2016). Promise of Bitter Melon (*Momordica Charantia*) Bioactives in Cancer Prevention and Therapy. *Semin. Cancer Biol.* 40–41, 116–129. doi:10.1016/j.semcancer.2016.07.002
- Reimand, J., Kull, M., Peterson, H., Hansen, J., and Vilo, J. (2007). g:Profiler—a Web-Based Toolset for Functional Profiling of Gene Lists from Large-Scale Experiments. *Nucleic Acids Res.* 35, W193. doi:10.1093/nar/gkm226
- Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., et al. (2003). Cytoscape: A Software Environment for Integrated Models of Biomolecular Interaction Networks. *Genome Res.* 13, 2498–2504. doi:10.1101/gr.1239303
- The UniProt Consortium (2021). UniProt: the Universal Protein Knowledgebase in 2021. *Nucleic Acids Res.* 49, D480–D489. doi:10.1093/nar/gkaa1100
- Trott, O., and Olson, A. J. (2010). AutoDock Vina: Improving the Speed and Accuracy of Docking with a New Scoring Function, Efficient Optimization and Multithreading. *J. Comput. Chem.* 31, 455–461. doi:10.1002/jcc.21334
- U.S. National Library of Medicine (2021). *Mesh Browser*. Bethesda, MD: U.S. National Library of Medicine.
- Wang, F., Wu, F.-X., Li, C.-Z., Jia, C.-Y., Su, S.-W., Hao, G.-F., et al. (2019). ACID: a Free Tool for Drug Repurposing Using Consensus Inverse Docking Strategy. *J. Cheminformatics* 11, 73. doi:10.1186/s13321-019-0394-z
- Yi, F., Li, L., Xu, L.-j., Meng, H., Dong, Y.-m., Liu, H.-b., et al. (2018). In Silico approach in Reveal Traditional Medicine Plants Pharmacological Material Basis. *Chin. Med.* 13, 33. doi:10.1186/s13020-018-0190-0

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2021 Briones, Young, Dayrit, De Jesus and Rojas. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Extending Association Rule Mining to Microbiome Pattern Analysis: Tools and Guidelines to Support Real Applications

Agostinetto Giulia^{1*}, Sandionigi Anna², Bruno Antonia¹, Pescini Dario³ and Casiraghi Maurizio¹

¹Department of Biotechnology and Biosciences, University of Milano-Bicocca, Milan, Italy, ²Quantia Consulting Srl, Milan, Italy, ³Department of Statistics and Quantitative Methods, University of Milano-Bicocca, Milan, Italy

OPEN ACCESS

Edited by:

Lydia Gregg,
Johns Hopkins University,
United States

Reviewed by:

Kazuhiro Takemoto,
Kyushu Institute of Technology, Japan
Vincenzo Bonnici,
University of Parma, Italy

*Correspondence:

Agostinetto Giulia
giulia.agostinetto@unimib.it

Specialty section:

This article was submitted to
Data Visualization,
a section of the journal
Frontiers in Bioinformatics

Received: 13 October 2021

Accepted: 07 December 2021

Published: 10 January 2022

Citation:

Giulia A, Anna S, Antonia B, Dario P
and Maurizio C (2022) Extending
Association Rule Mining to Microbiome
Pattern Analysis: Tools and Guidelines
to Support Real Applications.
Front. Bioinform. 1:794547.
doi: 10.3389/fbinf.2021.794547

Boosted by the exponential growth of microbiome-based studies, analyzing microbiome patterns is now a hot-topic, finding different fields of application. In particular, the use of machine learning techniques is increasing in microbiome studies, providing deep insights into microbial community composition. In this context, in order to investigate microbial patterns from 16S rRNA metabarcoding data, we explored the effectiveness of Association Rule Mining (ARM) technique, a supervised-machine learning procedure, to extract patterns (in this work, intended as groups of species or taxa) from microbiome data. ARM can generate huge amounts of data, making spurious information removal and visualizing results challenging. Our work sheds light on the strengths and weaknesses of pattern mining strategy into the study of microbial patterns, in particular from 16S rRNA microbiome datasets, applying ARM on real case studies and providing guidelines for future usage. Our results highlighted issues related to the type of input and the use of metadata in microbial pattern extraction, identifying the key steps that must be considered to apply ARM consciously on 16S rRNA microbiome data. To promote the use of ARM and the visualization of microbiome patterns, specifically, we developed microFIM (microbial Frequent Itemset Mining), a versatile Python tool that facilitates the use of ARM integrating common microbiome outputs, such as taxa tables. microFIM implements interest measures to remove spurious information and merges the results of ARM analysis with the common microbiome outputs, providing similar microbiome strategies that help scientists to integrate ARM in microbiome applications. With this work, we aimed at creating a bridge between microbial ecology researchers and ARM technique, making researchers aware about the strength and weaknesses of association rule mining approach.

Keywords: pattern mining, microbiome data, DNA metabarcoding, microbiome patterns, machine learning, association rule mining

1 INTRODUCTION

Studying microbiome patterns is now a hot-topic in different fields of application (Kyrpides et al., 2016; Wood-Charlson et al., 2020). From ecology to medicine, microbiomes are undoubtedly a cornerstone of research, acknowledged as being key participants in all ecosystems, including the human one (Duvallet et al., 2017; Layeghifard et al., 2017). In recent years, DNA sequencing strategies have become one of the main sources for studying microbial communities (Wood-Charlson et al., 2020). Further, 16S rRNA metabarcoding is currently the preferential method to obtain great amounts of information in a time and cost effective manner (Wood-Charlson et al., 2020), becoming one of the primary sources of data regarding microbiome studies (Gonzalez et al., 2018; Knight et al., 2018; Bokulich et al., 2020; Mitchell et al., 2020).

In this context, data mining approaches seem to be newfangled solutions for disclosing and understanding microbial ecosystems (Wood-Charlson et al., 2020; Galimberti et al., 2021; Ghannam and Techtman, 2021). Spanning from classification and signature extraction to interaction and trait associations (Pasolli et al., 2016; Qu et al., 2019), data mining strategies can identify hidden patterns that may help to predict biological functions (Noor et al., 2019; Thomposon et al., 2019). Investigating patterns and exploring their role in functional and predictive aspects are now pivotal to proxy the knowledge of microbial associations, both disentangling interactions and niche specialization (Chaffron et al., 2010; Faust and Raes, 2012; Ma et al., 2020).

Considering the size and complexity of High-Throughput Sequencing (HTS) 16S rRNA metabarcoding data, interpretation and summarization are not straightforward (Naulaerts et al., 2015) and, for this reason, pattern mining strategies have become essential for researchers to disentangle the high amount of information (Kyrpides et al., 2016; Wood-Charlson et al., 2020; Ghannam and Techtman, 2021).

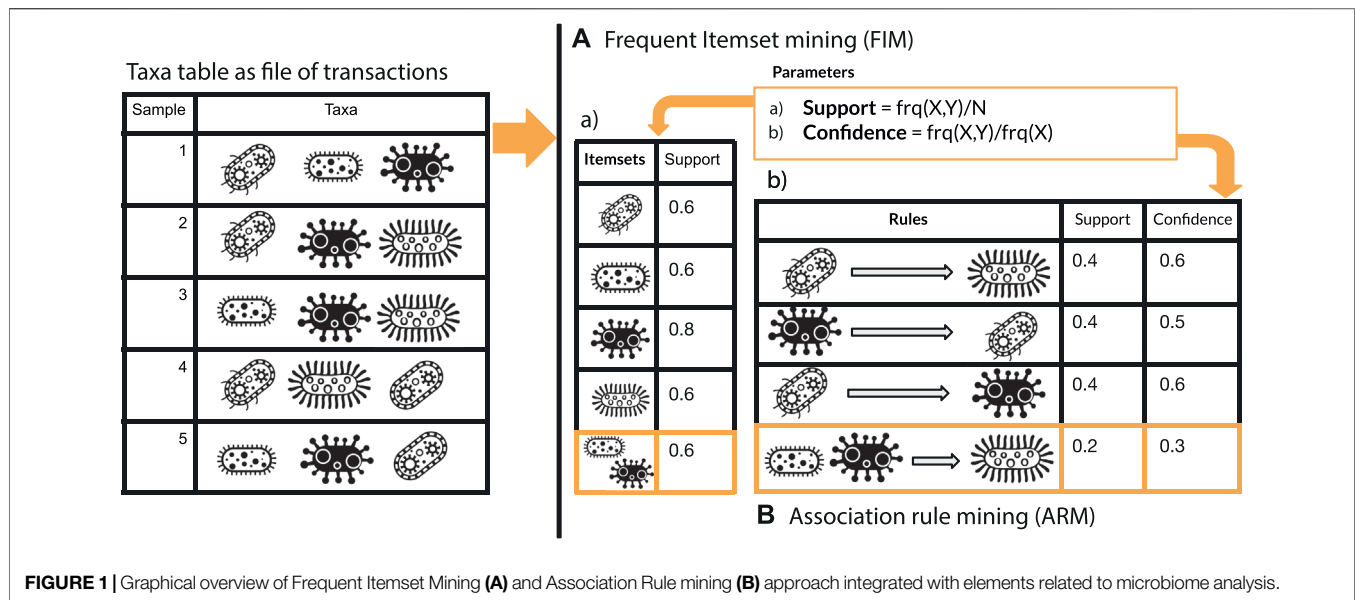
Recently, association rule mining (ARM) emerged as a promising technique to study microbiome patterns (Naulaerts et al., 2015; Tandon et al., 2016). Specifically, Tandon et al. (2016) have demonstrated the potentials of this technique on two microbiome datasets, in particular the HMP dataset (Turnbaugh et al., 2007) and two prebiotic studies (Kato et al., 2014; Xiao et al., 2014). From the classic application on market basket problems (Agrawal et al., 1993), association rule mining started to be applied to answer a wide range of biological questions. From annotation tasks (Manda et al., 2012; Manda et al., 2013; Manda, 2020) to protein interaction networks (Koyuturk et al., 2006), ARM was applied to a wide range of research fields, including genetics (Carmona-Saez et al., 2006; Alves et al., 2010; Karpinets et al., 2012; Ong et al., 2020), molecular biology (Agapito et al., 2015; Boutorh and Guessoum, 2016; Naulaerts et al., 2016), and biochemical disciplines (Yoon and Lee, 2011; Zhou et al., 2013; Naulaerts et al., 2016). Noticeably, the expression ‘association rule mining’ comprehends two main phases: 1) frequent itemset mining, the extraction of patterns intended as elements often co-occur together in a dataset (Agrawal et al., 1993), and 2) rule

calculation, to identify strong association between patterns previously extracted (Agrawal et al., 1993).

Despite the apparent simplicity of use, large datasets can produce high numbers of patterns, making their extraction difficult (Agrawal et al., 1993; Han et al., 2004; Karpinets et al., 2012; Naulaerts et al., 2015). Beside several algorithms have been developed to better capture reliable patterns, as for example Eclat (Agrawal et al., 1996), FP-Growth (Han et al., 2004) or Apriori (Agrawal et al., 1993), avoiding uninformative or spurious information is still a current issue (Naulaerts et al., 2015). Interesting measures such as support (frequency of a pattern) or pattern length are pivotal to control the generation and the evaluation of patterns discovered (Agrawal et al., 1993; Karpinets et al., 2012; Naulaerts et al., 2015). Still, a few issues exist in setting these parameters (Naulaerts et al., 2015). Considering the support, setting a low value leads to a high amount of patterns, difficult to explore and visualize. At the same time, setting a high support value can be detrimental for finding rare but informative patterns. Over and above, researchers try to identify metrics that can be used to pinpoint patterns of interest (and so called “interest measures”). In detail, several metrics have been implemented (Tan et al., 2002; Omiecinski, 2003; Franceschini et al., 2012; Tang et al., 2012), as for example lift or maximal entropy (Tatti and Mampaey, 2010; Hussein et al., 2015). Nevertheless, extracting effective information is not an easy task as the definition of interestingness is strictly associated with the biological question and the research field under study (Koyuturk et al., 2006; Karpinets et al., 2012; Naulaerts et al., 2015). Considering the rule calculation phase, issues regarding the evaluation of reliable rules remain (Karpinets et al., 2012; Naulaerts et al., 2015). In general, taking into account previous works, the most widely used parameters to evaluate both patterns and rules are support and confidence, where confidence is a measure that describes the strength of the association between the two elements of the rule (Naulaerts et al., 2015).

Recently, different works related to pattern mining applied to microbiome studies were published, such as MITRE (Bogart et al., 2019), MANIEA framework (Liu et al., 2021) and the work of Tandon et al. (2016). Nevertheless, as also highlighted by the work of Faust (2021), applying such an algorithm still has its limitations and, despite the efforts of recent works, guidelines for microbiome data applications have not been completely defined (Naulaerts et al., 2015; Faust, 2021). Different libraries have been implemented, such as pyfim (Muino and Borgelt, 2014), mlxtend (Raschka, 2018) and arules (Hahsler et al., 2011). A few frameworks have been recently developed and applied on real case studies (Tandon et al., 2016; Liu et al., 2021). However, tests to establish specific best practices for 16S rRNA metabarcoding data do not exist.

Apart from the availability of tools, the application of pattern mining to study microbiome patterns must consider the intrinsic biological aspect of microbiome data (Balint et al., 2016; Gloor et al., 2017). Beside the issues related to species abundances that should be filtered to obtain a solid input dataset, also metadata composition and taxonomy level should be considered. Further, microbiome matrices can be large and complex: composed of thousands of taxa and hundreds of samples (Faust, 2021;



Ghannam and Techtman, 2021), microbiome data can affect pattern mining approaches, sometimes obliging to set high but improper interest measures. This last point is crucial if we consider that 16S rRNA metabarcoding data can describe putative ecological properties and sparse microbial associations (Faust, 2021).

Given these premises, our work wants to shed light on the strengths and weaknesses of pattern mining strategy into the study of microbial patterns, in particular from 16S rRNA microbiome datasets. In detail, we show pitfalls of ARM applied on real case studies, highlighting issues related to the type of input and the use of metadata. Then, we identify the key steps that must be considered to apply ARM consciously on 16S rRNA microbiome data. Moreover, to facilitate the integration of ARM technique into microbiome pipeline, we developed microFIM (microbial Frequent Itemset Mining), a versatile user-friendly and open source Python tool that promotes the use of ARM integrating common microbiome practices, such as taxa tables and distance matrix visualizations. Besides the conventional parameters, microFIM implements interest measures to remove spurious information. Moreover, it merges the results of ARM analysis with the typical microbiome outputs, aiming at creating a bridge between microbial ecology research and ARM technique.

2 MATERIALS AND METHODS

This section comprehends two main paragraphs: 1) description of microFIM (microbial Frequent Itemset Mining) tool to promote microbiome pattern exploration with two simulated dataset and 2) microFIM analysis on real case microbiome datasets to highlight ARM potentials and caveats. microFIM was developed on the basis of Frequent Itemset Mining (Naulaerts et al., 2015), in which patterns of elements that co-occur can be extracted from a transactional dataset, typically (Naulaerts et al.,

2015). A pattern (or itemset) is called frequent if its support value within the dataset is greater than a given minimal support threshold. For an overview of the method and its translation in terms of bacterial composition instead of elements, please see **Figure 1**. A complete description of the approach with formalized expression can be found in the works of Tan et al., 2002 (Chapter 6), Goethals, 2005, and Naulaerts et al. (2015).

2.1 microFIM Implementation

To promote and integrate the use of ARM in microbiome studies, we developed microFIM (microbial Frequent Itemset Mining), a versatile open-source user-friendly tool implemented in Python (v. > 3; <https://github.com/qLSLab/microFIM>).

microFIM receives as input the taxa table and the metadata file used during the microbiome bioinformatic analysis. In particular, a taxa table is composed of rows and columns representing the taxa and their abundances for each sample. It derives from the conversion of the BIOM file into a CSV or TSV file (<https://biom-format.org/>). In general, considering the well-established QIIME2 microbiome platform (<https://qiime2.org/>; Bolyen et al., 2018), complete frameworks and scripts to analyse and obtain taxa tables are implemented.

To promote the usage to a wider group of researchers, the tool can be used both *via* Python functions and running the pre-settled scripts, which allow interactivity through the command-line, avoiding coding implementations. To favor easy integration in Python scripting and future implementation of additional functions and metrics, Python functions were divided into thematic sections. microFIM is composed by six main steps: 1) filtering taxa table with metadata, 2) converting taxa table into a transactional database to be read by ARM algorithms, 3) extract microbiome patterns, 4) calculate additional interest measures to evaluate the patterns extracted, 5) create the pattern table (a taxa table improved with patterns, presence-absence information among samples and interest measures) and 6) visualization of results.

Template files are provided to run microFIM scripts. Considering interest measures, we integrated support, pattern length and all-confidence metrics, which generates “hyperclique patterns” (Agrawal et al., 1993; Tan et al., 2002; Omiecinski, 2003; Xiong et al., 2006). Considering a pattern “X” composed of different items, all-confidence is calculated as the ratio between the support of “X” and the highest support retrieved from the elements of the pattern “X.” For example, a pattern X is composed of three elements that, considering the entire dataset, have the following support threshold: 0.3, 0.6 and 0.8. Overall, the pattern X has a support of 0.3. All-confidence will be calculated as the ratio between the support of X—0.3—and the higher support within X—0.8, resulting in 0.37. All-confidence, in this way, is defined as the smallest confidence of all rules which can be produced from a pattern, i.e., all rules produced from a pattern will have a confidence greater or equal to its all-confidence value (Tan et al., 2002; Omiecinski, 2003). In detail, confidence is an indication of how often a rule has been found to be true, so it is considered as a measure of rule reliability (Hornik et al., 2005; Hahsler et al., 2011; Naulaerts et al., 2015).

In order to show the usage and the potentials of microFIM, we tested the tool on simulated matrices (available in **Supplementary Tables S1, S2**) and on real case studies. In particular, the cases selected are: 1) the ECAM dataset (Bokulich et al., 2016), 2) the vaginal microbiome dataset of Ravel et al. (2011) and 3) the Montassier dataset (Montassier et al., 2016). Details about the application of microFIM on real case studies are described in the next sections. Parameters used to run microFIM on simulated matrices are the following: 0.3 as minimum support threshold, a minimum of two elements and a maximum of 10 to extract patterns.

In the Results section, a complete scheme of the tool is provided. microFIM is mainly based on four Python libraries: *fim* (Muino and Borgelt, 2014), *Pandas* (McKinney, 2010; Reback et al., 2020), *Numpy* (Harris et al., 2020), and *plotly* (<https://plotly.com/>). It is available as a conda environment (<https://docs.anaconda.com/AnacondaSoftwareDistribution/2020>) and all the details about tutorials and installation are available in our Github repository (<https://github.com/qLSLab/microFIM>). Python notebooks and an example of microFIM usage *via* scripting are also reported in the repository. In general, beside the focus of this work, microFIM may potentially be used for a wide range of applications. As the primary resource input consists in a matrix describing the presence-absence of an element (rows) in a dataset (columns, representing samples), fields of study in which it can be applied may be various, also merely consider the analysis of OTU (Operational Taxonomic Unit) or ESV (Exact Sequence Variants) instead of taxa (Schloss and Westcott, 2011; Callahan et al., 2017) of 16S rRNA metabarcoding data.

2.2 Real Case Studies Analysis

To show the caveats and potentials of association rule mining, we used microFIM on three real case studies: the ECAM dataset (Early Childhood Antibiotics and the Microbiome; Bokulich et al., 2016), the vaginal microbiome case study of Ravel et al. (2011) and Montassier case study (Montassier et al., 2016). Different input types were selected based on taxonomy level and metadata composition. In detail, the ECAM dataset collects a total of 875

samples, describing the gut microbiome of the first 2 years of life of 43 infants. Presence-absence tables were created taking account of the taxonomic rank. In particular, we used: 1) the taxa table obtained directly from QIIME2 datasets (Bolyen et al., 2018) in which only taxa assigned to genus level, with a relative abundance > 0.1% in more than 15% of samples, are considered (Input 1—data are available in **Supplementary Table S3**); 2) family table obtained from collapsing the previous Input 1 *via* QIIME2 plugins (<https://github.com/qiime2/q2-taxa>; Input 2—**Supplementary Table S4**); 3) a taxa table consisting only of taxa with complete taxonomy at the genus level (Input 3—**Supplementary Table S5**). Metadata as type of delivery and antibiotic exposition were considered to evaluate patterns extraction.

Considering the vaginal microbiome dataset (Ravel et al., 2011), we obtained from MLRepo repository (Vangay et al., 2019) the taxa table obtained via the MLRepo pipeline (Vangay et al., 2019). The dataset collects 388 samples, investigating the vaginal microbiome of 396 asymptomatic North American women. Additional presence-absence tables were created taking account of the taxonomic rank, in particular from the original dataset obtained from MLRepo, also family and genus levels were considered. Low and high nugent score values (a scoring system for vaginal swabs to diagnose bacterial vaginosis) were considered for the evaluation regarding metadata filtering.

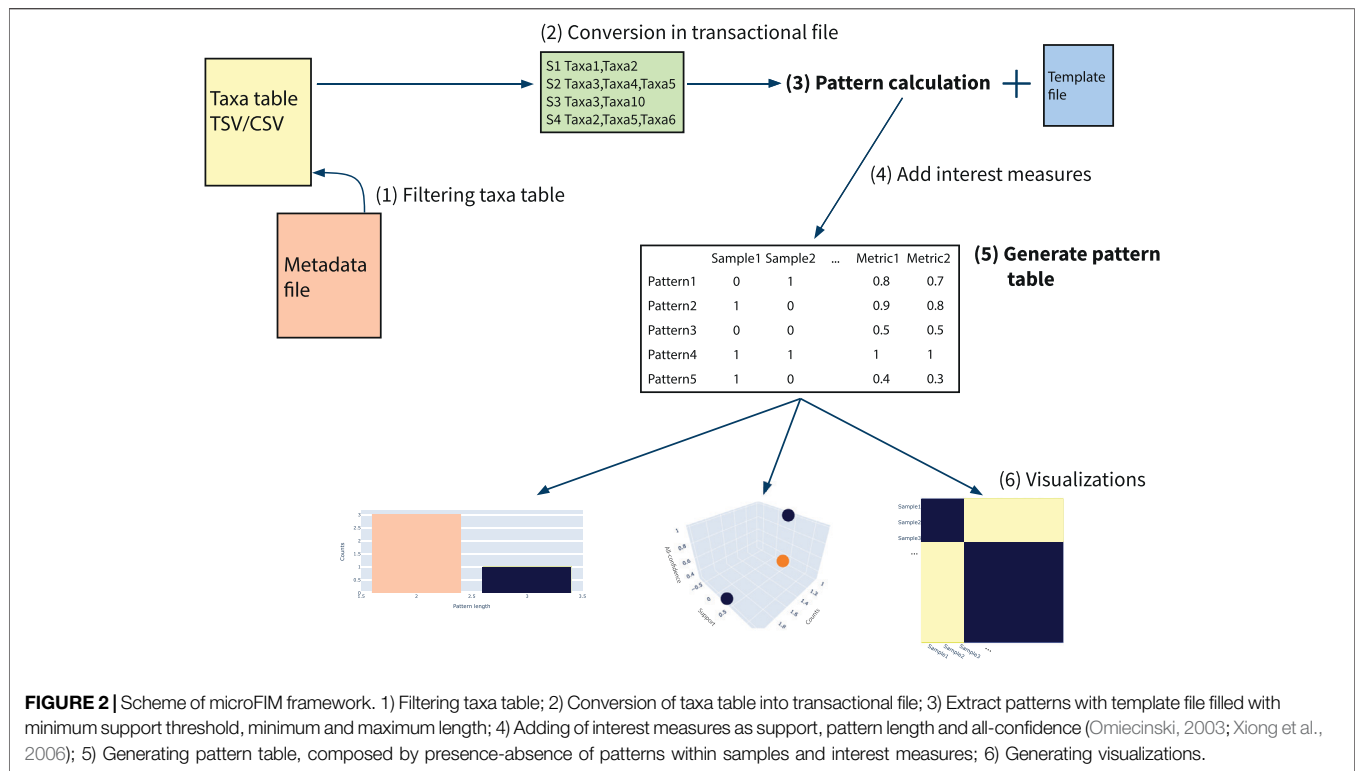
Finally, the dataset of Montassier et al. (2016) was included. The dataset collects 28 samples from patients with non-Hodgkin lymphoma undergoing allogeneic hematopoietic stem cell transplantation (HSCT) in order to identify microbes that predict the risk of BSI (bloodstream infection). OTU table and taxa table obtained with MLRepo pipeline were selected (Vangay et al., 2019).

For the ECAM and Ravel et al. (2011) datasets, minimum support threshold of 0.2, minimum length of 3 and a maximum length of 15 elements were used. Montassier et al. (2016) datasets were analysed considering a minimum support of 0.9, a minimum length of 5 and a maximum length of 10. After pattern extraction, interest measures as support, pattern length and all-confidence were calculated (Tan et al., 2002; Omiecinski, 2003; Xiong et al., 2006). Distributions of number of patterns, length and support were evaluated considering both ARM analysis and interest measures filtering. A minimum of 0.5 and 0.8 of all-confidence were used to evaluate hypercliques patterns (Tan et al., 2002; Omiecinski, 2003; Xiong et al., 2006). Considering metadata filtering, pattern extraction was performed with the previous settings. A minimum of 0.8 of all-confidence was used to evaluate hypercliques patterns (Tan et al., 2002; Omiecinski, 2003; Xiong et al., 2006). Visualizations were created with *plotly* and *pandas* Python libraries. Both datasets, results and metadata files are available in **Supplementary Material**.

3 RESULTS

3.1 microFIM Tool: Extending Association Rule Mining to Microbiome Pattern Analysis

Association rule mining demonstrates its useful properties in different contexts (Naulaerts et al., 2015; Tandon et al., 2016). To



promote the use of ARM in the microbial community field, we implemented microFIM, a versatile open-source project developed in Python and freely available at <https://github.com/qLSLab/microFIM>.

In this section, we explain the framework of usage, the main steps of pattern extraction and filtering and insights of visualizations available. In addition, two main examples are reported, in order to show the workflow of the tool. In **Figure 2** a scheme of microFIM framework is reported. In particular, microbiome data (taxa table) can be filtered (step 1) and then converted into a transactional dataset (step 2), in order to be read as input by association rule mining algorithm. Subsequently, patterns can be generated setting parameters via a template file to be filled (tutorials and templates are available at <https://github.com/qLSLab/microFIM>) (step 3). In detail, minimum support threshold, minimum and maximum length of patterns must be specified. Pattern extraction was implemented via pyfim library (Muino and Borgelt, 2014). At this stage, the default algorithm used is Eclat (Muino and Borgelt, 2014), but other algorithms are available within the pyfim library (Apriori or FP-Growth; Muino and Borgelt, 2014). The set of interest measures initially calculated are “support” and “pattern length” (which describes the number of elements belonging to a pattern). Further, other interest measures are added (step 4) and can be used to filter patterns. In microFIM implementation, all-confidence interest measure was included, in order to help remove spurious information (Tan et al., 2002; Omiecinski, 2003; Xiong et al., 2006). As described in **Section 2**, all-confidence can be used to set the smallest confidence of all rules that can be produced from a pattern, i.e., all rules produced from the pattern will have a confidence greater or equal to its all-confidence value, creating the basis for rule

reliability exploration at the pattern level (Tan et al., 2002; Hornik et al., 2005; Omiecinski, 2003; Xiong et al., 2006; Hahsler et al., 2011; Naulaerts et al., 2015).

The main result of this step is the creation of the pattern table (step 5). Conceptually similar to the microbiome taxa table, the pattern table described the presence of a pattern for each sample, integrating the interest measures previously calculated (step 4). microFIM visualizations comprehend distributions of patterns considering support, length and interest measure values. To describe the relationships between samples considering patterns found, a Jaccard matrix can be also obtained and visualized (step 6).

To better show the potentials of microFIM, we included a demonstrative analysis of both simulated data and data belonging to real case studies (see the next **Section 3**). In particular, as also described in the **Section 2**, simulated data are composed of two main matrices with a dimension of 10 samples and 5 taxa. In **Figures 3A,B** a graphical representation of the simulated matrices is shown. Through microFIM, ARM analysis was performed. The final output of the analysis is the pattern table, represented in **Figures 3C,D** and available in **Supplementary Tables S6, S7**, respectively. The pattern table integrates the interest measures of length, support and all-confidence and, as it is a dataframe, patterns can be filtered and further visualized with Python libraries or other data analysis tools easily. In addition, results of the pattern table can be visualized with microFIM through the following plots: scatter plot, bar chart and heatmap. In **Figures 3E,F**, heatmaps built on Jaccard distance results are shown.

In detail, Dataset 1 (**Figure 3A**; **Supplementary Table S1**) is a full-presence dataset. This means that ARM can potentially

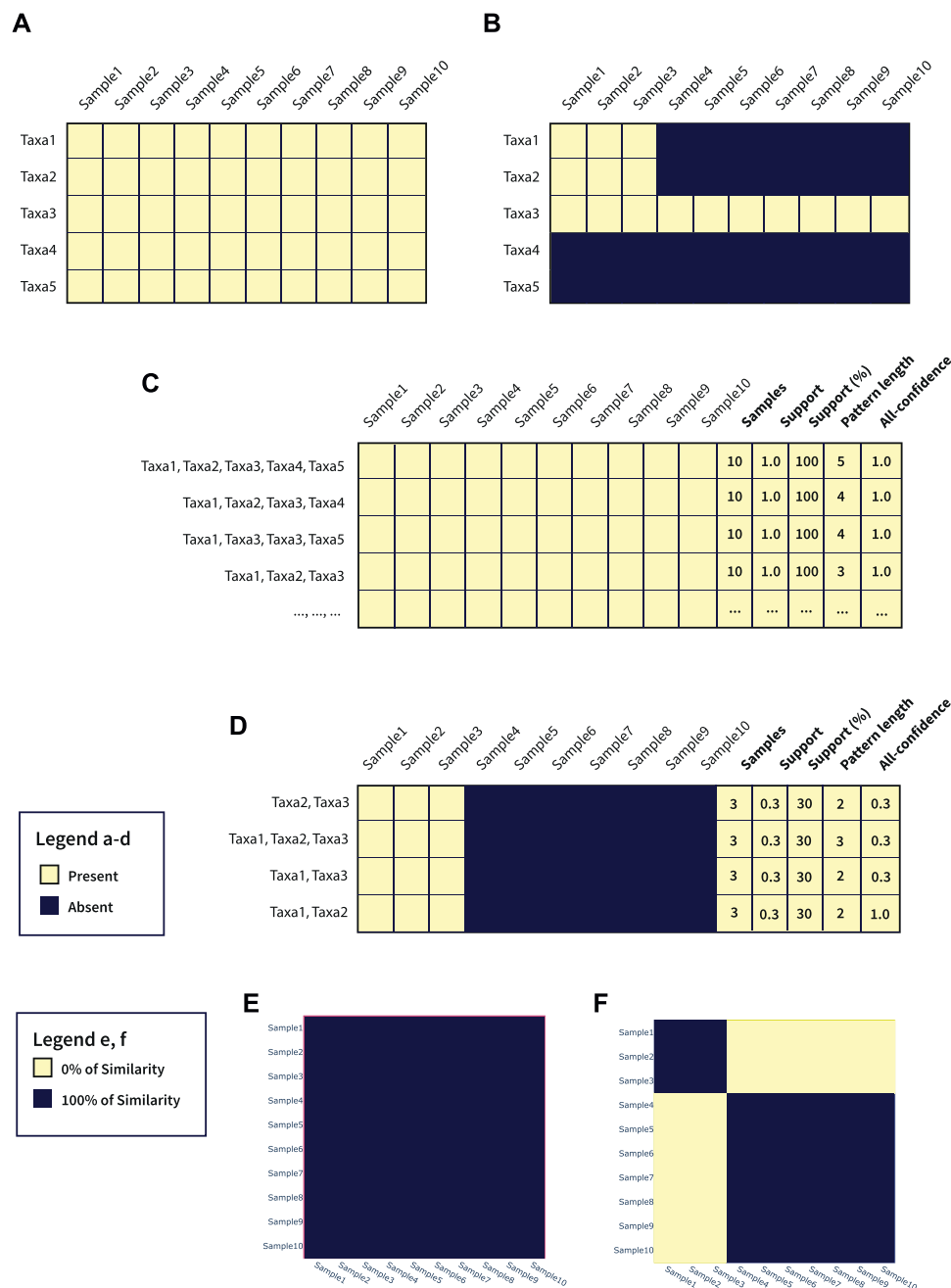


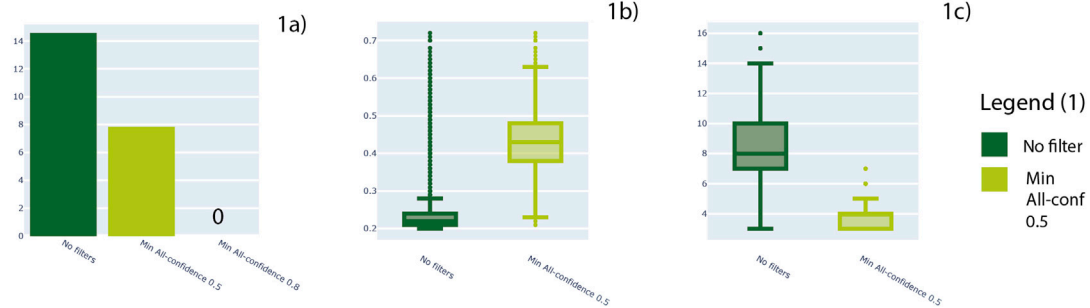
FIGURE 3 | (A) Graphical representation of Table 1; **(B)** Graphical representation of Table 2; **(C)** Pattern table generated from Table 1; **(D)** Pattern table generated from Table 2; **(E)** Jaccard heatmap plot of Table 1; **(F)** Jaccard heatmap plot of Table 2.

generate all the combinations of patterns from a length of 1 to a length of 5. All patterns will have a 1.0 of support and a 1.0 of all-confidence, as they are all associated with each other. In this case, considering only the pattern composed by Taxa1, Taxa2, Taxa3, Taxa4, and Taxa5, with a length equal to 5 and a support equal to 1.0, can be sufficient to resume the information within the dataset. In addition, these settings can be adjusted directly by running the algorithm, avoiding the creation of uninformative patterns and reducing calculation time. In **Figure 3E**, Jaccard

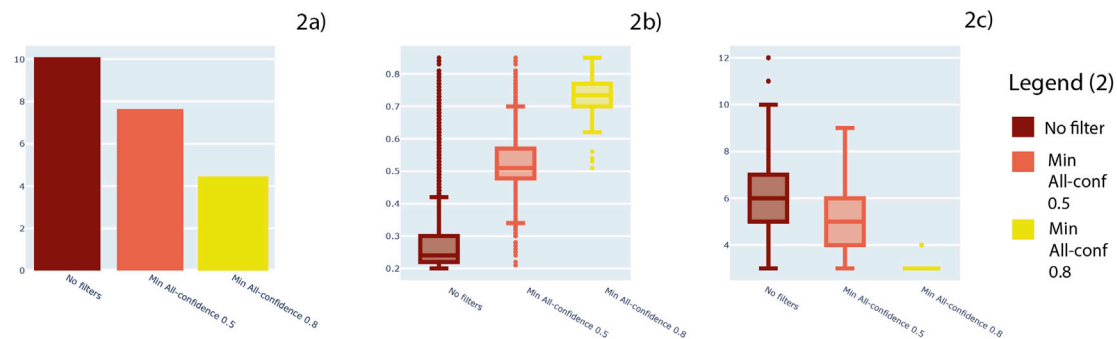
heatmap shows also the 100% similarity between Dataset 1 samples. The complete pattern list obtained by Dataset 1 is available in **Supplementary Table S6**.

Considering Dataset 2 (**Figure 3B**; **Supplementary Table S2**), instead, a different composition can be observed. In particular, Taxa1, Taxa2 and Taxa3 co-occur in samples 1, 2, and 3. In addition, Taxa3 is present in all the samples (**Figure 3B**). As we ran an ARM analysis considering a minimum length of 2, the pattern composed by only Taxa3 was not detected. However, the

ECAM genus table (Input 1)



ECAM family table (Input 2)



ECAM genus table (Input 3)

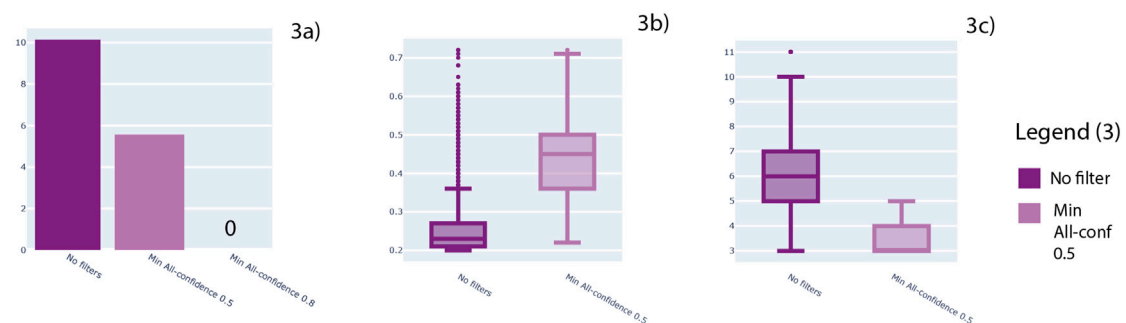


FIGURE 4 | For Input 1, 2 and 3, here number of patterns obtained (1a, 2a, 3a), distribution of support values (1b, 2b, 3b) and distribution of pattern lengths (1c, 2c, 3c) are shown. In particular, three levels of analysis are shown: no filters applied to patterns, a minimum all-confidence of 0.5 and a minimum all-confidence of 0.8.

pattern built by Taxa1, Taxa2 and Taxa3 was detected, with a pattern length of 3 and a support of 0.3. Focus the attention on Taxa1-Taxa2 pattern, the value of all-confidence is equal to 1.0, meaning that there is a strong association between them and the rules generated from this pattern will have a minimum confidence of 1.0. Details about patterns extracted from Dataset 2 are available in **Supplementary Table S7**.

3.2 microFIM Applied on Real Case Studies

Association rule mining is a data mining technique widely used in very different research fields and applications. This chapter is

dedicated to the use of ARM, in particular the pattern mining step, on real microbiome case studies. In detail, three case studies was chosen to demonstrate the potentials of ARM and microFIM: the ECAM dataset (Bokulich et al., 2016), the vaginal microbiome case study of Ravel et al. (2011) and the Montassier case study (Montassier et al., 2016) (see **Section 2** for details). Considering the potential of ARM to reconstruct patterns, we focused the analysis on three main aspects: the type of input used, the filter of patterns whose elements are highly related to each other (also called hyperclique patterns; Xiong et al., 2006) and the use of metadata to filter and apply ARM.

To evaluate how ARM can be used on microbiome data, different types of inputs were considered. In particular, for the ECAM case study, we used: 1) the ECAM taxa table obtained directly from QIIME2 datasets (Bolyen et al., 2018) in which only taxa assigned to genus level, with a relative abundance > 0.1% in more than 15% of samples, are considered (Input 1—data are available in **Supplementary File S3**); 2) family table obtained from collapsing the original one *via* QIIME2 plugins (Input 2—**Supplementary File S4**); 3) a taxa table consisting only of taxa with complete taxonomy at the genus level (Input 3—**Supplementary File S5**).

Minimum support thresholds of 0.2, minimum length of 3 and maximum length of 15 were considered. In **Figure 4** we show the results about the number of patterns retrieved considering three levels of analysis: output after the analysis previously described, patterns filtered with a minimum all-confidence of 0.5 and patterns filtered with a minimum all-confidence of 0.8. In **Figure 4**, for each filter, the distribution of support values and pattern length are provided.

In detail, Input 1 (**Supplementary File S3**) generated a total of 1,844,696 patterns. The mean support achieved by the patterns generated is 0.3 and a median of 0.2, with a minimum value of 0.2 and maximum value of 0.7. Regarding the pattern length, the mean value is 8.45, while the median is 8, with a minimum value of 3 and maximum value of 16.

Family table (Input 2—**Supplementary File S5**) generated a total of 23,997 patterns. The mean support achieved by the patterns generated is 0.28 and a median of 0.24, with a minimum value of 0.2 and maximum value of 0.85. Regarding the pattern length, the mean value is 6.38, while the median is 6, with a minimum value of 3 and maximum value of 12.

Regarding genus table (Input 3—**Supplementary File S6**), ARM analysis generated a total of 25,250 patterns. The mean support achieved by the patterns generated is 0.25 and a median of 0.23, with a minimum value of 0.2 and maximum value of 0.85. Regarding the pattern length, the mean value is 6.14, while the median is 6, with a minimum value of 3 and maximum value of 11. All the results are available in **Supplementary Tables S6–S8**, respectively, and can be visualized in **Figure 4**.

In order to consider the putative informative patterns, a framework involving hypercliques patterns (Xiong et al., 2006) was applied. In particular, the all-confidence metric was considered at 0.5 and 0.8 thresholds for all the datasets analysed (Inputs 1–3).

Regarding the Input 1 (**Supplementary File S3**), a total of 2,213 patterns were extracted considering an all-confidence of 0.5, while no patterns were obtained with 0.8 threshold. First all-confidence threshold resulted in patterns with a mean and a median support value was 0.43, with a minimum value of 0.21 and a maximum of 0.72. Pattern length consisted in a mean of 3.9, a median length of 4, with minimum and maximum of 3 and 7, respectively.

Regarding the Input 2 (**Supplementary File S4**), a total of 2,081 patterns were extracted considering an all-confidence of 0.5. A mean support of 0.53 and a median support was 0.51 were observed, with a minimum value of 0.21 and a maximum of 0.85. Pattern length consisted of a mean of 4.98, a median length of 5,

with minimum and maximum of 3 and 9, respectively. A total of 78 patterns were extracted considering an all-confidence of 0.8. A mean support of 0.72 and a median support was 0.73 were observed, with a minimum value of 0.51 and a maximum of 0.85. Pattern length consisted of a mean of 3.23, a median length of 3, with minimum and maximum of 3 and 4, respectively.

Regarding the Input 3 (**Supplementary File S5**), instead, a total of 25,250 patterns were extracted considering an all-confidence of 0.5, while no patterns were obtained with 0.8 threshold. First all-confidence threshold resulted in patterns with a mean of 0.25 and a median support value of 0.23, with a minimum value of 0.2 and a maximum of 0.72. Pattern length consisted in a mean of 6.14, a median length of 6, with minimum and maximum of 3 and 11, respectively.

For demonstrative purposes, a Jaccard heatmap considering samples belonging to the first sampling date of the ECAM dataset of the Input 3 table (**Supplementary Table S5**) was generated, in order to show a potential use of Jaccard distance on pattern analysis (available in **Supplementary Figure S11**). In general, results are summarized in **Figure 4** and tables are available in **Supplementary Tables S8–S10**, respectively.

Overall, Input 1 obtained the highest number of patterns, achieving 1,844,696 patterns. The support distribution has a great range of values for all the three datasets, from 0.2 to almost 0.8. Also length achieved a wide range of values, considering patterns from 3 elements length to almost 16. In general, a great reduction in the number of patterns was observed considering the all-confidence filtering (**Figure 4**—sections 1a, 2a and 3a). In parallel, this filter resulted in higher support values (**Figure 4**—sections 1b, 2b and 3b) and lower pattern length (**Figure 4**—sections 1c, 2c and 3c).

Metadata filtering was applied to the genus ECAM dataset, considering two category types: antibiotic administration and type of delivery. The complete results of the pattern analysis are available in **Supplementary Table S12**. Overall, a total of 141,480 patterns were obtained from the data belonging antibiotic administration, while the opposite obtained a total of 8,223. Vaginal delivery resulted in a total of 45,412 patterns, while cesarean delivery samples resulted in 10,288. Also in this case, the usage of all-confidence filtering drastically reduced the number of explorable patterns, achieving the following results: 2 and 1 patterns for antibiotic administration and vaginal delivery, respectively, and 0 patterns for the opposites.

microFIM was also applied to other two real case studies: vaginal microbiome obtained by the work of Ravel et al. (2011) and the dataset of Montassier case study (Montassier et al., 2016). Considering the first one, different input types and metadata filtering were used: in particular, the dataset was obtained from the MLRepo collection (Vangay et al., 2019). Then, family level and genus level dataset were obtained. Dataset can be identified as Input 4 (dataset available in MLRepo; Vangay et al., 2019—**Supplementary File S15A**), Input 5 (dataset at the family level—**Supplementary File S15B**) and Input 6 (dataset at the genus level—**Supplementary File S15C**). As for the ECAM analysis, results are presented considering the three main input types and the number of distribution of patterns are evaluated as the previous scheme.

In particular, Input 4 (**Supplementary File S15A**) generated a total of 83 patterns. The mean support achieved by the patterns generated is 0.2 and a median of 0.2, with a minimum value of 0.2 and maximum value of 0.5. Regarding the pattern length, the mean value is 3.1, while the median is 3, with a minimum value of 3 and maximum value of 4. Family table (Input 5—**Supplementary File S15B**) generated a total of 226 patterns. The mean support achieved by the patterns generated is 0.25 and a median of 0.23, with a minimum value of 0.2 and maximum value of 0.55. Regarding the pattern length, the mean value is 3.68, while the median is 4, with a minimum value of 3 and maximum value of 6. Regarding genus table (Input 6—**Supplementary File S15C**), ARM analysis generated a total of 225 patterns. The mean support achieved by the patterns generated is 0.25 and a median of 0.24, with a minimum value of 0.2 and maximum value of 0.46. Regarding the pattern length, the mean value is 3.77, while the median is 4, with a minimum value of 3 and maximum value of 6. All the results are available in **Supplementary Tables S15D–F**, respectively, and can be consulted in **Supplementary Table S14**.

Minimum all-confidence of 0.5 and 0.8 were considered to evaluate hypercliques patterns. Regarding the Input 4 (**Supplementary File S15A**), 16 patterns were extracted considering an all-confidence of 0.5, while no patterns were obtained with 0.8 threshold. First all-confidence threshold resulted in patterns with a mean of 0.23 and a median support value was 0.21, with a minimum value of 0.2 and a maximum of 0.48. Pattern length consisted in a mean of 3.06, a median length of 3, with minimum and maximum of 3 and 4, respectively.

Input 5 (**Supplementary File S15B**) obtained two patterns, considering an all-confidence of 0.5, while no patterns were obtained with 0.8 threshold. The 0.5 all-confidence threshold resulted in patterns with 0.46 and 0.55 support values. Both patterns have a length of 3.

Regarding the Input 6 (**Supplementary File S15C**), 15 patterns were extracted considering an all-confidence of 0.5, while no patterns were obtained with 0.8 threshold. First all-confidence threshold resulted in patterns with a mean and a median support value was 0.3, with a minimum value of 0.25 and a maximum of 0.38. Pattern length consisted in a mean of 3.13, a median length of 3, with minimum and maximum of 3 and 4, respectively.

Overall, the support distribution has a low range of values for all the three input files, from 0.2 to almost 0.5. Length is around 3 elements per pattern. In general, also in this case a great reduction in the number of patterns was observed considering the all-confidence filtering (**Supplementary Table S14**).

Metadata filtering was applied to the dataset, considering the nugent category, low and high levels. The complete results of the pattern analysis are available in **Supplementary Table S14**. Overall, a total of 15,836 patterns were obtained from the data belonging to high nugent score value, while the opposite obtained a total of 21. The usage of all-confidence filtering drastically reduced the number of explorable patterns, obtaining 16 patterns for high nugent score value.

Strengths	Opportunities
<ul style="list-style-type: none"> Allow exploration of high dimensional datasets Versatile Method established in several fields 	<ul style="list-style-type: none"> Explore complex microbial patterns (composed by group of taxa) Applicable to different microbial contexts Stimulate new microbial association approaches
<ul style="list-style-type: none"> Depends on input type Depends on the biological question Need of visualization strategies for high dimensional data 	<ul style="list-style-type: none"> Computational efforts Requires additional efforts in setting the parameters Hard to be tested on real case studies
Weaknesses	Threats

FIGURE 5 | Overview of the main strengths, weaknesses, opportunities and threats (SWOT analysis) related to the use of frequent itemset mining as a tool for microbiome pattern analysis.

Finally, Montassier dataset (Montassier et al., 2016) was tested considering the OTU table and taxa table obtained from MLRepo pipeline (Vangay et al., 2019). A minimum support threshold of 0.9 was considered, with a minimum length of 5 and a maximum length of 10. A total of 446 patterns were obtained considering the taxa table, while 9 patterns were obtained considering the OTU table.

Distributions of pattern and length are similar between the two input files. In particular, a mean support of 0.93 and a mean length of 5.1 (5–6) were detected.

4 DISCUSSION

Pattern mining strategies are now newfangled solutions for disclosure of microbial patterns (Tandon et al., 2016; Liu et al., 2021). However, besides the power of these techniques, great efforts must be undertaken to extrapolate relevant patterns that can be integrated into biological contexts (Naulaerts et al., 2015; Faust, 2021).

Basically, the strategy consists of two main phases: 1) extraction of patterns (also known as “frequent itemset mining”) and 2) rules calculation. In this work, we focused in particular on the first phase, as great potential can be achieved considering the exploration of patterns at any length and subsequently be filtered to create reliable associations.

In detail, our **Section 4** will touch two main topics: 1) considerations about parameter settings to perform pattern mining strategies in the context of 16S rRNA metabarcoding data and 2) guidelines and future perspectives to support real applications. In order to present an overview of frequent itemset mining as a tool for microbiome pattern analysis, we developed a

SWOT (Strengths, Weaknesses, Opportunities, Threats) analysis (Figure 5).

4.1 Run Association Rule Mining Could Not Be Enough Without Care in Setting Parameters

As described above, pattern mining strategies can be powerful to get insights from large and complex datasets (Naulaerts et al., 2015). However, pattern analysis may have limitations (Faust, 2021). In this work, we provide ARM analysis on both simulated and real datasets and propose microFIM (<https://github.com/qLSLab/microFIM>), a Python tool specifically suited for microbiome pattern analysis. Our results will consider the pattern composition obtained through our framework (Section 2) without considering their biological implications, as it is beyond the scope of this work.

Considering the application of ARM on simulated datasets, we showed that initial settings can reduce the amount of information retrievable, both considering interest measures as support or length and all-confidence metric.

Regarding the application on the real case studies, a few considerations can be made. First of all, the type of input can change the reliability of results: different numbers of patterns have been generated considering different input types. In particular, both considering aspects related to data visualization and interpretation, the taxonomy level of investigation must be considered.

A second point that arises is the minimum support threshold to choose. The choice can be both related to biological questions, as for example which is the minimum number of samples to retain a pattern interesting, but also on technicalities. In detail, exploring all the potential patterns cannot be reliable and useful, as the number of patterns can be very high, related also to great computational efforts and visualization issues (Naulaerts et al., 2015). For this reason, we started using a support of 0.2, that means that only the taxa that co-occur in at least the 20% of samples were considered (up to 175 of 875 for the ECAM dataset and up to 77 of 388 for the Ravel case study). However, this is a case-specific threshold as no guidelines exist to set a correct support threshold in this research field. The wrong value can potentially hide information and, at the same time, create spurious patterns. In addition, it can generate misleading results without taking into account the Simpson's paradox (Tan et al., 2002), a phenomenon in which a pattern appears frequently but disappears or drastically changes when the data are combined differently, as for example considering only a set of samples (Tan et al., 2002).

Nevertheless, once patterns are generated, filtering steps can be added, in order to both reduce the information and better evaluate specific patterns, with peculiar characteristics. Filters can include the length of patterns or additional interest measures (Agrawal et al., 1993; Karpinets et al., 2012; Naulaerts et al., 2015).

Pattern length, in particular, can be also included before running the analysis, as algorithms take into account a minimum and a maximum value of pattern length, in order to reduce the number of explorable patterns (Agrawal et al., 1993).

However, this choice must be done before exploring the results. Of course, it is possible to reduce the number of patterns after extraction, but computational efforts and running time must be considered (Agrawal et al., 1993; Naulaerts et al., 2015). Pattern length can also vary based on the research field of application and the biological questions. In the ECAM case study, for example, we observed different median values of pattern length, from minimum values of 3 to maximum of 16, suggesting also different levels of analysis.

However, other metrics can be included to filter patterns (Tan et al., 2002; Omiecinski, 2003; Franceschini et al., 2012; Tang et al., 2012). Usually they are called "interest measures" and are generally used to evaluate a set of peculiar patterns, in order to filter the interesting ones (Tatti and Mampaey, 2010; Hussein et al., 2015; Naulaerts et al., 2015). Also in this case, the biological question can guide how to properly set the filtering step. In this work, we used all-confidence metrics, which generate hyperclique patterns (Omiecinski, 2003; Xiong et al., 2006). The application of this metric helps to find groups of items (in this case species or taxa) where items belonging to the same pattern are highly affiliated with each other and can generate rules with the minimum threshold chosen. Using this approach reduces drastically the number of patterns and, in addition, allows to filter only strong associated groups. In this case, the amount of information was drastically reduced considering the two thresholds of all-confidence considered (0.5 and 0.8). This reduction can promote a manual exploration of results and pave the way for exploring strong associations and putative rules. Clearly, other interest measures can be applied. All-confidence may not be the only interest measures useful for microbiome analysis. Other metrics can be selected to filter patterns, but they must be identified based on specific questions related to the research field of application (Naulaerts et al., 2015).

4.2 Fitting Association Rule Mining for Microbiome Studies: Guidelines to Support Real Applications

Frequent itemset mining and, subsequently, association rule mining, is a pattern mining technique able to explore items that co-occur with a certain frequency, as sets of commercial products that customers buy together in the classic supermarket basket problem (Agrawal et al., 1993; Naulaerts et al., 2015). The flexibility of frequent itemset mining techniques is demonstrated by the wide range of bioinformatics applications, from for example SNPs association studies to annotations and motif association exploration (Carmona-Saez et al., 2006; Koyuturk et al., 2006; Alves et al., 2010; Karpinets et al., 2012; Manda et al., 2012; Manda et al., 2013; Zhou et al., 2013; Agapito et al., 2015; Boutorh and Guessoum, 2016; Naulaerts et al., 2016; Manda, 2020; Ong et al., 2020). It is a powerful instrument to explore patterns from large and complex data sets (Agrawal et al., 1993; Karpinets et al., 2012; Naulaerts et al., 2015), providing different algorithms and a wide range of parameters to filter patterns of interest. Besides the most used, as support (frequency of a pattern or a rule in the dataset) or length (the number of species

contained in a pattern), other metrics can be included in the pattern analysis (Naulaerts et al., 2015; Agrawal et al., 1993; Hornik et al., 2005). Beside its potentials, great efforts have to be made to perform pattern mining strategies on microbiome data and obtain reliable and interpretable results, with sound biological implications. As mentioned above, a few points raised from the works done. From threshold choices to input data types, setting pattern analysis is not an easy task. Considering the peculiarities of microbiome data and the flexibility of the technique, here we propose five statements to guide researchers before starting ARM analysis.

4.2.1 Setting the Input Data

This point highlights the importance of the type of pattern to be considered. In the microbial ecology field, a lot of interest probably regards the investigation of species patterns, in order to evaluate community patterns and putative ecological processes. However, this is not straightforward if we consider 16S rRNA metabarcoding data: taxonomy does not always reach a species level and this uncertainty can negatively impact pattern reconstruction. In addition, noise derived from contamination or sequencing biases can be present (Faust and Raes, 2012; Balint et al., 2016; Gloor et al., 2017; Faust, 2021). However, precautions can be taken: removing uncertain taxa or cleaning the table based on abundance thresholds or statistical methods is possible (Faust and Raes, 2012; Balint et al., 2016; Gloor et al., 2017). Different levels of taxonomy can be used as input, as we also demonstrated in the previous sections. Of course, choices must be taken with conscience as they will impact on the final result and therefore the interpretation must be correctly contextualized.

4.2.2 Consider the Use of Metadata

The inclusion or filtering considering metadata information can improve the reliability of the method, both looking for specific patterns linked to metadata and also to better explore the dataset. In this way, we can reduce the information to be explored, lowering the support value, retaining rare or patterns related to specific metadata, and preventing Simpson's paradox issues (Agrawal et al., 1993; Naulaerts et al., 2015).

4.2.3 Individuate What is Interesting for the Specific Case Study

The definition of what is interesting depends on the biological context at issue. No simple guidelines exist, as the application of pattern mining on microbiome data is still in its infancy (Naulaerts et al., 2015). Testing and developing new metrics is an important field of research and can make a difference to track reliable patterns that can be further used for classification tasks or functional analysis. In this work, we applied the all-confidence metric (Omiecinski, 2003; Xiong et al., 2006). However, we believe that other interest measures can be applied and a wide variety of them are available in other tools already developed (Hahsler et al., 2005; Hahsler et al., 2011). In general, this step allows to drastically reduce the number of explorable patterns (Tan et al., 2002; Omiecinski, 2003; Xiong et al., 2006).

Basically, length can be used to clean the information extracted via ARM. As ARM can generate patterns at any length, single

items or only pairs of items can be pruned, in order to find interesting associations composed by 3 or more elements. From a biological point of view, exploring longer microbial patterns can enhance microbial community investigations and pave the way for high-order interactions exploration (Faust, 2021).

4.2.4 Consider Computational Time

As fully described in previous works, data dimensions and density drastically increase time calculation and memory usage (Agrawal et al., 1993; Naulaerts et al., 2015). Reducing input data can make ARM more reliable and faster to be performed (Agrawal et al., 1993; Naulaerts et al., 2015). In addition, beside the common concept of pattern, closed and maximal patterns exist. Both result in a faster extraction, but with a reduction of information (Agrawal et al., 1993; Naulaerts et al., 2015).

Overall, the inclusion of interest measures directly into the ARM framework may favour the development of new faster algorithms, leading the technique directly to the exploration of specific patterns (Omiecinski, 2003; Xiong et al., 2006; Naulaerts et al., 2015).

4.2.5 Tools and Visualization Strategies

To better suit pattern mining for microbiome data applications, tools and visualization techniques are essentials (Naulaerts et al., 2015). In detail, in this work we tried to concept a new pattern mining output combining the common microbiome output with pattern analysis. The pattern table can be an important resource to perform and visualize pattern results in a microbial perspective. In addition, it allows further statistical analysis that is usually performed for microbiome data. Considering the visualization process, we set up different plots to have an overview of pattern distributions and create a Jaccard matrix to show the distance between samples. However, different visualization methods exist, based on tables, matrices and graphs (Naulaerts et al., 2015). Here we cite the R packages *arulesviz*, *FPViz* and *WiFiViz* (Hornik et al., 2005; Hahsler et al., 2011; Naulaerts et al., 2015). Even though these visualizations allow different strategies to explore data, issues related to high dimensional dataset remain and none of them are conceptualized for microbiome analysis. At the same time, collecting human readable information can facilitate data visualization strategies and interpretation (Naulaerts et al., 2015), but of course interesting measures must be considered. Finally, considering practicality of use, several ARM implementations can be utilized (Naulaerts et al., 2015). Moreover, frameworks have been implemented, often accompanied by GUI (Graphical User Interface) or interactivity components (Naulaerts et al., 2015). However, a deepening in the microbiome field has not been established yet.

4.2.6 Evaluation and Benchmarking Strategies

From a computational point of view, the complexity and dynamics of microbial communities leads to difficulties in developing and testing methods to evaluate them. In general, it was demonstrated that microbial co-occurrence analysis may be an extraordinarily promising approach for studying microbiomes (Faust and Raes, 2012). Several works explained how co-

occurrences reveal indications about ecological processes shaping community structure (Lima-Mendez et al., 2015), exploring hub species and potential microorganisms relationships (Berry and Widder, 2014). Further, Ma et al. (2020) showed how global microbial co-occurrence analysis and network reconstruction may be an encouraging strategy to reveal patterns and explore new mechanisms. However, besides these results, transform microbiome data into purposeful biological insights remain challenging, as also demonstrated by different evaluations (Faust and Raes, 2012; Berry and Widder, 2014), and open questions still remain (Faust and Raes, 2012; Layeghifard et al., 2017; Ma et al., 2020; Faust, 2021). The use of ARM on microbiome data models or datasets created *in-silico* will be necessary to disentangle the potentials of ARM in the microbiome research field, also considering the range of microbiome aspects that can be considered (Weiss et al., 2016; Hosoda et al., 2020; Faust, 2021). In particular, tests should examine how the technique is affected by noise signals, both related to sequencing and laboratory protocols (Weiss et al., 2016). In addition, as microbiome data may potentially describe a complex and intricate ecological community, several ecological aspects can be evaluated with ARM, both describing the generation of redundant information and the difficulty associated with extracting patterns due to specific ecological behaviors, as for example competition, exclusion or symbiosis (Faust and Raes, 2012; Weiss et al., 2016; Faust, 2021).

In general, recent advancements in data integration and data reuse strategies may enhance the exploration of microbial patterns from large-scale studies (Jordan and Mitchell, 2015; Ma et al., 2020; Su et al., 2020; Ghannam and Techtman, 2021). Microbiome simulators and *in vitro* studies can be a great instrument for benchmarking works and improve guidelines to apply ARM (Faust, 2021). Beside the potential of ARM on large scale analysis, giving a great overview of data under investigation (Naulaerts et al., 2015), these advancements may contribute to developing tests and benchmarking strategies in order to set ARM for microbial pattern research looking at biological implication, specifically.

Concluding, all the challenges mentioned above can disentangle ARM analysis for microbiome pattern exploration. As the output of the analysis can be extensive and redundant, results should be interpreted with caution. The associations extracted do not necessarily imply causality. Instead, it suggests a strong co-occurrence relationship between species. Causality, on the other hand, requires knowledge about the causal and effect attributes in the data (Tan et al., 2002). There are several approaches to evaluate the robustness of an output. In this first work, pattern length, support and all-confidence were explored and included in the microFIM tool. From a biological perspective, filtering results with these parameters could help to highlight meaningful patterns, but may not be enough. Further, we tried to depict issues that we think must be considered before using an ARM approach for specific biological traits. As there is an interest in research to

exploit data mining techniques, citing for example the works of Srivastava et al., 2019 or Zakrzewski et al., 2017, we also think that suiting ARM for microbiome analysis will be a great resource in the future. Considering the huge amount of data available and produced with the advent of High-Throughput DNA Sequencing (HTS) technologies, an increasing selection of large-scale data science strategies seems to have enormous potential in resolving challenges in microbiome pattern exploration (Jordan and Mitchell, 2015; Kypides et al., 2016). Association rule mining and microFIM tools may have great potential not only with 16S rRNA metabarcoding data, but also in a wide range of applications. As also supported by Naulaerts et al. (2016), ARM analysis is a versatile technique: the integration of files such as taxa tables guarantees the usage also on a wide variety of datasets belonging from different sources, as for example the QIITA platform (<https://qiita.ucsd.edu/>; Gonzales et al., 2018) or the MLrepo (<https://knights-lab.github.io/MLRepo/>; Vangay et al., 2019), but not only. Beside the main focus of this work and microFIM development, very different types of data can be analysed and integrated with ARM framework. From gene associations to merely metabarcoding projects, whose output has the same structure of 16S rRNA taxa table, microFIM may potentially pave the way for multiple usages, creating a bridge with several research fields and applications.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/**Supplementary Material**, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

AG conceived the idea and analyzed the data. AG, BA, SA drafted the manuscript and figures. All authors listed have made a substantial, direct and intellectual contribution to the work, and approved it for publication.

ACKNOWLEDGMENTS

We would like to thank Simone Bosaglia and Alberto Brusati for their constant and effective support. We also thank Dr. Karoline Faust for the precious suggestions. Icon made by Freepik from www.flaticon.com.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fbinf.2021.794547/full#supplementary-material>

REFERENCES

- Agapito, G., Guzzi, P. H., and Cannataro, M. (2015). DMET-miner: Efficient Discovery of Association Rules from Pharmacogenomic Data. *J. Biomed. Inform.* 56, 273–283. doi:10.1016/j.jbi.2015.06.005
- Agrawal, R., Imieliński, T., and Swami, A. (1993). Mining Association Rules between Sets of Items in Large Databases. *SIGMOD Rec.* 22, 207–216. doi:10.1145/170036.170072
- Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., and Verkamo, A. I. (1996). Fast Discovery of Association Rules. *Data Min. Knowl. Discov.* 12 (1), 307–328.
- Alves, R., Rodriguez-Baena, D. S., and Aguilar-Ruiz, J. S. (2010). Gene Association Analysis: a Survey of Frequent Pattern Mining from Gene Expression Data. *Brief. Bioinform.* 11 (2), 210–224. doi:10.1093/bib/bbp042
- Anaconda Software Distribution (2020). *Anaconda Documentation*. Austin, TX, USA: Anaconda Inc. Available at: <https://docs.anaconda.com/>.
- Bálint, M., Bahram, M., Eren, A. M., Faust, K., Fuhrman, J. A., Lindahl, B., et al. (2016). Millions of Reads, Thousands of Taxa: Microbial Community Structure and Associations Analyzed via Marker Genes. *FEMS Microbiol. Rev.* 40 (5), 686–700. doi:10.1093/femsre/fuw017
- Berry, D., and Widder, S. (2014). Deciphering Microbial Interactions and Detecting keystone Species with Co-occurrence Networks. *Front. Microbiol.* 5, 219. doi:10.3389/fmicb.2014.00219
- Bogart, E., Creswell, R., and Gerber, G. K. (2019). MITRE: Inferring Features from Microbiota Time-Series Data Linked to Host Status. *Genome Biol.* 20 (1), 186. doi:10.1186/s13059-019-1788-y
- Bokulich, N. A., Chung, J., Battaglia, T., Henderson, N., Jay, M., Li, H., et al. (2016). Antibiotics, Birth Mode, and Diet Shape Microbiome Maturation during Early Life. *Sci. Transl. Med.* 8, 343ra82. doi:10.1126/scitranslmed.aad7121
- Bokulich, N. A., Ziemski, M., Robeson, M. S., and Kaehler, B. D. (2020). Measuring the Microbiome: Best Practices for Developing and Benchmarking Microbiomics Methods. *Comput. Struct. Biotechnol. J.* 18, 4048–4062. doi:10.1016/j.csbj.2020.11.049
- Bolyen, E., Rideout, J. R., Dillon, M. R., Bokulich, N. A., Abnet, C., Al-Ghalith, G. A., et al. (2018). QIIME 2: Reproducible, Interactive, Scalable, and Extensible Microbiome Data Science. *PeerJ* 6, e27295v1. doi:10.1038/s41587-019-0209-9
- Boutorh, A., and Guessoum, A. (2016). Complex Diseases SNP Selection and Classification by Hybrid Association Rule Mining and Artificial Neural Network-Based Evolutionary Algorithms. *Eng. Appl. Artif. Intelligence* 51, 58–70. doi:10.1016/j.engappai.2016.01.004
- Callahan, B. J., McMurdie, P. J., and Holmes, S. P. (2017). Exact Sequence Variants Should Replace Operational Taxonomic Units in Marker-Gene Data Analysis. *ISME J.* 11 (12), 2639–2643. doi:10.1038/ismej.2017.119
- Carmona-Saez, P., Chagoyen, M., Rodriguez, A., Trelles, O., Carazo, J. M., and Pascual-Montano, A. (2006). Integrated Analysis of Gene Expression by Association Rules Discovery. *BMC bioinformatics* 7 (1), 54–16. doi:10.1186/1471-2105-7-54
- Chaffron, S., Rehrauer, H., Pernthaler, J., and Von Mering, C. (2010). A Global Network of Coexisting Microbes from Environmental and Whole-Genome Sequence Data. *Genome Res.* 20 (7), 947–959. doi:10.1101/gr.104521.109
- Duvallet, C., Gibbons, S. M., Gurry, T., Irizarry, R. A., and Alm, E. J. (2017). Meta-analysis of Gut Microbiome Studies Identifies Disease-specific and Shared Responses. *Nat. Commun.* 8 (1), 1784. doi:10.1038/s41467-017-01973-8
- Faust, K., and Raes, J. (2012). Microbial Interactions: from Networks to Models. *Nat. Rev. Microbiol.* 10 (8), 538–550. doi:10.1038/nrmicro2832
- Faust, K. (2021). Open Challenges for Microbial Network Construction and Analysis. *ISME J.* 15, 3111–3118. doi:10.1038/s41396-021-01027-4
- Franceschini, A., Szklarczyk, D., Frankild, S., Kuhn, M., Simonovic, M., Roth, A., et al. (2012). STRING v9.1: Protein-Protein Interaction Networks, with Increased Coverage and Integration. *Nucleic Acids Res.* 41 (D1), D808–D815. doi:10.1093/nar/gks1094
- Galimberti, A., Bruno, A., Agostinetto, G., Casiraghi, M., Guzzetti, L., and Labra, M. (2021). Fermented Food Products in the Era of Globalization: Tradition Meets Biotechnology Innovations. *Curr. Opin. Biotechnol.* 70, 36–41. doi:10.1016/j.copbio.2020.10.006
- Ghannam, R. B., and Techtman, S. M. (2021). Machine Learning Applications in Microbial Ecology, Human Microbiome Studies, and Environmental Monitoring. *Comput. Struct. Biotechnol. J.* 19, 1092–1107. doi:10.1016/j.csbj.2021.01.028
- Gloor, G. B., Macklaim, J. M., Pawlowsky-Glahn, V., and Egozcue, J. J. (2017). Microbiome Datasets Are Compositional: and This Is Not Optional. *Front. Microbiol.* 8, 2224. doi:10.3389/fmicb.2017.02224
- Goethals, B. (2005). “Frequent Set Mining,” in *Data Mining and Knowledge Discovery Handbook* (Boston, MA: Springer), 377–397. doi:10.1007/0-387-25465-X_17
- Gonzalez, A., Navas-Molina, J. A., Kosciolk, T., McDonald, D., Vázquez-Baeza, Y., Ackermann, G., et al. (2018). Qiita: Rapid, Web-Enabled Microbiome Meta-Analysis. *Nat. Methods* 15 (10), 796–798. doi:10.1038/s41592-018-0141-9
- Hahsler, M., Chelluboina, S., Hornik, K., and Buchta, C. (2011). The Arules R-Package Ecosystem: Analyzing Interesting Patterns from Large Transaction Data Sets. *J. Machine Learn. Res.* 12, 2021–2025. doi:10.5555/1953048.2021064
- Han, J., Pei, J., Yin, Y., and Mao, R. (2004). Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach. *Data Mining Knowledge Discov.* 8 (1), 53–87. doi:10.1023/B:DAMI.0000005258.31418.83
- Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., et al. (2020). Array Programming with NumPy. *Nature* 585 (7825), 357–362. doi:10.1038/s41586-020-2649-2
- Hornik, K., Grün, B., and Hahsler, M. (2005). arules-A Computational Environment for Mining Association Rules and Frequent Item Sets. *J. Stat. Softw.* 14 (15), 1–25. doi:10.18637/jss.v014.i15
- Hosoda, S., Nishijima, S., Fukunaga, T., Hattori, M., and Hamada, M. (2020). Revealing the Microbial Assemblage Structure in the Human Gut Microbiome Using Latent Dirichlet Allocation. *Microbiome* 8 (1), 95–12. doi:10.1186/s40168-020-00864-3
- Hussein, N., Alashqur, A., and Sowon, B. (2015). Using the Interestingness Measure Lift to Generate Association Rules. *J. Adv. Comput. Sci. Technolog* 4 (1), 156. doi:10.14419/jacst.v4i1.4398
- Jordan, M. I., and Mitchell, T. M. (2015). Machine Learning: Trends, Perspectives, and Prospects. *Science* 349 (6245), 255–260. doi:10.1126/science.aaa8415
- Karpinet, T. V., Park, B. H., and Uberbacher, E. C. (2012). Analyzing Large Biological Datasets with Association Networks. *Nucleic Acids Res.* 40 (17), e131. doi:10.1093/nar/gks403
- Kato, T., Fukuda, S., Fujiwara, A., Suda, W., Hattori, M., Kikuchi, J., et al. (2014). Multiple Omics Uncovers Host-Gut Microbial Mutualism during Prebiotic Fructooligosaccharide Supplementation. *DNA Res.* 21 (5), 469–480. doi:10.1093/dnares/dsu013
- Knight, R., Vrbanc, A., Taylor, B. C., Aksenov, A., Callewaert, C., Debelius, J., et al. (2018). Best Practices for Analysing Microbiomes. *Nat. Rev. Microbiol.* 16 (7), 410–422. doi:10.1038/s41579-018-0029-9
- Koyutürk, M., Kim, Y., Subramaniam, S., Szpankowski, W., and Grama, A. (2006). Detecting Conserved Interaction Patterns in Biological Networks. *J. Comput. Biol.* 13 (7), 1299–1322. doi:10.1089/cmb.2006.13.1299
- Kyrpides, N. C., Eloe-Fadrosh, E. A., and Ivanova, N. N. (2016). Microbiome Data Science: Understanding Our Microbial Planet. *Trends Microbiol.* 24 (6), 425–427. doi:10.1016/j.tim.2016.02.011
- Layeghifard, M., Hwang, D. M., and Guttman, D. S. (2017). Disentangling Interactions in the Microbiome: a Network Perspective. *Trends Microbiol.* 25 (3), 217–228. doi:10.1016/j.tim.2016.11.008
- Lima-Mendez, G., Faust, K., Henry, N., Decelle, J., Colin, S., Carcillo, F., et al. (2015). Ocean Plankton. Determinants of Community Structure in the Global Plankton Interactome. *Science* 348, 1262073. doi:10.1126/science.1262073
- Liu, M., Ye, Y., Jiang, J., and Yang, K. (2021). MANIEA: A Microbial Association Network Inference Method Based on Improved Eclat Association Rule Mining Algorithm. *Bioinformatics* 2021, btab241. doi:10.1093/bioinformatics/btab241
- Ma, B., Wang, Y., Ye, S., Liu, S., Stirling, E., Gilbert, J. A., et al. (2020). Earth Microbial Co-occurrence Network Reveals Interconnection Pattern across Microbiomes. *Microbiome* 8, 82–12. doi:10.1186/s40168-020-00857-2
- Manda, P., McCarthy, F., and Bridges, S. M. (2013). Interestingness Measures and Strategies for Mining Multi-Ontology Multi-Level Association Rules from Gene Ontology Annotations for the Discovery of New GO Relationships. *J. Biomed. Inform.* 46 (5), 849–856. doi:10.1016/j.jbi.2013.06.012
- Manda, P. (2020). Data Mining Powered by the Gene Ontology. *Wires Data Mining Knowl Discov.* 10 (3), e1359. doi:10.1002/widm.1359

- Manda, P., Ozkan, S., Wang, H., McCarthy, F., and Bridges, S. M. (2012). Cross-ontology Multi-Level Association Rule Mining in the Gene Ontology. *PLoS ONE* 7, e47411. doi:10.1371/journal.pone.0047411
- McKinney, W. (2010). "Data Structures for Statistical Computing in Python," in Proceedings of the 9th Python in Science Conference, Austin, Texas, June 2010, 445, 51–56. doi:10.25080/Majora-92bf1922-00a
- Mitchell, A. L., Almeida, A., Beracochea, M., Boland, M., Burgin, J., Cochrane, G., et al. (2020). MGnify: the Microbiome Analysis Resource in 2020. *Nucleic Acids Res.* 48 (D1), D570–D578. doi:10.1093/nar/gkz1035
- Montassier, E., Al-Ghalith, G. A., Ward, T., Corvec, S., Gastinne, T., Potel, G., et al. (2016). Erratum to: Pretreatment Gut Microbiome Predicts Chemotherapy-Related Bloodstream Infection. *Genome Med.* 8 (1), 61–11. doi:10.1186/s13073-016-0321-0
- Muñio, D. P., and Borgelt, C. (2014). Frequent Item Set Mining for Sequential Data: Synchrony in Neuronal Spike Trains. *Intell. Data Anal.* 18 (6), 997–1012. doi:10.3233/ida-140681
- Naulaerts, S., Meysman, P., Bittremieux, W., Vu, T. N., Vanden Berghe, W., Goethals, B., et al. (2015). A Primer to Frequent Itemset Mining for Bioinformatics. *Brief. Bioinform.* 16 (2), 216–231. doi:10.1093/bib/bbt074
- Naulaerts, S., Moens, S., Engelen, K., Berghe, W. V., Goethals, B., Laukens, K., et al. (2016). Practical Approaches for Mining Frequent Patterns in Molecular Datasets. *Bioinform. Biol. Insights* 10, 37–47. doi:10.4137/BBI.S38419
- Noor, E., Cherkaoui, S., and Sauer, U. (2019). Biological Insights through Omics Data Integration. *Curr. Opin. Syst. Biol.* 15, 39–47. doi:10.1016/j.coisb.2019.03.007
- Omiecinski, E. R. (2003). Alternative Interest Measures for Mining Associations in Databases. *IEEE Trans. Knowl. Data Eng.* 15, 57–69. doi:10.1109/TKDE.2003.1161582
- Ong, H. F., Mustapha, N., Hamdan, H., Rosli, R., and Mustapha, A. (2020). Informative Top-K Class Associative Rule for Cancer Biomarker Discovery on Microarray Data. *Expert Syst. Appl.* 146, 113169. doi:10.1016/j.eswa.2019.113169
- Pasolli, E., Truong, D. T., Malik, F., Waldron, L., and Segata, N. (2016). Machine Learning Meta-Analysis of Large Metagenomic Datasets: Tools and Biological Insights. *Plos Comput. Biol.* 12 (7), e1004977. doi:10.1371/journal.pcbi.1004977
- Qu, K., Guo, F., Liu, X., Lin, Y., and Zou, Q. (2019). Application of Machine Learning in Microbiology. *Front. Microbiol.* 10, 827. doi:10.3389/fmicb.2019.00827
- Raschka, S. (2018). MLxtend: Providing Machine Learning and Data Science Utilities and Extensions to Python's Scientific Computing Stack. *J. Open Source Softw.* 3 (24), 638. doi:10.21105/joss.00638
- Ravel, J., Gajer, P., Abdo, Z., Schneider, G. M., Koenig, S. S., McCulle, S. L., et al. (2011). Vaginal Microbiome of Reproductive-Age Women. *Proc. Natl. Acad. Sci. U S A.* 108 (Suppl. 1), 4680. doi:10.1073/pnas.1002611107
- Reback, J., McKinney, W. J., Den Van Bossche, J., Augspurger, T., Cloud, P., and Sinhrks (2020). *Pandas-dev/pandas: Pandas 1.0*. 3. Zenodo. doi:10.5281/zenodo.3509134
- Schloss, P. D., and Westcott, S. L. (2011). Assessing and Improving Methods Used in Operational Taxonomic Unit-Based Approaches for 16S rRNA Gene Sequence Analysis. *Appl. Environ. Microbiol.* 77 (10), 3219–3226. doi:10.1128/AEM.02810-10
- Srivastava, D., Baksi, K. D., Kuntal, B. K., and Mande, S. S. (2019). "EviMass": A Literature Evidence-Based Miner for Human Microbial Associations. *Front. Genet.* 10, 849. doi:10.3389/fgene.2019.00849
- Su, X., Jing, G., Zhang, Y., and Wu, S. (2020). Method Development for Cross-Study Microbiome Data Mining: Challenges and Opportunities. *Comput. Struct. Biotechnol. J.* 18, 2075–2080. doi:10.1016/j.csbj.2020.07.020
- Tan, P.-N., Kumar, V., and Srivastava, J. (2002). Selecting the Right Interestingness Measure for Association Patterns. *Proc. ACM SIGKDD Int.* 2002, 32–41. doi:10.1145/775047.775053
- Tandon, D., Haque, M. M., and Mande, S. S. (2016). Inferring Intra-community Microbial Interaction Patterns from Metagenomic Datasets Using Associative Rule Mining Techniques. *PloS one* 11 (4), e0154493. doi:10.1371/journal.pone.0154493
- Tang, L., Zhang, L., Luo, P., and Wang, M. (2012). "Incorporating Occupancy into Frequent Pattern Mining for High Quality Pattern Recommendation," in Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM '12) (New York, NY, United States: Association for Computing Machinery), 75–84. doi:10.1145/2396761.2396775
- Tatti, N., and Mampaey, M. (2010). Using Background Knowledge to Rank Itemsets. *Data Min. Knowl. Disc.* 21 (2), 293–309. doi:10.1007/s10618-010-0188-4
- Thompson, J., Johansen, R., Dunbar, J., and Munsy, B. (2019). Machine Learning to Predict Microbial Community Functions: an Analysis of Dissolved Organic Carbon from Litter Decomposition. *PLoS One* 14 (7), e0215502. doi:10.1371/journal.pone.0215502
- Turnbaugh, P. J., Ley, R. E., Hamady, M., Fraser-Liggett, C. M., Knight, R., and Gordon, J. I. (2007). The Human Microbiome Project. *Nature* 449 (7164), 804–810. doi:10.1038/nature06244
- Vangay, P., Hillmann, B. M., and Knights, D. (2019). Microbiome Learning Repo (ML Repo): A Public Repository of Microbiome Regression and Classification Tasks. *Gigascience* 8 (5), giz042. doi:10.1093/gigascience/giz042
- Weiss, S., Van Treuren, W., Lozupone, C., Faust, K., Friedman, J., Deng, Y., et al. (2016). Correlation Detection Strategies in Microbial Data Sets Vary Widely in Sensitivity and Precision. *ISME J.* 10 (7), 1669–1681. doi:10.1038/ismej.2015.235
- Wood-Charlson, E. M., Anubhav, D., Auberry, D., Blanco, H., Borkum, M. I., Corilo, Y. E., et al. (2020). The National Microbiome Data Collaborative: Enabling Microbiome Science. *Nat. Rev. Microbiol.* 18 (6), 313–314. doi:10.1038/s41579-020-0377-0
- Xiao, S., Fei, N., Pang, X., Shen, J., Wang, L., Zhang, B., et al. (2014). A Gut Microbiota-Targeted Dietary Intervention for Amelioration of Chronic Inflammation Underlying Metabolic Syndrome. *FEMS Microbiol. Ecol.* 87 (2), 357–367. doi:10.1111/1574-6941.12228
- Xiong, H., Tan, P.-N., and Kumar, V. (2006). Hyperclique Pattern Discovery. *Data Min. Knowl. Disc.* 13 (2), 219–242. doi:10.1007/s10618-006-0043-9
- Yoon, Y., and Lee, G. (2011). Subcellular Localization Prediction through Boosting Association Rules. *Ieee/acm Trans. Comput. Biol. Bioinform.* 9 (2), 609–618. doi:10.1109/TCBB.2011.131
- Zakrzewski, M., Proietti, C., Ellis, J. J., Hasan, S., Brion, M. J., Berger, B., et al. (2017). Calypso: a User-Friendly Web-Server for Mining and Visualizing Microbiome-Environment Interactions. *Bioinformatics* 33 (5), 782–783. doi:10.1093/bioinformatics/btw725
- Zhou, C., Meysman, P., Cule, B., Laukens, K., and Goethals, B. (2013). "Mining Spatially Cohesive Itemsets in Protein Molecular Structures," in Proceedings of the 12th International Workshop on Data Mining in Bioinformatics (BioKDD '13) (New York, NY, United States: Association for Computing Machinery), 42–50. doi:10.1145/2500863.2500871

Conflict of Interest: Author SA was employed by the company Quantia Consulting Srl.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Giulia, Anna, Antonia, Dario and Maurizio. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Interactive, Visual Simulation of a Spatio-Temporal Model of Gas Exchange in the Human Alveolus

Kerstin Schmid^{1*}, Andreas Knoten², Alexander Mück², Keram Pfeiffer³,
Sebastian von Mammen² and Sabine C. Fischer¹

¹Supramolecular and Cellular Simulations, Center for Computational and Theoretical Biology, Faculty of Biology, University of Würzburg, Würzburg, Germany, ²Human Computer Interaction, Institute of Computer Science, Faculty of Mathematics and Computer Science, University of Würzburg, Würzburg, Germany, ³Behavioral Physiology and Sociobiology, Biocenter, Faculty of Biology, University of Würzburg, Würzburg, Germany

OPEN ACCESS

Edited by:

Lydia Gregg,
Johns Hopkins University,
United States

Reviewed by:

Anamaria Crisan,
Salesforce, United States
Michael Corrin,
University of Toronto Mississauga,
Canada

*Correspondence:

Kerstin Schmid
kerstin.schmid@uni-wuerzburg.de

Specialty section:

This article was submitted to
Data Visualization,
a section of the journal
Frontiers in Bioinformatics

Received: 11 September 2021

Accepted: 17 December 2021

Published: 26 January 2022

Citation:

Schmid K, Knoten A, Mück A, Pfeiffer K,
von Mammen S and Fischer SC (2022)
Interactive, Visual Simulation of a
Spatio-Temporal Model of Gas
Exchange in the Human Alveolus.
Front. Bioinform. 1:774300.
doi: 10.3389/fbinf.2021.774300

In interdisciplinary fields such as systems biology, good communication between experimentalists and theorists is crucial for the success of a project. Theoretical modeling in physiology usually describes complex systems with many interdependencies. On one hand, these models have to be grounded on experimental data. On the other hand, experimenters must be able to understand the interdependent complexities of the theoretical model in order to interpret the model's results in the physiological context. We promote interactive, visual simulations as an engaging way to present theoretical models in physiology and to make complex processes tangible. Based on a requirements analysis, we developed a new model for gas exchange in the human alveolus in combination with an interactive simulation software named *Alvin*. *Alvin* exceeds the current standard with its spatio-temporal resolution and a combination of visual and quantitative feedback. In *Alvin*, the course of the simulation can be traced in a three-dimensional rendering of an alveolus and dynamic plots. The user can interact by configuring essential model parameters. *Alvin* allows to run and compare multiple simulation instances simultaneously. We exemplified the use of *Alvin* for research by identifying unknown dependencies in published experimental data. Employing a detailed questionnaire, we showed the benefits of *Alvin* for education. We postulate that interactive, visual simulation of theoretical models, as we have implemented with *Alvin* on respiratory processes in the alveolus, can be of great help for communication between specialists and thereby advancing research.

Keywords: interactive simulation, visualization, theoretical modeling, lung physiology, requirements analysis, spatio-temporal resolution, education

1 INTRODUCTION

Systems biology is a highly interdisciplinary research field that integrates theoretical modeling and experimental data (Gavaghan et al., 2006). A key component of projects with valuable scientific progress is close cooperation between experimentalists and theorists (Byrne et al., 2006; Drubin and Oster, 2010; Welsh et al., 2006). However, this entails certain challenges. Different ways of thinking and terminologies or jargon often hinder communication between the disciplines. Ongoing efforts to bridge the gap include educational reviews [e.g., (Sharpe, 2017; Fischer, 2019)], summer schools,

special research programs (<https://www.newton.ac.uk/event/cgp/>) and large multi-laboratory initiatives such as the Virtual Physiological Human (Viceconti et al., 2008) or The Virtual Brain (<https://www.thevirtualbrain.org>). Key components of these approaches are informative visualizations and the possibility of hands-on experience.

The goal of our study was to create a tool to better present modeling results to experimenters. To this end, we consider communicating results of mathematical modeling in physiology. In publications, models are usually presented as follows (Mogilner et al., 2011): The model definition is given in terms of mathematical equations, occasionally supported by schematic diagrams describing the model structure. For the corresponding simulations, all parameter values are listed and the output is visualized in graphs and compared with experimental data, where appropriate. When modeling spatial structures and processes, the simulation output is presented in still images or, if possible, animations (Chao, 2003; Lin et al., 2004; Saber and Heydari, 2012). As an alternative for the communication of state of the art theoretical models, we promote interactive, visual simulation. Previous approaches include computer-aided diagnosis software (Xiong et al., 2017; Conover et al., 2018) or systems for medical education (Jacob et al., 2012; Jamniczky et al., 2012; Costabile, 2021). We focus on the human lung. Existing interactive systems for teaching in this field address respiratory mechanics (Kuebler et al., 2007; Warliah et al., 2012) or gas exchange (Kapitan, 2008). All above systems for teaching convey established educational content. They have not been intended to advance the current state of research. In contrast, (Winkler et al., 1995) argue that their interactive system has great utility beyond its educational use. They have developed an application that provides an interactive interface with a simulation of a multi-compartment model. Ventilation mechanics, gas transport, gas mixing and gas exchange are considered. However, the actual process of gas exchange, the key functionality of the human lung, remains as abstract as the site where it occurs.

We thus focused on the smallest functional unit of the lung - the alveolus. The overarching goal was to provide an interactive visualization of the process of gas exchange in the human alveolus for research and education. We refined and combined existing models (Weibel et al., 1993; Dash et al., 2016) to cover the complete transport of oxygen into hemoglobin. The resulting model provided the computational core for an interactive simulation software named *Alvin*. *Alvin* facilitates investigations of relationships between morphological and physiological factors and the course of gas exchange. The software enables systematic investigations of our model with respect to experimental data. We aimed to maximize the usability of *Alvin* for both research-related and educational usage. As an exemplary use case in research, we present a plausibility check of pulmonary diffusion capacity measurements. Concerning the applicability of *Alvin* in teaching, we present the details of its integration into a digital physiology lab course for undergraduate students and the results of a corresponding survey among its participants. The software is available for download at <https://go.uni-wue.de/alvin>.

Particular about our work is the development of the mathematical model with the aim of visualization in combination with the requirements-based engineering of the simulation software. This resulted in an advanced gas exchange model and an interactive application that exceed the existing standard. Specifically, design features as the ability to run and compare multiple simulation instances at the same time and the combination of providing parameter value presets as well as allowing parameter configurations by the user are key contributions to the field. This results in an educationally valuable application that also allows revealing unknown underlying assumptions of results presented in the literature. Taken together, our work demonstrates that an interactive, visual simulation is a versatile and powerful tool to visualize modeling results for both researchers and students.

2 METHODS

On the basis of our goals, corresponding requirements were defined in a user-centered engineering approach. Our interdisciplinary team included a development team (AK, AM, KS) and supervising experts (SvM for games engineering, SCF for mathematical modelling, KP for physiology education). Concepts on requirements were first drafted within the development team. These concepts were then either acknowledged by experts/stakeholders in a quality gateway or returned for revision. The higher-level requirements could be categorized into three groups: Scientific (S), educational (E) and accessibility (A) requirements.

- S.1. Gas exchange model suitable for interactive configuration.
- S.2. Interfaces for interaction.
- S.3. Quantitative simulation output.
- S.4. Visual feedback that emphasizes the connection between structure and function of the alveolus.
- E.1. Presentation of educationally relevant respiratory phenomena.
- E.2. Facilitate autonomous work with the application.
- A.1. Compatibility with common devices (computers or tablets with windows, iOS or linux).
- A.2. Simple and clear GUI (to enhance the intuitive use of the system).
- A.3. Applicability to the widest possible range of scientific issues.

In an iterative process, system requirements and final design requirements were developed from these higher-level user requirements (and recorded in a total of 166 GitLab issues). The complete set of requirements is listed in Section S1.1 of the **Supplementary Material**.

3 RESULTS AND DISCUSSION

3.1 Integrative Alveolar Gas Exchange Model

The human lung consists of progressively branching bronchi and bronchioles, and blood vessels follow this structure (Hsia et al.,

2016). The respiratory zone begins where the first alveoli adjoin the bronchioles (Haefeli-Bleuer and Weibel, 1988). Alveoli are hollow protrusions that have a large surface area and a thin tissue barrier. They are surrounded by a dense network of fine capillaries (Weibel and Gomez, 1962). Within an alveolus, inhaled air passes through the cavity and gas exchange with the capillary blood takes place through the tissue barrier (Weibel, 2009). An alveolus thus represents the smallest functional unit of the lung. We established a spatio-temporal model of gas exchange in the human alveolus based on empirically established models (Weibel et al., 1993; Dash et al., 2016) (requirement S.1). This entailed the integration of the established models and the alignment of their numerical scales. Any gaps in the model had to be identified and closed. Finally, the new model was validated against data from the literature.

3.1.1 Model

The process of gas exchange in an alveolus can be divided into two sequential steps (Roughton and Forster, 1957): 1. The diffusion of oxygen through the tissue barrier into the blood and red blood cells and 2. its binding to hemoglobin (Hb). For each step, we adopted an established model describing this process (Weibel et al., 1993; Dash et al., 2016). By integrating the two sub-models into a complete model we can simulate the entire process of gas exchange inside an alveolus. The diffusion of oxygen across the alveolar wall is calculated based on Fick's law (Weibel et al., 1993), resulting in

$$\nu = \text{DMO}_2 \cdot \Delta p\text{O}_2 = K_{\text{O}_2} \cdot \frac{s}{\tau} \cdot \Delta p\text{O}_2 \quad (1)$$

The oxygen flow ν across the barrier is a function of the pressure gradient $\Delta p\text{O}_2$ between air and blood and morphological parameters that contribute to the so called membrane diffusing capacity for oxygen DMO_2 . More precisely, DMO_2 comprises the ratio between surface area s and barrier thickness τ multiplied by the permeability coefficient K_{O_2} . Standing alone, this calculation would yield a mean quantity of oxygen flow in the alveolus. However, the potential of visualization should be exploited and the course of diffusion along the capillary should be shown in the alveolar model. This is particularly interesting as partial pressures of respiratory gases inside the blood are not homogeneous in the alveolar region. Gas exchange leads to oxygen (O_2) and carbon dioxide (CO_2) pressure gradients in the alveolar capillary. In a healthy individual, blood enters this area with a low partial pressure of oxygen ($p\text{O}_2$) and a high partial pressure of carbon dioxide ($p\text{CO}_2$). Diffusion of O_2 from the alveolus into the capillary and of CO_2 out of the capillary into the alveolus gradually increases $p\text{O}_2$ and decreases $p\text{CO}_2$ until the distribution of gases reaches equilibrium (Powers and Dhamoon, 2019). Hence, the course of pressure gradients depends on the efficiency of gas diffusion and the blood flow velocity. To map O_2 and CO_2 pressure gradients in our model, a representative capillary was divided into subsections of equal size (**Figure 1**). Oxygen diffusion from the alveolar space into the different sections is calculated successively starting with the first section. Here, blood enters with a preset $p\text{O}_2$. This involves a partial pressure gradient with respect to the alveolar space. The

diffusion along this gradient is calculated according to **Eq. 1**. The absolute amount of oxygen that reaches this capillary section is calculated from this oxygen flow and the blood flow velocity. It affects the $p\text{O}_2$ of the blood in the next section, which is considered in a new calculation cycle and so on.

The quantity of CO_2 diffusing out of the capillary and into the alveolus is determined via the respiratory exchange ratio from the quantity of oxygen that is taken up by the blood. The respiratory exchange ratio is defined as the amount of CO_2 produced divided by the amount of O_2 consumed. This ratio is assessed by analyzing exhaled air in comparison with the environmental air and its average value for the human diet is around 0.82 (Sharma et al., 2020). Taken together, this provides a time-resolved model for the first step of gas exchange: The diffusion of oxygen from inhaled air into the capillary blood of the alveolus and of carbon dioxide in the reverse direction.

In a second step, the binding of O_2 and CO_2 to hemoglobin was adopted from (Dash et al., 2016), such that

$$S_{\text{HbO}_2} = \frac{(p\text{O}_2/p50)^{nH}}{1 + (p\text{O}_2/p50)^{nH}} \quad (2)$$

Hemoglobin oxygen saturation (S_{HbO_2}) is expressed as a Hill function depending on $p\text{O}_2$, the Hill coefficient nH and $p50$, the value of $p\text{O}_2$ at which hemoglobin is 50% saturated with O_2 . The parameter nH , in turn, depends on $p\text{O}_2$. Polynomial expressions describe the dependence of $p50$ on $p\text{CO}_2$ in the blood, blood temperature, the pH inside erythrocytes (pH_{rbc}) and concentration of the organic phosphate 2,3-bisphosphoglycerate ([2,3]-DPG). These dependencies have been described and fitted to several experimental data sets (Dash et al., 2016) for a wide range of parameter values (fulfills requirement A.3.1). In our model, S_{HbO_2} is calculated for each section according to the $p\text{O}_2$ and $p\text{CO}_2$ gradients along the capillary sections determined in step 1. Hence, we obtain the distribution of blood oxygen saturation along the capillary as the main output of our model.

Together, this yields a model for the complete process of oxygen transport from inhaled air into hemoglobin in the blood with spatio-temporal resolution. All parameters essential for the model and their default values were collected from the literature and represent a normal, healthy condition (**Table 1**).

3.1.2 Model Validation

In a first step of model validation, we analysed whether the two sub models from step 1 and step 2 had been sensibly adapted from the literature. In our model, oxygen diffusion is estimated for a single alveolus with a surface area of $121,000 \mu\text{m}^2$. Other parameters affecting DMO_2 (namely tissue barrier thickness and permeability coefficient, see **Eq. 1**) were adopted without change. DMO_2 of the whole lung in relation to body weight (bw) was estimated as $0.079 \text{ ml}/(\text{s} \times \text{mmHg} \times \text{kg})$ (Weibel et al., 1993). To compare our model result ($\text{DMO}_2^{(\text{model})} = 6 \times 10^{-9} \text{ ml}/(\text{s} \times \text{mmHg})$) with Weibel's estimate, it needs to be extrapolated to the organ scale. Multiplying $\text{DMO}_2^{(\text{model})}$ by the number of alveoli in the human lung (480×10^6 (Ochs et al., 2004)) results in a $\text{DMO}_2^{(\text{model, extrapolated})}$ of $2.88 \text{ ml}/(\text{s} \times \text{mmHg})$. This value is

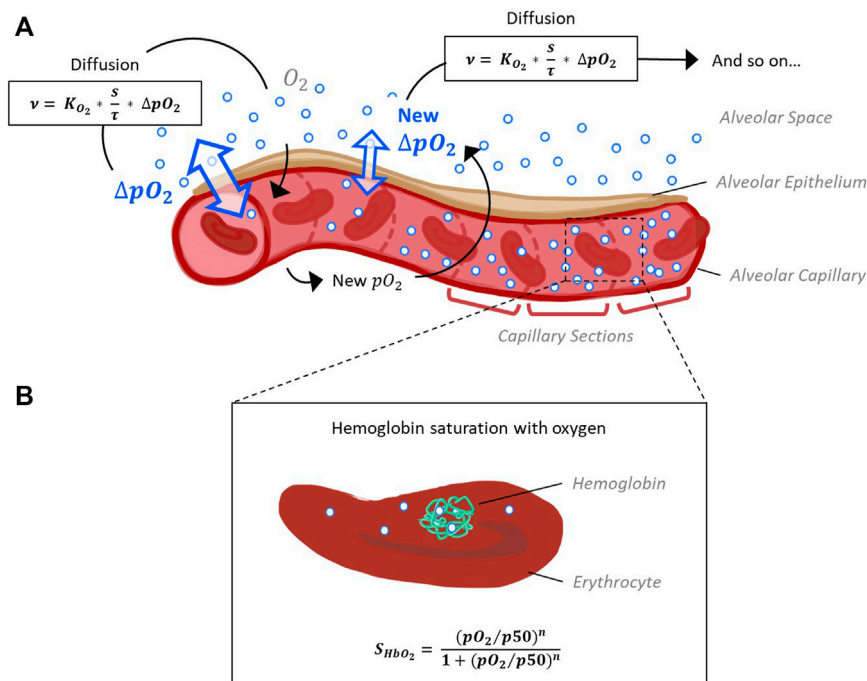


FIGURE 1 | Schematic representation of the model capillary with erythrocytes, separated from alveolar space by a single cell layer of alveolar epithelium. **(A)** In order to reconstruct O_2 and CO_2 pressure gradients along the capillary, it is divided into sections of equal size. The pressure gradient between alveolar space and blood (Δp_{O_2}) and the resulting flow of oxygen along this gradient is calculated for each section subsequently, as oxygen flow into one section affects p_{O_2} and thus Δp_{O_2} of the next section. Calculation of oxygen diffusion depending on Δp_{O_2} is based on Fick's law (Weibel et al., 1993). **(B)** According to the p_{O_2} and p_{CO_2} gradients along the capillary sections determined in step 1, hemoglobin oxygen saturation (S_{HbO_2}) is calculated for each section. The corresponding Hill equation has been defined and fitted to experimental data (Dash et al., 2016).

TABLE 1 | Model parameters and their default values. Values of morphological and physiological parameters of the gas exchange model were collected from literature. All values given are mean values referring to a single alveolus.

Parameter	Unit	Default value	References	Value range
Alveolar p_{O_2}	mmHg	100	Sharma et al. (2020)	1–150
Blood p_{O_2}	mmHg	40	Dash et al. (2016)	1–150
Alveolar p_{CO_2}	mmHg	40	Sharma et al. (2020)	1–150
Blood p_{CO_2}	mmHg	45	Dash et al. (2016)	1–150
Surface area	μm^2	121,000	Mercer et al. (1994)	0–210,000
Thickness of tissue barrier	μm	1.11	Gehr et al. (1978); Weibel et al. (1993)	0.1–3.0
Blood flow velocity	mm/s	1	Abstracted from: Weibel et al. (1993); Petersson and Glenny, (2014)	0.01–2
Blood volume	μm^3	404,000 (50% "capillary recruitment")	Abstracted from: Gehr et al. (1978); Ochs et al. (2004); Okada et al. (1992)	1–808,000
Blood temperature	$^{\circ}C$	37	Dash et al. (2016)	20–44
Erythrocyte pH (pH_{rbc})		7.24	Dash et al. (2016)	5.8–8.2
Concentration of [2,3]-DPG	mM	4.65	Dash et al. (2016)	1–10
Capillary length	μm	500	Weibel et al. (1993)	*not adjustable
Capillary volume	μm^3	808,000	Ochs et al. (2004); Gehr et al. (1978)	*not adjustable
Capillary radius	μm	3.15	Mühlfeld et al. (2010)	*not adjustable
Number of capillaries		52	Calculated from capillary volume, radius and length	*not adjustable

distinctly lower than the DMO_2 estimated by Weibel et al., assuming a standard body weight of 70 kg: $DMO_2^{(Weibel, bw\ 70\ kg)} = 5.53\ ml/(s \times mmHg)$. This estimate has been based on morphometric studies in fully inflated, fluid-filled lungs (Weibel et al., 1993). It is recognized that in an air-filled lung, however,

only about 60–70% of the alveolar surface is exposed to air (Gil et al., 1979; Bachofen et al., 1987). The default value for surface area in our model was taken from studies on perfusion-fixed, air-filled lungs (Mercer et al., 1994). Hence, our combination of parameter values for the surface area of a single alveolus (Mercer

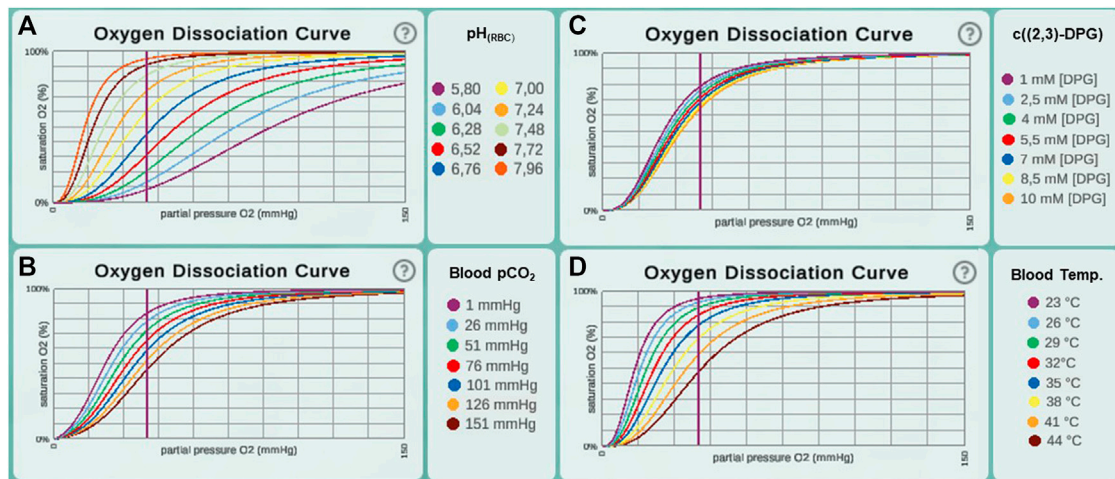


FIGURE 2 | Oxygen dissociation curves recreated in *Alvin* for different ranges of parameter values from the original paper (Dash et al., 2016). This includes value ranges for the parameters **(A)** pH in erythrocytes (pH_{RBC}), **(B)** blood pCO_2 , **(C)** concentration of [2,3]-DPG and **(D)** blood temperature.

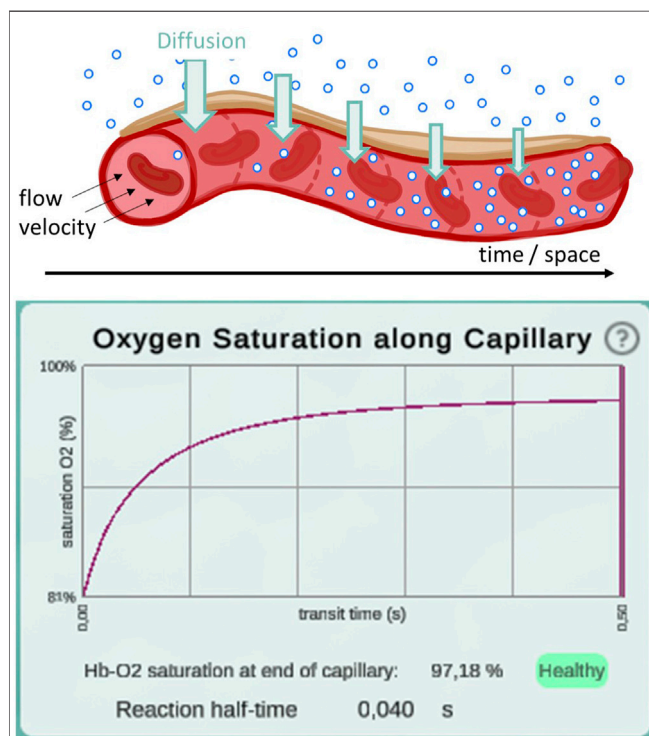


FIGURE 3 | Illustration of the diffusion gradient along the model capillary (top) and a screenshot of the plot displaying oxygen saturation along capillary between 81 and 97% (bottom). This screenshot was taken from a simulation with pO_2 values of 97 mmHg in the alveolar space and 46 mmHg in the deoxygenated blood. All other parameters remained at their default settings. Reaction half-time is defined as the time point at which 50% of the oxygenation that blood undergoes during its transit along the alveolus is reached.

et al., 1994) and the number of alveoli in the human lung (Ochs et al., 2004) produce a result that falls short of the previous estimate. However, the discrepancy is explained by known

differences in the morphometric methods used. We deliberately chose the surface value from the study on an air-filled lung to be as close as possible to the *in vivo* situation. The sub model describing hemoglobin oxygen saturation was adopted from the literature (Dash et al., 2016) without further modifications. Hb- O_2 dissociation curves across the different parameter ranges from this publication [Figure 4 E-H in (Dash et al., 2016)] were recreated and indicate a correct implementation of the model (Figure 2).

In a second step, the complete integrative model was validated. We used published experimental data to validate our model. A key contribution of our model is the temporal and spatial resolution. Rather than determining mean values, oxygen partial pressure and saturation gradients along the alveolar capillary are generated. This allows validation of the model in a physiological context. For default parameter settings, 50% of the oxygenation that blood undergoes during its transit along the alveolus is completed after 0.04 s (Figure 3). This measurement was performed for an increase in saturation from 81 to 97%, reaching the reaction half-time at 89%. The corresponding measurement in mice is 0.037 s (Tabuchi et al., 2013) and it has been argued that there are only slight differences between species (Lindstedt, 1984). In summary, we showed that we have correctly adopted and sensibly modified the individual models. Our new integrative model provides results that are consistent with experimental data.

3.1.3 Model Discussion

Our mathematical model was assembled from two existing sub models (Weibel et al., 1993; Dash et al., 2016). One sub model describes the diffusion rate of oxygen from the air into the blood depending on morphological properties (Weibel et al., 1993). In this preceding work, the lung has been defined simplistically as a single container of air and the partial pressure of oxygen in the blood has been considered constant. Some simplifications still exist in our new model. For example, the introduction of a

breathing pattern was neglected: Partial pressure changes in alveolar space only occur when respective parameter values are modified by the user (suggests that O_2 diffusing out of the alveolus is instantly replaced and CO_2 diffusing into the alveolus is evacuated immediately). Also, blood flow was approximated as a continuous flow of a homogeneous plasma/erythrocyte mixture. However, our new integrative model also features improvements compared to the original models. Instead of steady states, it provides information about oxygen transport over the continuous course of time. It has already been noted that a time-dependent modeling approach is better suited to reconstruct gas exchange in lung tissue than steady-state approaches (Sapoval et al., 2020). Accordingly, the temporal resolution is a valuable improvement to the model.

For validation, we compared reaction half-time results from our model with what has been reported in the literature (Tabuchi et al., 2013). Reaction half-time is defined as the time that elapses until 50% of the oxygenation that blood undergoes during its transit along the alveolus is complete. We measured 40 ms with default parameter settings. Experimentally, a half-time of 37 ms has been determined in mice (Tabuchi et al., 2013). Corresponding theoretical predictions have been slightly lower at 18–32 ms. Tabuchi et al. argue that this discrepancy is due to the fact that the oxygenation process already takes place in the precapillary arterioles, but for the prediction only capillaries were considered. Since only capillaries are considered in *Alvin* as well, we may suspect that our value underestimates the *in vivo* human reaction half-time slightly.

In our model, capillaries are divided into an arbitrary number of sections. The finer grained this discretisation, i.e. the smaller the individual sections and the larger their number, the larger is the resolution of calculated gas dynamics and, thus, the resulting accuracy. However, as described in the following section, our model forms the basis of a visual simulation. With higher resolution, the computational demand grows, especially due to the three-dimensional rendering of the respective capillary sections. Therefore, we manually optimised this detail to maximise the accuracy without jeopardising the simulation's interactivity.

3.2 Visualization and Interactivity: The *Alvin* Application

Interaction with content positively influences its conception (Pike et al., 2009; He et al., 2021) and helps to explore concepts. In parallel with the mathematical model, we developed the *Alvin* simulation software to support the conception and exploration of the gas exchange process in a single alveolus. Addressing the scientific, educational and accessibility requirements (see Methods), we aimed at maximal usability of the software for both research-related and educational applications. Overall, *Alvin* should impart an understanding of the relationship between structure and function of the alveolus.

3.2.1 Visualization

Alvin is a desktop-based application implemented in Unity. It is available for Windows, macOS and Linux (fulfills (A.1)). The user

interface of *Alvin* consists of the following core components: a three-dimensional model of an alveolus illustrating the simulation process, a configuration menu for model parameter values and a panel displaying dynamic graphs (**Figure 4**) (fulfills A.2.1). A key feature is the ability to run and compare multiple simulation instances at the same time.

The animated, three-dimensional model of an alveolus illustrates the current state of the simulation (**Figure 4**, center, see also Section S1.3 of the **Supplementary Material** for further details) (fulfills S.4.1). The alveolus is visually filled with small representations for air molecules, animated to signify Brownian motion. Each one is representing roughly 2×10^9 molecules of oxygen (red spheres), carbon dioxide (blue spheres) or nitrogen (white spheres), respectively. Thickening or thinning of the tissue layer indicates value changes of the model parameter “thickness of tissue barrier”. Erythrocytes are animated and move along the cut-open capillary. The number of erythrocytes proportionally corresponds to a standard value of 5×10^6 cells per μL blood (Pagana et al., 2019). Their relative position on this path is constantly tracked. Oxygen partial pressure (**Eq. 1**) and hemoglobin oxygen saturation (**Eq. 2**) gradients are calculated along the same path. This information is combined to color erythrocytes according to their oxygen saturation and to cumulatively total the amount of oxygen taken up by the erythrocytes over the course of the simulation (see **Figure 4**, graph “oxygen uptake”).

Hence, simulated gas exchange can be retraced by observing the amount of gas spheres crossing the tissue barrier from one side to the other and changes in capillary and erythrocyte coloring (S.4.2). Quantitative outcome of the simulation can be monitored on three different graphs (S.3.1) (**Figure 4**, right). They show hemoglobin oxygen saturation as a function of pO_2 in the blood (oxygen dissociation curve) (E.1.2), or of time (oxygen saturation along capillary). Finally, the total amount of oxygen taken up is tracked as a function of the time since the simulation was started or reset. Graphs of different simulation instances are indicated by their respective instance color.

3.2.2 Interactivity

The parameter panel (**Figure 4**, left) allows users to configure model parameter values. Changes in parameter values yield run-time updates in the 3D visualization and the quantitative graphs (S.2.1). A traffic light color code and keywords provide classification of the chosen parameter values with regard to their healthy or pathological ranges (E.2.2). More information can be obtained by clicking the respective info button (indicated by a question mark) (E.2.1). Model parameters are grouped in terms of the tissue components to which they relate (A.2.2). Visual highlighting in the 3D alveolus model emphasizes these connections (S.4.3). For instance, all tissue components except the capillary are grayed out when the cursor is over the window for model parameters relating to the blood. To examine the process in the 3D model in more detail, it can be moved, rotated or zoomed. Detailed quantitative information can be obtained by hovering over a graph with the mouse. The instance menu allows direct comparison of different parameter settings by running several simulation instances simultaneously

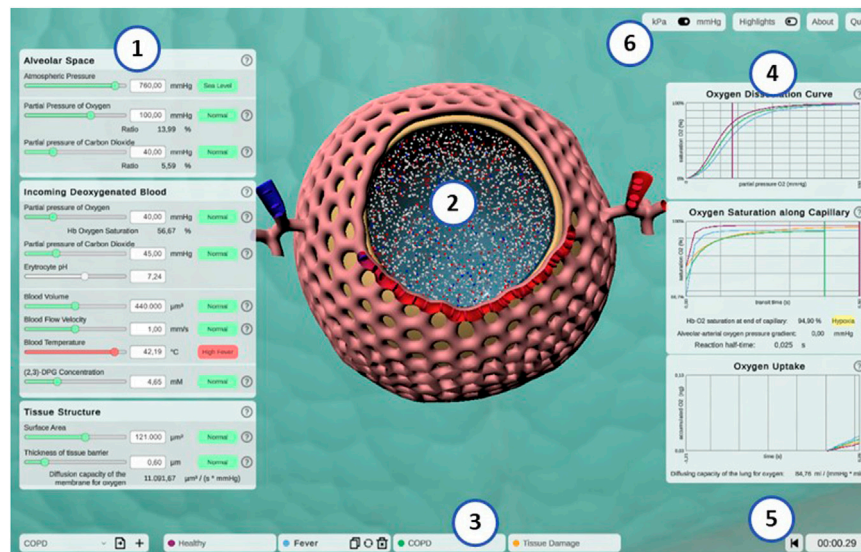


FIGURE 4 | Screenshot of the interactive application *Alvin*. (1) Model parameters are grouped in categories and can be configured by the user. Colors and information text provide possible real-world interpretation of the values. (2) Animated simulation of an alveolus for the active parameter set provides visualization of the effect of the model parameter values. (3) To increase exploratory value, multiple simulation instances can be compared. (4) Quantitative simulation output is displayed with plots color-coded for each active instance of the simulation. (5) Simulation time is displayed and can be reset. (6) Utility functions and settings are available.

TABLE 2 | Parameter value shifts in presets representing pathogenic conditions. For every condition, pathophysiological issues or symptoms are represented by increased (↑) or decreased (↓) values of the respective model parameters.

Pathogenic condition	Pathophysiology/Symptom	Parameter value shift
Pneumonia	Fever	Temperature ↑
	Tissue damage Accumulation of fluids and dead cells	Surface area ↓ Barrier thickness ↑
ARDS (acute respiratory distress syndrome)	Collapse (alveolar aelectasis)	Surface area ↓↓
	Fever	Temperature ↑
COPD (chronic obstructive pulmonary disease)	Impaired exhalation	Alveolar pCO ₂ ↑ and blood pCO ₂ ↑
	Impaired exhalation	Alveolar pO ₂ ↓
	Tissue damage	Surface area ↓
Pulmonary fibrosis	Thickened and scarred connective tissue	Barrier thickness ↑
	Impaired inhalation	Alveolar pCO ₂ ↓
Pulmonary embolism	shunt	Blood volume ↓↓
	shunt	Blood flow velocity ↓↓

(S.2.2) (**Figure 4**, bottom). Characteristic coloring and custom naming facilitate distinguishing between different simulation instances. A selected instance can be copied, deleted or reset to its initial parameter values. Parameter presets for healthy and common pathogenic conditions are provided (E.1.1) (**Table 2**). Finally, the user interface contains control elements to monitor or reset simulation time (S.2.3 and S.3.2) and to toggle between pressure units (A.3.2) and visual highlighting modes. More technical details on the implementation of *Alvin* are provided in Section S1.2 of the **Supplementary Material**. Taken together, these features present interrelationships of the gas exchange process as one explores the system. For example, the user can decrease the alveolar partial pressure of oxygen and observe how this affects the progression of oxygen binding to hemoglobin along the alveolar capillary. One could also observe at what

alveolar pO₂ the blood O₂ saturation reaches a critically low value at the end of the process. Another example would be to increase the tissue barrier thickness and observe how much the blood oxygen saturation decreases despite unchanged alveolar partial pressures.

3.2.3 Discussion on Visualization and Interactivity

Alvin intends to increase understanding of the complex relationships of gas exchange by highlighting connections and allowing comparison of multiple simulations. Previous interactive systems for gas exchange have pursued a similar goal. (Winkler et al., 1995) have modeled the lung as a complex of abstract gas exchange units (compartments) that can be simulated under individual conditions. (Kapitan, 2008) have created a model of gas exchange that is based on the alveolar

gas equation (Sharma et al., 2020) and takes the ratio of ventilation to perfusion into account. Both systems enable simulation of inhomogeneous distribution of ventilation and perfusion. This provides valuable insights into higher-level relationships. In both systems, individual gas exchange units and the whole complex are visualized by means of abstract schematic representations. What happens in detail and how it looks like remains unanswered. *Alvin* fills this gap. The site of gas exchange is no longer abstract—a 3D model illustrates an alveolus in realistic proportions. It conveys the structure of important components (capillary net, tissue barrier). The connection between structure and function is interactively explored in the simulation. Blood flow and tissue thickness in the 3D model adapt to the parameter settings and directly affect the simulation process. What further sets *Alvin* apart from the two systems mentioned above is the possibility of running multiple simulation instances simultaneously. This allows different conditions to be compared directly instead of being modeled and explored one after the other. However, the design of the instance menu in *Alvin* still has a limitation. While qualitative output of several simulation instances can be compared directly, the user is required to switch tabs along the instance menu to compare parameter settings and visual output on the 3D model. This issue should be addressed in future improvements to the system.

The combination of providing parameter value presets as well as allowing parameter configurations by the user enables a presentation of the model that expands existing best-practice (Mogilner et al., 2011). *Alvin* includes a multitude of visualization elements and interaction possibilities. They aim at an intuitive usage of the application and understanding of the gas exchange simulation. It should be assessed whether the use of *Alvin* is actually perceived as intuitive. For this purpose, in the context of a use case study (described in Section 3.3.2), we had a group of users fill out a standardized questionnaire to measure intuitive usability.

3.3 Applying *Alvin*: Use Case Studies

We provide two concrete examples for the application of *Alvin*. One of our goals was to ensure that researchers can flexibly explore the model simulation. Here, we demonstrate how the interactive simulation can be used to interpret data from the literature. Second, we report on *Alvin*'s integration into a university level virtual class. The application was used to convey basic and important respiratory processes in the context of a given instructional framework that combined a traditional lecture and instructor based- as well as self-learning.

3.3.1 *Alvin* in Research: Interpreting Data and Testing Predictions

To present a possible use case of *Alvin* for research, we employ the application to check the plausibility of pulmonary diffusion capacity measurements. The pulmonary diffusion capacity (D_{LO_2}) describes the lungs' capacity to transport oxygen from the air to the blood. It is defined as the oxygen consumption $\dot{V}O_2$ in L/min (oxygen uptake over time) divided by the mean oxygen pressure gradient between alveolar air and capillary blood ΔpO_2 (Lindstedt, 1984).

$$D_{LO_2} = \frac{\dot{V}O_2}{\Delta pO_2} \quad (3)$$

Physiological estimates of D_{LO_2} are usually derived from measurements of diffusion capacity for carbon monoxide (D_{LCO}) (Forster, 1964; Crapo and Crapo, 1983). Normal values of D_{LO_2} at rest are around 30 ml/(mmHg × min) (Hsia et al., 2016). Determination of D_{LO_2} based on morphometric data has resulted in a value of 158 ml/(mmHg × min) (Weibel, 2009) and thereby exceeds physiological approximations considerably. There are several reasons for this discrepancy (Hsia et al., 2016). One of them is that for the morphological estimation, a complete perfusion of the capillaries is assumed and the entire alveolar surface is included in the calculations (Weibel, 1970). Under normal conditions, only about 50% of capillary segments in the alveolar wall are perfused by erythrocytes and thus contribute to gas exchange (Okada et al., 1992) (Figure 5A). Increasing blood pressure (e.g., due to increased cardiac output) leads to recruitment of further capillary segments. In the perfusion fixed, air-filled lung, only about 60–70% of the alveolar surface area is exposed to air (Gil et al., 1979; Bachofen et al., 1987). In addition, lung volume changes during respiration depending on the transpulmonary pressure. It has been proposed that alveolar recruitment may be responsible for these volume changes, i.e., opening and closing of alveoli (Carney et al., 1999). However, *in situ* studies rather suggest an increase in alveolar size (D'Angelo, 1972). In terms of the model parameters in *Alvin*, both hypotheses manifest themselves in changes in the alveolar surface area available for gas exchange. A surface area of 207,000 μm^2 , measured in inflation-fixed lung tissue (Stone et al., 1992), describes a maximum surface exposure of 100%. The default surface area setting in *Alvin* is 121,000 μm^2 and thus corresponds to an exposure of 58%. This value was taken from a study in which the tissue was perfusion fixed (Mercer et al., 1994). Capillary recruitment in *Alvin* is reflected in capillary blood volume, for which the default value 404,000 μm^3 represents 50% recruitment. By mimicking the ratios of capillary recruitment and alveolar surface area in *Alvin*, one can directly trace the effect on D_{LO_2} . 100% alveolar surface exposure and 100% capillary recruitment in *Alvin* yield a D_{LO_2} of 200 ml/(mmHg × min). 58% alveolar surface exposure and 50% capillary recruitment result in a D_{LO_2} of 61 ml/(mmHg × min).

Alveolar surface area and capillary recruitment impact D_{LO_2} estimates almost linearly (Figure 5B). Additionally, it is interesting to observe their synergistic effect, as ventilation and perfusion are regulated to match (reviewed in (Wagner, 1981; Petersson and Glenny, 2014)). Parallel increase of both alveolar surface exposure and capillary recruitment lead to a non-linear increase in D_{LO_2} , slowly at first and then more rapidly. Consistently, anti-parallel combination of these factors yields generally low D_{LO_2} estimates, with a peak at 50% each. Quantification of this relationship in *Alvin* can be used to interpret other data from the literature. For instance, D_{LO_2} has been estimated from measurements of D_{LCO} and pulmonary blood flow (Kulish, 2006). To recreate these estimates, pulmonary blood flow, expressed in volume per unit time, was

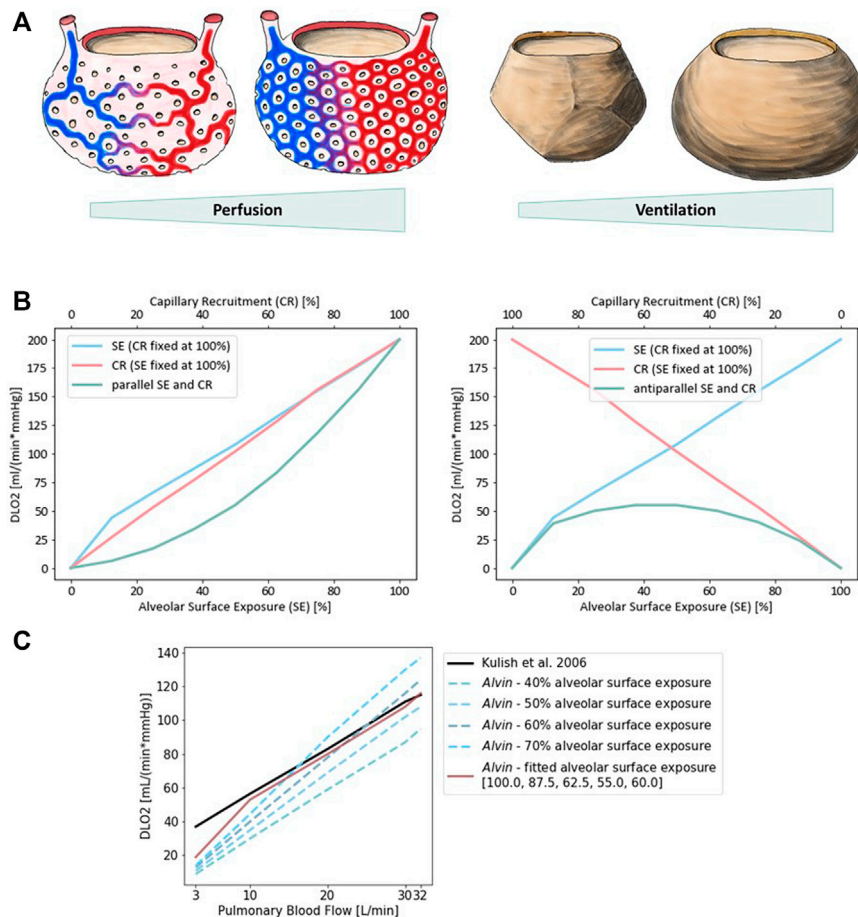


FIGURE 5 | Diffusion capacity of the lung for oxygen (D_{LO_2}) strongly depends on perfusion and ventilation. **(A)** Illustration of capillary recruitment (left) and alveolar expansion (right). **(B)** Diffusion capacity of the lung for oxygen (D_{LO_2}) depending on capillary recruitment and alveolar expansion for a parallel (left) and antiparallel combination (right). Alveolar expansion and the ensuing surface exposure are simulated in *Alvin* by increasing alveolar surface area from 0 (0%) to 207,000 μm^2 (100%) in steps of 12.5%. Capillary recruitment is represented by capillary blood volume increase from 0 (0%) to 808,000 μm^3 (100%) in steps of 12.5% in *Alvin*. **(C)** Comparison to published D_{LO_2} estimates (Kulish, 2006) (black). Pulmonary blood flow was interpreted as blood volume in *Alvin*, assuming a flow velocity of 1.5 mm/s and morphological features (mean capillary length of 500 μm (Weibel et al., 1993) and maximum volume of alveolar capillary bed 808,000 μm^3 (Gehr et al., 1978; Ochs et al., 2004)). Alveolar surface exposure was fixed at constant values (blue dashed lines) and adjusted with increasing pulmonary blood flow (red line).

interpreted as alveolar blood volume in *Alvin*. Assuming a constant blood flow velocity of 1.5 mm/s, the alveolar blood volume was obtained from the mean capillary length of 500 μm (Weibel et al., 1993) and the maximum volume of alveolar capillary bed 808,000 μm^3 (Ochs et al., 2004; Gehr et al., 1978). Under these conditions, D_{LO_2} was determined in *Alvin* with varying alveolar surface area settings (Figure 5C). The resulting D_{LO_2} graphs all differed in slope from the published data (Kulish, 2006). Thus, Kulish's predictions did not appear to have been based on constant alveolar surface exposure. By adjusting alveolar surface area values (100, 87.5, 62.5, 55.0 and 60% surface exposure) along with increasing blood flow (3, 10, 20, 30 and 32 L/min), the results could finally be reconstructed. This fitting was not successful at very low blood flow values.

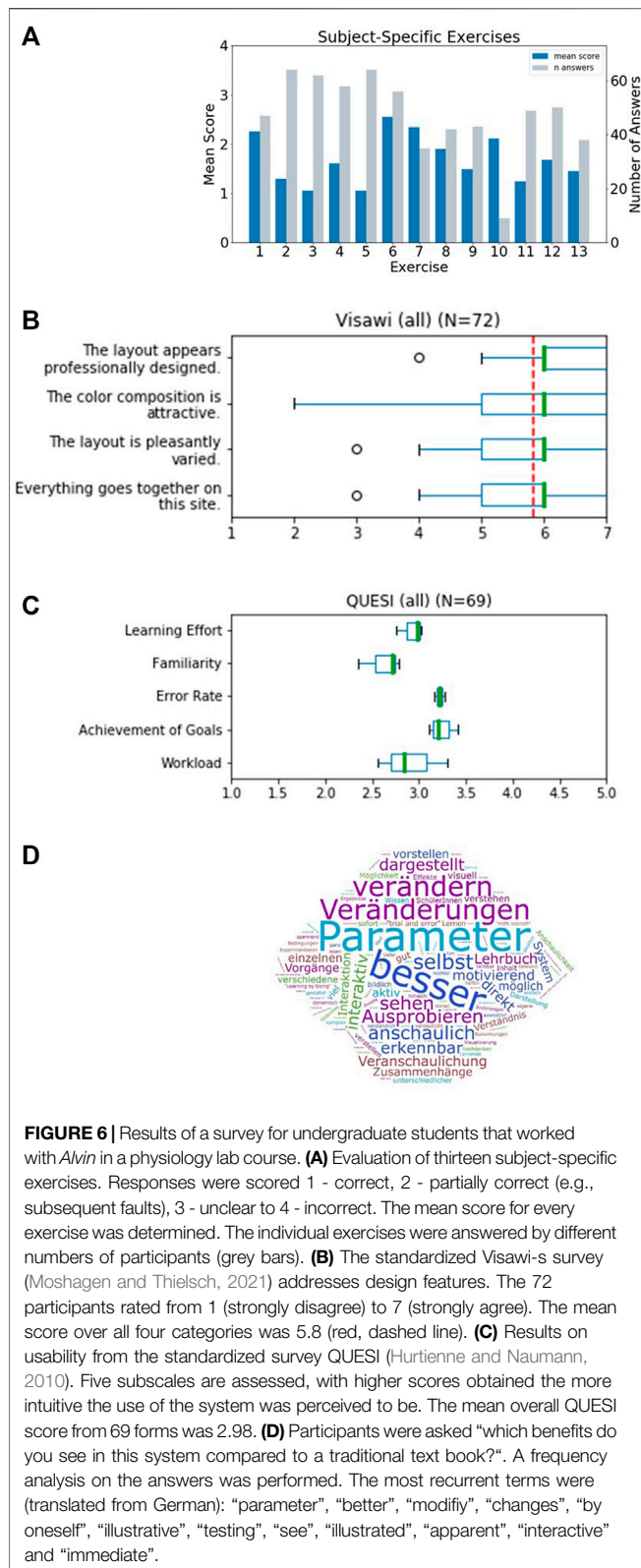
This is only one example of how to employ *Alvin* to investigate correlations in a broader sense or to reproduce data from the literature to gain further insight. Further questions could address the kinetics of gas exchange. One possibility would be to

investigate the threshold conditions under which the blood is still sufficiently oxygenated within the transit time.

3.3.2 *Alvin* in Higher Education: Physiology Lab Course

For application in teaching, the benefits of an interactive simulation have been perceived and exploited since the 1980s (Dewhurst et al., 1988; Davis and Mark, 1990) and are still being pursued today (Jacob et al., 2012; Tworek et al., 2013). Therefore, we integrated *Alvin* into a university level class on human biology, specifically an online practical session on blood and respiration. *Alvin* was used to support the online session by providing an interactive model of the cooperation of the bloodstream and the respiratory system. The suitability of *Alvin* for this course was measured with an online questionnaire.

The course was scheduled for 2 h and 45 min. The participants consisted of students of teaching Biology, specifically of the German levels of *Grundschule* (elementary school/grades 1–4,



mostly third year students), *Mittelschule* (secondary school/grades 5–8, mostly third year students) and *Gymnasium* (grammar school/grades 5–13, mostly fifth year students).

After an introduction into the topic “Blood and Respiration” in the form of a 45 min lecture, *Alvin* was presented briefly, explaining how to use the application and interpret the 3D model and graphs. Participants were given a few minutes to familiarize themselves with *Alvin*. They were then asked for feedback as they worked with the application. An online questionnaire was provided to collect responses. Participation was voluntary and could be withdrawn throughout the event. Submitting the questionnaire as a whole, or answering individual questions, was not mandatory. The questionnaire was split in four parts. The entire questionnaire, translated from German, can be found in the Supplementary Material (**Supplementary Section S2.1**).

The first part consisted of a generic demographic questionnaire, extended by specific questions to assess the formal background of the students and their experience with the subject. We received $N = 73$ valid submissions which were at least partially answered. Of the $N = 73$ surveys received, 11 self-identified as male, 56 as female. The participants all had some prior knowledge of respiratory physiology acquired in a physiology lecture in the previous semester and/or in school or training. In this lecture, basics about the structure and physiology of the lungs as well as the functions of the blood as a transporter of respiratory gases were explained. About half of the group ($N = 34$) could be assumed to have even deeper prior knowledge, as they stated that they had studied further literature in addition to the lecture in question. Participants could be divided into groups with prior knowledge level 1 and 2 accordingly. None of the participants reported being affected by color blindness. The second part contained 13 different exercises addressing respiratory processes in the alveolus. These exercises provided instructions on how to integrate *Alvin* into solution approaches. Among other things, these exercises highlighted well-known relationships and phenomena such as the Bohr effect (Riggs, 1988). Responses were rated on a scale of 1–4 (with 1 indicating perfect answers). The individual exercises were answered by different numbers of participants (**Figure 6A**). Exercise 7 and 10 were answered by less than half of the participants and were therefore not included in the mean overall score of 1.6. Participants with prior knowledge of level 1 performed similarly well to participants with prior knowledge of level 2 (**Supplementary Figure S1**).

The third part consisted of two standardized questionnaires to assess the visual aesthetics and the usability of the application: Visawi-s (Visual Aesthetics of Websites Inventory- short version) (Moshagen and Thielsch, 2021) and QUESI (Questionnaire for Measuring the Subjective Consequences of Intuitive Use) (Hurtienne and Naumann, 2010). Visawi-s (Moshagen and Thielsch, 2021) captures four central aspects of aesthetics from the user's perspective: simplicity, diversity, colorfulness and craftsmanship. Participants were presented with statements targeting these four aspects. They rated them on a scale from 1 (strongly disagree) to 7 (strongly agree). The mean overall ($N = 72$) Visawi-s score was 5.8 (see **Figure 6B**). The standardized QUESI provided a measure of usability (Hurtienne and Naumann, 2010). It is based on the assumption that intuitive use is the unconscious application of prior knowledge leading to effective interaction. It can be divided into the following subscales: Subjective mental workload, perceived achievement

of goals, perceived effort of learning, familiarity, and perceived error rate. The total score of the questionnaire is equal to the mean across all five subscales. Generally, higher scores represent a higher probability of intuitive use. Participants' ($N = 69$) assessments of the use of *Alvin* resulted in a QUESI score of 2.98 (Figure 6C). Published benchmark values for mobile devices and applications (Naumann and Hurtienne, 2010) range from 2.39 (Alcatel One Touch 311) to 4.23 (Nintendo Wii). Familiar products generally perform better in the QUESI (Naumann and Hurtienne, 2010). Hence, participants' prior experience with similar systems in a broader sense, for example, with computer games in general, is important. The majority of our participants ($N = 59$) reported rarely (yearly to never) playing computer games. The minority ($N = 29$) reported using computer games frequently (monthly to daily).

Finally, the questionnaire included customized questions on the use of *Alvin* (evaluation can be found in **Supplementary Section S2.2**) and free-form questions aimed at the acceptance of the software in the educational context. One of them was "Which benefits do you see in this system compared to a traditional text book?". A frequency analysis on answers revealed the highest recurrence for the terms "parameter", "better", "modify", "changes", "by oneself", "illustrative", "testing", "see", "illustrated", "apparent", "interactive" and "immediate" (Figure 6D). A question asking for general feedback was responded to in part with constructive criticism. In particular, it was noted that the content of *Alvin* and the subject-specific tasks were too complex for this introductory event. Or that more time would have been necessary to familiarize oneself with the application. In addition, some reported problems switching between the German lecture content and the English-language application. The participants solved the subject-specific exercises for the most part correctly. It can thus be concluded that *Alvin* is suitable to assist in solving such tasks. Responses to free-text questions suggest which aspects of working with *Alvin* stood out as particularly positive. These include the possibility to interact with the simulation by configuring model parameters and the freedom to independently test different conditions. It was also perceived positively that the simulated processes are presented very illustratively in *Alvin*.

3.3.3 Discussion of Use Cases

Our exemplary use cases show the applicability of *Alvin* in research and in education. We showed an investigation of the dependencies of D_{LO_2} on surface area and blood flow in *Alvin*. Physiological estimates often only consider information about blood flow (Kulish, 2006). By reproducing these estimates in *Alvin*, one can draw conclusions about the alveolar surface. At particularly low blood flow values, it is not possible to reproduce the physiological estimates for D_{LO_2} in *Alvin*. This could have different causes. In the logic of the model and the definition of D_{LO_2} , it is ensured that D_{LO_2} is zero when the blood volume is zero. The physiological estimates in (Kulish, 2006) do not seem to meet this criterion. (note: One cannot be certain, however, because in Kulish et al. (Kulish, 2006) the lowest reported value for blood flow is 3 L/min). It is possible that our model does not produce reliable results in the range of low blood volume

values. Another possibility is that the derivation of D_{LO_2} from D_{LCO} is not reliable in low ranges. This plausibility check shows how *Alvin* can be used to support or challenge published data. Drawing on known relationships, additional information can be obtained from previous results.

We also showed that *Alvin* is helpful for communicating respiratory processes in the training of undergraduate students. Well-known processes or phenomena like the Bohr-Effect (Riggs, 1988) can be recreated in *Alvin* and compared with results reported in the literature. Interactivity of the simulation enables experimentation with the model and exploration of its limitations. This aspect was also positively highlighted by participants of the physiology lab course in free-form answers of our questionnaire. The results of the QUESI and VISAWI questionnaires on their own do not allow for quantitative conclusions on usability or aesthetics of the application. This would require comparing them to corresponding results from comparable test situations (for example, about similar systems). At this point, one can only state that the replies did not hint at unknown issues. Instead, they were aligned with our expectations that participants should be able to operate the system autonomously and find its use appealing and relatively intuitive.

In summary, the integration of *Alvin* into physiology classes at the university level was successful. Beyond that, issues were pointed out where the implementation could be optimized in the future. Prominent and consistent were requests for more time to engage with *Alvin*. We deliberately refrained from providing the application to the participants in advance of this course to avoid a mutual influence of the participants regarding their experience with *Alvin*. This was important for the evaluation with the standardized questionnaires. For general use in teaching, however, this does not have to be taken into account. On the contrary, an exchange between students about the system could increase its learning value. We conclude that *Alvin* is less suitable to be included in a single physiology lesson. Instead, we recommend that students be made aware of the app ahead of time or to invest several course sessions.

4 CONCLUSION AND OUTLOOK

Interactive, visual simulations allow communicating modeling results and thereby help to further our understanding of the process under study. We presented *Alvin*, an application for simulating gas exchange in a single alveolus. The simulation is based on a mathematical model for the entire transport process of oxygen from the air to hemoglobin of the blood. We claim that having the goal of an interactive, visual simulation in mind when developing a mathematical model is beneficial for the modeling process. It resulted in a specific requirement for the model: In order to be able to map the course of the simulation on a three-dimensional tissue model, it had to be temporally and spatially resolved. Models evolve by being revised and improved over and over again (Drubin and Oster, 2010). If one assumes that a model can be better developed the more experts review it, then it is advantageous to make the model freely and intuitively accessible.

We argue that interactive visualization offers an engaging way to communicate theoretical models to other scientists and students. When cooperating with experimenters, it is important for theorists to present their models in the most accessible way possible. This creates as large a basis for discussion as possible in order to jointly plan further experiments or model refinements. By making model parameters intuitively configurable, any experimenter can compare his or her own measurements with the modeling results. By including undergraduate students in the target group for *Alvin*, we ensured that only a minimum of prior knowledge is required for its usage.

In the future, we plan to extend our model to encompass a system of multiple alveoli and their associated vessels. This will allow us to address further questions and complex relationships regarding gas exchange in lung tissue. It is known that the ventilation-perfusion relationship, and therefore the diffusion-perfusion relationship, has a strong influence on D_{LO_2} (Hyde et al., 1967; Hammond and Hempleman, 1987). An evolution of *Alvin* that includes an alveolar sac or a whole acinus with differently ventilated and perfused alveoli can provide valuable insights. This could also be used, for example, to further investigate the hypothesis of precapillary oxygen uptake (Tabuchi et al., 2013). It states that the oxygenation process already takes place in the precapillary arterioles before the blood reaches the alveolar capillary bed.

Rather than just presenting the data that results from a newly developed model, it is worthwhile to implement the model in a way that allows for interaction. Visualizing the simulation makes the engagement with the model more intuitive and accessible to a broader target group. Empiricists and theorists look at a system from different angles. Some work in a bottom-up fashion and take local samples and draw conclusions for the overall system. Others create abstract models for the overall system top-down and try to approach the truth by introducing more and more details. Only by working closely together can these two perspectives efficiently contribute to reliable results and become a “middle-out” approach (Noble, 2008). The communication of the achieved findings or predictions plays an important role here. We contend that interactive, visual simulations of theoretical models, as we have implemented with *Alvin* on respiratory processes in the alveolus, will make an important contribution to bridging the gap between empiricists and theorists.

REFERENCES

- Bachofen, H., Schürch, S., Urbinelli, M., and Weibel, E. R. (1987). Relations Among Alveolar Surface Tension, Surface Area, Volume, and Recoil Pressure. *J. Appl. Physiol.* (1985) 62, 1878–1887. doi:10.1152/jappl.1987.62.5.1878
- Byrne, H. M., Alarcon, T., Owen, M. R., Webb, S. D., and Maini, P. K. (2006). Modelling Aspects of Cancer Dynamics: a Review. *Philos. Trans. A. Math. Phys. Eng. Sci.* 364, 1563–1578. doi:10.1098/rsta.2006.1786
- Carney, D. E., Bredenberg, C. E., Schiller, H. J., Picone, A. L., McCANN, U. G., Gatto, L. A., et al. (1999). The Mechanism of Lung Volume Change during Mechanical Ventilation. *Am. J. Respir. Crit. Care Med.* 160, 1697–1702. doi:10.1164/ajrcm.160.5.9812031
- Chao, E. Y. (2003). Graphic-based Musculoskeletal Model for Biomechanical Analyses and Animation. *Med. Eng. Phys.* 25, 201–212. doi:10.1016/S1350-4533(02)00181-9

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/**Supplementary Material**, further inquiries can be directed to the corresponding author. We provide the source code of *Alvin* at <https://github.com/scfischer/schmid-et-al-2022>.

ETHICS STATEMENT

Ethical review and approval was not required for the study on human participants in accordance with the local legislation and institutional requirements. The patients/participants provided their written informed consent to participate in this study.

AUTHOR CONTRIBUTIONS

Alvin concept and design: AK, AM, KS. Implementation of *Alvin*: AK, AM, KS. Model development and validation: KS. Planning and supervision of the use case in teaching: KP, AK, KS. Demonstration of possible application in research: KS. Supervision: SF, SvM, KP. Manuscript preparation: KS, SF, AK, SvM. All authors contributed to the article and approved the submitted version.

ACKNOWLEDGMENTS

We thank Andreas Hocke and Katja Hönzke for inspiring discussions and support of the project. We thank Wolfgang Kübler and Matthias Ochs for valuable feedback on *Alvin* and members of the CCTB for testing *Alvin*. KS and SCF acknowledge the support by a grant from Universitätsbund Würzburg (AZ21-16).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fbinf.2021.774300/full#supplementary-material>

- Conover, T., Hlavacek, A. M., Migliavacca, F., Kung, E., Dorfman, A., Figliola, R. S., et al. (2018). An Interactive Simulation Tool for Patient-specific Clinical Decision Support in Single-Ventricle Physiology. *J. Thorac. Cardiovasc. Surg.* 155, 712–721. doi:10.1016/j.jtcvs.2017.09.046
- Costabile, M. (2021). Design, Implementation, and Assessment of an Interactive Simulation to Teach Undergraduate Immunology Students Hemolytic Disease of the Newborn. *Adv. Physiol. Educ.* 45, 299–306. doi:10.1152/advan.00008.2021
- Crapo, J. D., and Crapo, R. O. (1983). Comparison of Total Lung Diffusion Capacity and the Membrane Component of Diffusion Capacity as Determined by Physiologic and Morphometric Techniques. *Respir. Physiol.* 51, 183–194. doi:10.1016/0034-5687(83)90039-7
- D’Angelo, E. (1972). Local Alveolar Size and Transpulmonary Pressure *In Situ* and in Isolated Lungs. *Respiration Physiol.* 14, 251–266. doi:10.1016/0034-5687(72)90032-1

- Dash, R. K., Korman, B., and Bassingthwaite, J. B. (2016). Simple Accurate Mathematical Models of Blood HbO₂ and HbCO₂ Dissociation Curves at Varied Physiological Conditions: Evaluation and Comparison with Other Models. *Eur. J. Appl. Physiol.* 116, 97–113. doi:10.1007/s00421-015-3228-3
- Davis, T. L., and Mark, R. G. (1990). Teaching Physiology through Simulation of Hemodynamics. *Proc. Comput. Cardiol.* 1990, 649–652. doi:10.1109/CIC.1990.144303
- Dewhurst, D. G., Brown, G. J., and Meehan, A. S. (1988). Microcomputer Simulations of Laboratory Experiments in Physiology. *Altern. Lab. Anim.* 15, 280–289. doi:10.1177/026119298801500403
- Drubin, D. G., and Oster, G. (2010). Experimentalist Meets Theoretician: A Tale of Two Scientific Cultures. *Mol. Biol. Cell* 21, 2099–2101. doi:10.1091/mbc.E10-02-0143
- Fischer, S. C. (2019). “An Introduction to Image-Based Systems Biology of Multicellular Spheroids for Experimentalists and Theoreticians,” in *Computational Biology*. Editor H. Husi (Brisbane, AU: Codon Publications). doi:10.15586/computationalbiology.2019.ch1
- Forster, R. E. (1964). “Diffusion of Gases,” in *Handbook of Physiology* (Baltimore, MD, USA: Waverly Press), 839872.
- Gavaghan, D., Garny, A., Maini, P. K., and Kohl, P. (2006). Mathematical Models in Physiology. *Philos. Trans. A. Math. Phys. Eng. Sci.* 364, 1099–1106. doi:10.1098/rsta.2006.1757
- Gehr, P., Bachofen, M., and Weibel, E. R. (1978). The normal Human Lung: Ultrastructure and Morphometric Estimation of Diffusion Capacity. *Respir. Physiol.* 32, 121–140. doi:10.1016/0034-5687(78)90104-4
- Gil, J., Bachofen, H., Gehr, P., and Weibel, E. R. (1979). Alveolar Volume-Surface Area Relation in Air- and saline-filled Lungs Fixed by Vascular Perfusion. *J. Appl. Physiol. Respir. Environ. Exerc. Physiol.* 47, 990–1001. doi:10.1152/jappl.1979.47.5.990
- Haefeli-Bleuer, B., and Weibel, E. R. (1988). Morphometry of the Human Pulmonary Acinus. *Anat. Rec.* 220, 401–414. doi:10.1002/ar.1092200410
- Hammond, M. D., and Hempleman, S. C. (1987). Oxygen Diffusing Capacity Estimates Derived from Measured VA/Q Distributions in Man. *Respir. Physiol.* 69, 129–147. doi:10.1016/0034-5687(87)90022-3
- He, C., Micallef, L., He, L., Peddinti, G., Aittokallio, T., and Jacucci, G. (2021). Characterizing the Quality of Insight by Interactions: A Case Study. *IEEE Trans. Vis. Comput. Graph* 27, 3410–3424. doi:10.1109/TVCG.2020.2977634
- Hsia, C. C. W., Hyde, D. M., and Weibel, E. R. (2016). Lung Structure and the Intrinsic Challenges of Gas Exchange. *Compr. Physiol.*, 827–895. doi:10.1002/cphy.c150028
- Hurtienne, J., and Naumann, A. (2010). “QUESI – A Questionnaire For Measuring The Subjective Consequences Of Intuitive Use,” in *Interdisciplinary College 2010. Focus Theme: Play, Act and Learn*. Editors R. Porzel, N. Sebanz, and M. Spitzer (Sankt Augustin: Fraunhofer Gesellschaft), 536.
- Hyde, R. W., Rynes, R., Power, G. G., and Nairn, J. (1967). Determination of Distribution of Diffusing Capacity in Relation to Blood Flow in the Human Lung. *J. Clin. Invest.* 46, 463–474. doi:10.1172/JCI105548
- Jacob, C., von Mammen, S., Davison, T., Sarraf-Shirazi, A., Sarpe, V., Esmaeili, A., et al. (2012). “LINDSAY Virtual Human: Multi-Scale, Agent-Based, and Interactive,” in *Advances in Intelligent Modelling and Simulation: Artificial Intelligence-Based Models and Techniques in Scalable Computing. Studies in Computational Intelligence*. Editors J. Kołodziej, S. U. Khan, and T. Burczyński (Berlin, Heidelberg: Springer), 327–349. doi:10.1007/978-3-642-30154-4_14
- Jamniczky, H., Jacob, C., Novakowski, S., Davison, T., Mammen, S. v., Gingras, C., et al. (2012). The LINDSAY Virtual Human Project: Anatomy and Physiology Come to Life. *FASEB J.* 26, lb28. doi:10.1096/fasebj.26.1_supplement.lb28
- Kapitan, K. S. (2008). Teaching Pulmonary Gas Exchange Physiology Using Computer Modeling. *Adv. Physiol. Educ.* 32, 61–64. doi:10.1152/advan.00099.2007
- Kuebler, W. M., Mertens, M., and Pries, A. R. (2007). A Two-Component Simulation Model to Teach Respiratory Mechanics. *Adv. Physiol. Educ.* 31, 218–222. doi:10.1152/advan.00001.2007
- Kulish, V. (2006). *Human Respiration: Anatomy and Physiology, Mathematical Modeling, Numerical Simulation and Applications*. Chilworth, Southampton, USA: WIT Press. Google-Books-ID: J2rQCwAAQBAJ.
- Lin, D. W., Johnson, S., and Hunt, C. A. (2004). Modeling Liver Physiology: Combining Fractals, Imaging and Animation. *Conf. Proc. IEEE Eng. Med. Biol. Soc.* 2, 3120–3123. doi:10.1109/IEMBS.2004.1403881
- Lindstedt, S. L. (1984). Pulmonary Transit Time and Diffusing Capacity in Mammals. *Am. J. Physiol.* 246, R384–R388. doi:10.1152/ajpregu.1984.246.3.R384
- Mercer, R. R., Russell, M. L., and Crapo, J. D. (1994). Alveolar Septal Structure in Different Species. *J. Appl. Physiol.* (1985) 77, 1060–1066. doi:10.1152/jappl.1994.77.3.1060
- Mogilner, A., Edelstein-Keshet, L., and Bloom, K. (2011). Guidelines for Publishing Papers Containing Theory and Modeling. *MBoC* 22, 907–908. doi:10.1091/mbc.E11-01-0088
- Moshagen, M., and Thielsch, M. T. (2010). Facets of Visual Aesthetics. *Int. J. Human-Computer Stud.* 68, 689–709. doi:10.1016/j.ijhcs.2010.05.006
- Mühlfeld, C., Weibel, E. R., Hahn, U., Kummer, W., Nyengaard, J. R., and Ochs, M. (2010). Is Length an Appropriate Estimator to Characterize Pulmonary Alveolar Capillaries? A Critical Evaluation in the Human Lung. *Anat. Rec. (Hoboken)* 293, 1270–1275. doi:10.1002/ar.21158
- Naumann, A., and Hurtienne, J. (2010). “Benchmarks for Intuitive Interaction with mobile Devices,” in *Proceedings of the 12th International Conference on Human Computer Interaction with mobile Devices and Services - MobileHCI '10* (Lisbon, Portugal: ACM Press), 401. doi:10.1145/1851600.1851685
- Noble, D. (2008). “The Orchestra: Organs and Systems of the Body,” in *The Music of Life: Biology beyond Genes* (Oxford, United Kingdom: OUP Oxford), 74–87.
- Ochs, M., Nyengaard, J. R., Jung, A., Knudsen, L., Voigt, M., Wahlers, T., et al. (2004). The Number of Alveoli in the Human Lung. *Am. J. Respir. Crit. Care Med.* 169, 120–124. doi:10.1164/rccm.200308-1107OC
- Okada, O., Presson, R. G., Jr, Kirk, K. R., Godbey, P. S., Capen, R. L., and Wagner, W. W., Jr (1992). Capillary Perfusion Patterns in Single Alveolar walls. *J. Appl. Physiol.* (1985) 72, 1838–1844. doi:10.1152/jappl.1992.72.5.1838
- Pagana, K., Pagana, T., and Pagana, T. (2019). *Mosby's Diagnostic and Laboratory Test Reference*. 14 edn. Amsterdam, Netherlands: Elsevier.
- Petersson, J., and Glenn, R. W. (2014). Gas Exchange and Ventilation-Perfusion Relationships in the Lung. *Eur. Respir. J.* 44, 1023–1041. doi:10.1183/09031936.00037014
- Pike, W. A., Stasko, J., Chang, R., and O'Connell, T. A. (2009). The Science of Interaction. *Inf. Visualization* 8, 263–274. doi:10.1057/ivs.2009.22
- Powers, K. A., and Dhamoon, A. S. (2019). “Physiology, Pulmonary, Ventilation and Perfusion,” in *StatPearls* (Treasure Island, FL: StatPearls Publishing).
- Riggs, A. F. (1988). The Bohr Effect. *Annu. Rev. Physiol.* 50, 181–204. doi:10.1146/annurev.ph.50.030188.001145
- Saber, E. M., and Heydari, G. (2012). Flow Patterns and Deposition Fraction of Particles in the Range of 0.1–10µm at Trachea and the First Third Generations under Different Breathing Conditions. *Comput. Biol. Med.* 42, 631–638. doi:10.1016/j.compbiomed.2012.03.002
- Sapoval, B., Kang, M. Y., and Dinh-Xuan, A. T. (2020). Modeling of Gas Exchange in the Lungs. *Compr. Physiol.* 11, 1289–1314. doi:10.1002/cphy.c190019
- Sharma, S., Hashmi, M. F., and Burns, B. (2020). “Alveolar Gas Equation,” in *StatPearls* (Treasure Island, FL: StatPearls Publishing).
- Sharpe, J. (2017). Computer Modeling in Developmental Biology: Growing Today, Essential Tomorrow. *Development* 144, 4214–4225. doi:10.1242/dev.151274
- Stone, K. C., Mercer, R. R., Gehr, P., Stockstill, B., and Crapo, J. D. (1992). Allometric Relationships of Cell Numbers and Size in the Mammalian Lung. *Am. J. Respir. Cell Mol Biol* 6, 235–243. doi:10.1165/ajrcmb/6.2.235
- Tabuchi, A., Styp-Rekowska, B., Slutsky, A. S., Wagner, P. D., Pries, A. R., and Kuebler, W. M. (2013). Precapillary Oxygenation Contributes Relevantly to Gas Exchange in the Intact Lung. *Am. J. Respir. Crit. Care Med.* 188, 474–481. doi:10.1164/rccm.201212-2177OC
- Tworek, J. K., Jamniczky, H. A., Jacob, C., Hallgrímsson, B., and Wright, B. (2013). The LINDSAY Virtual Human Project: An Immersive Approach to Anatomy and Physiology. *Anat. Sci. Ed.* 6, 19–28. doi:10.1002/ase.1301
- Viceconti, M., Clapworthy, G., and Van Sint Jan, S. (2008). The Virtual Physiological Human - a European Initiative for In Silico Human Modelling. *J. Physiol. Sci.* 58, 441–446. doi:10.2170/physiolsci.RP009908
- Wagner, P. D. (1981). Ventilation/perfusion Relationships. *Clin. Physiol.* 1, 437–451. doi:10.1111/j.1475-097x.1981.tb00911.x

- Warlia, L., Rohman, A. S., and Rusmin, P. H. (2012). Model Development of Air Volume and Breathing Frequency in Human Respiratory System Simulation. *Proced. - Soc. Behav. Sci.* 67, 260–268. doi:10.1016/j.sbspro.2012.11.328
- Weibel, E. R., Federspiel, W. J., Fryder-Doffey, F., Hsia, C. C., König, M., Stalder-Navarro, V., et al. (1993). Morphometric Model for Pulmonary Diffusing Capacity. I. Membrane Diffusing Capacity. *Respir. Physiol.* 93, 125–149. doi:10.1016/0034-5687(93)90001-Q
- Weibel, E. R., and Gomez, D. M. (1962). Architecture of the Human Lung. Use of Quantitative Methods Establishes Fundamental Relations between Size and Number of Lung Structures. *Science* 137, 577–585. doi:10.1126/science.137.3530.577
- Weibel, E. R. (1970). Morphometric Estimation of Pulmonary Diffusion Capacity. I. Model and Method. *Respir. Physiol.* 11, 54–75. doi:10.1016/0034-5687(70)90102-7
- Weibel, E. R. (2009). What Makes a Good Lung? *Swiss Med. Wkly* 139, 375–386. smw-12270.
- Welsh, E., Jirotko, M., and Gavaghan, D. (2006). Post-genomic Science: Cross-Disciplinary and Large-Scale Collaborative Research and its Organizational and Technological Challenges for the Scientific Research Process. *Philos. Trans. A. Math. Phys. Eng. Sci.* 364, 1533–1549. doi:10.1098/rsta.2006.1785
- Winkler, T., Krause, A., and Kaiser, S. (1995). Simulation of Mechanical Respiration Using a Multicompartment Model for Ventilation Mechanics and Gas Exchange. *Int. J. Clin. Monit. Comput.* 12, 231–239. doi:10.1007/BF01207204
- Xiong, G., Sun, P., Zhou, H., Ha, S., Hartaigh, B. O., Truong, Q. A., et al. (2017). Comprehensive Modeling and Visualization of Cardiac Anatomy and Physiology from CT Imaging and Computer Simulations. *IEEE Trans. Vis. Comput. Graph* 23, 1014–1028. doi:10.1109/TVCG.2016.2520946

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Schmid, Knot, Mück, Pfeiffer, von Mammen and Fischer. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Naview: A d3.js Based JavaScript Library for Drawing and Annotating Voltage-Gated Sodium Channels Membrane Diagrams

Marcelo Querino Lima Afonso^{1*†‡}, Néli José da Fonseca Júnior^{2‡}, Thainá Godinho Miranda¹ and Lucas Bleicher^{1*}

OPEN ACCESS

Edited by:

Lydia Gregg,
Johns Hopkins University,
United States

Reviewed by:

Daniel Haehn,
University of Massachusetts Boston,
United States
Bjorn Sommer,
Royal College of Art, United Kingdom

*Correspondence:

Marcelo Querino Lima Afonso
marceloqla@ufmg.br
Lucas Bleicher
bleicher@ufmg.br

†Present address:

Marcelo Querino Lima Afonso,
Departamento de Bioquímica e
Imunologia, Instituto de Ciências
Biológicas, Universidade Federal de
Minas Gerais, Belo Horizonte, Brazil

[‡]These authors have contributed
equally to this work

Specialty section:

This article was submitted to
Data Visualization,
a section of the journal
Frontiers in Bioinformatics

Received: 11 September 2021

Accepted: 05 January 2022

Published: 11 February 2022

Citation:

Afonso MQL, da Fonseca Júnior NJ,
Miranda TG and Bleicher L (2022)
Naview: A d3.js Based JavaScript
Library for Drawing and Annotating
Voltage-Gated Sodium Channels
Membrane Diagrams.
Front. Bioinform. 2:774417.
doi: 10.3389/fbinf.2022.774417

¹Departamento de Bioquímica e Imunologia, Instituto de Ciências Biológicas, Universidade Federal de Minas Gerais, Belo Horizonte, Brazil, ²Cellular Structure and 3D Bioimaging, European Molecular Biology Laboratory, European Bioinformatics Institute, Hinxton, United Kingdom

Voltage-gated sodium channels (Nav) are membrane proteins essential to initiating and propagating action potential in neurons and other excitable cells. For a given organism there are often multiple, specialized sodium channels found in different tissues, whose mutations can cause deleterious effects observed in numerous diseases. Consequently, there is high medical and pharmacological interest in these proteins. Scientific literature often uses membrane diagrams to depict important patterns in these channels including the six transmembrane segments (S1–S6) present in four different homologous domains (D1–D4), the S4 voltage sensors, the pore-lining residue segments and the ion selectivity filter residues, glycosylation and phosphorylation residues, toxin binding sites and the inactivation loop, among others. Most of these diagrams are illustrated either digitally or by hand and programs specifically dedicated to the interactive and data-friendly generation of such visualizations are scarce or non-existing. This paper describes Naview, an open-source javascript visualization compatible with modern web browsers for the dynamic drawing and annotation of voltage-gated sodium channels membrane diagrams based on the D3.js library. By using a graphical user interface and combining user-defined annotations with optional UniProt code as inputs, Naview allows the creation and customization of membrane diagrams. In this interface, a user can also map and display important sodium channel properties, residues, regions and their relationships through symbols, colors, and edge connections. Such features can facilitate data exploration and provide fast, high-quality publication-ready graphics for this highly active area of research.

Keywords: membrane plot, voltage gated sodium channel (Nav), d3.js, data visualization, javascript

INTRODUCTION

Voltage-gated sodium (Na⁺) channels are key signaling membrane proteins responsible for electrical excitability, also involved in biological processes in non-excitable cells, and of considerable physiological and pharmacological interest (Cardoso and Lewis, 2018). Voltage-gated Na⁺ channels (Navs) can generate and propagate action potentials in excitable cells due to channel opening and fast inactivation mechanisms that regulate the permeation of Na⁺ ions across the

membrane (Capes et al., 2012; Xia et al., 2013; Kubota et al., 2017). These channels are present in a large variety of organisms, the domain architecture of human Navs being observed in all animals. Their dysfunction is involved in severe diseases such as epileptic seizures, migraines cardiac arrhythmias, as well as pain-related neuropathies (Xia et al., 2013; Erickson et al., 2018). Sodium channels are involved in multiple physiological roles within a given organism, including the transmission of somatosensory signals, angiogenesis, muscle contraction, and immune cell maturation (Cardoso and Lewis, 2018). In addition, insect sodium channels are potential targets for both natural and synthetic insecticides and are therefore of agricultural interest (Zhang et al., 2016).

Each channel consists of an alpha subunit and auxiliary beta subunits that modify the properties of the first (Widmark et al., 2011). The alpha subunit is composed of a single chain of four sub-units in tandem (Domains I-IV), each formed by a structure of six transmembrane helices (6TM, H1-H6) that associate as tetramers to form a channel. Small extracellular and intracellular loops connect each helix, and the pore loops and large intracellular loops connect each domain (Yu and Catterall, 2003). In mammals, nine isoforms of these channels are found (Gene names SCN1A-SCN11A) possessing different functional roles, properties, and tissue-specific distributions among cells of the central and peripheral nervous systems (Chowdhury and Chanda, 2019). Post-translational modifications such as glycosylations and phosphorylations are part of the cellular modulation repertoire of these channels *in vivo*, being mostly found within the intracellular loop between the first and second domain of these channels (Scheuer, 2011; Laedermann et al., 2015; Cardoso and Lewis, 2018).

Graphical representations of the Nav alpha subunit transmembrane architecture are widely used in the scientific literature, with the earliest examples dated from the late 1980s—(Tanabe et al., 1988; Trimmer et al., 1989; Chiamvimonvat et al., 1996; Marban et al., 1998; Yu and Catterall, 2003; Yamaoka et al., 2006; Wood and Iseppon, 2018; Zybur et al., 2021). In these diagrams, membranes are shown as rectangles or cylinders, and loops as curved lines. Features commonly described by such plots include the voltage sensing helix S4, the fast inactivation motif IFM, glycosylation and phosphorylation sites, drug binding sites, important mutation sites, relevant sites for subunit interaction, and toxin binding sites. These features are usually displayed as either text or symbols inside the diagram.

Although sodium channel diagrams have been used for over 30 years, the availability of tools dedicated to an automated generation of such plots has been limited, but options for simpler diagrams with varying features are available. TOPO2 (Johns, 2010) reads an input indicating the number of segments in a protein chain, start/finish residues for transmembrane or partially inserted segments and residues to be colored and generates a simplified color diagram. Topology diagrams can also be drawn by using the output of a topology detection software such as HERA (Hutchinson and Thornton, 1990) and feeding it to topology drawing software such as TopDraw (Bond,

2003). This approach can also be used for globular proteins, but does not allow for individual residue/segment annotation, and includes no information about membrane insertion, being restricted to the secondary structure topology obtained from a PDB file. Membrane diagrams with individual annotations can be created using TMRPres2D (Spyropoulos et al., 2004) using user-provided info or importing information about transmembrane boundaries using public databases. The LaTeX based Protter web application (Omasits et al., 2014) and Textopo (Beitz, 2000) are capable of generating membrane protein diagrams in which each residue is displayed as geometric forms (often as circles). Whereas annotations can be easily included in both programs as symbols, text or specific colors, secondary structures cannot be easily distinguished in the diagrams of Protter and Textopo.

Various commercial and open source alternatives dedicated to drawing chemical compounds such as MarvinSketch, ChemDoodle, BKchem, XDrawChem, JChemPaint, ACD/ChemSketch, and MolView often have modules dedicated to the 3D visualization of proteins, but generating 2D diagrams (Krause et al., 2000; Todsen, 2014; Bergwerf, 2015). ChemDraw is one of the few alternatives including the possibility of drawing such diagrams in a highly dynamic and easy-to-use interface but lacking the possibility of direct inclusion of protein related data (Cousins, 2005).

Sodium channels membrane diagrams remain popular despite the increasing deposition of Nav structures in the last years, especially by cryogenic electron microscopy (Ahuja et al., 2015; Pan et al., 2018; Xu et al., 2019; Jiang et al., 2021), and the vast number of software dedicated to the 3D visualization of protein molecules such as PyMOL (Schrödinger, 2015), UCSF Chimera (Pettersen et al., 2004), VMD (Humphrey et al., 1996), Jmol (Jmol development team, 2016), and JavaScript based tools such as 3Dmol (Rego and Koes 2015), iCn3D (Wang et al., 2020), Litemol (Sehna et al., 2017), NGL Viewer (Rose and Hildebrand 2015) and Mol* (Sehna et al., 2021). Often used alongside figures rendered from 3D structures, the persistent usage of Nav diagrams could be attributed to their summarizing capacity. The alpha subunit of Navs often possess a length of more than 1,500 amino acids which can be challenging to depict when their complex topology is taken into account: four domains of six transmembrane helices and a reentrant loop, long and short interdomain loops disposed on either the intra or extracellular faces of the plasma membrane. Due to this the explicit representation of some features could require multiple 3D poses.

This publication describes Naview, an open-source d3.js based JavaScript library for drawing and annotating voltage-gated sodium channels membrane diagrams. Naview can highlight essential Nav features by using custom data provided by the user to modify the text, color, and connecting lines at specific helix/loop elements or residues.

METHODS

Implementation

Naview is implemented as an open-source d3.js based JavaScript web component, which can be used by importing its main CDN

TABLE 1 | Property table example. First column must be formatted with the “Resid” header followed by digits indicating each residue for a property to be mapped. The following property columns have header strings and are followed by float or integer numbers indicating the value of a property for each residue of a Nav.

Resid	Property
1	0.2871809547
2	0.9835970474
3	0.3891381106
4	0.2391246386

file (naview.js) into web pages. The complete documentation of each of the library’s 107 functions and eight global variables can be found at: <http://bioinfo.icb.ufmg.br/naview/public/docs/index.html>. Naview is freely available under the Apache License 2.0. The complete source code and additional information related to library usage can be found at GitHub (<https://github.com/marceloqla/NaView/>). In addition to the web component, Naview can also be used as a web application (<http://bioinfo.icb.ufmg.br/naview/>), developed in PHP, allowing direct access to any sodium channel available in the UniProtKb. Naview Style Editor is a graphical user interface that allows plot customization, the upload of residue mapped properties and residue/element interactions, and the download of the plot figures as Scalable Vector Graphics (SVG) or Portable Network Graphics (PNG). The styling information can also be exported as a text file that can be reused in new diagrams.

Data Input and Processing

Two main inputs are generally supplied to Naview for generating a Nav alpha-subunit diagram (**Figure 1**):

- 1) A mandatory UniProt formatted text string (hereafter named *Raw Text*) containing the required data plotting a Nav alpha subunit. In the web application version, it is automatically

TABLE 2 | Relationship table example. Four columns are allowed with the following headers: “source”, “target”, “raw_weight”, and “type”. First and second columns indicating the interacting residues or elements. The “raw_weight” column contains an edge weight for color or width mapping. The last column “type” can be used to indicate edge types which can be weighted or colored separately.

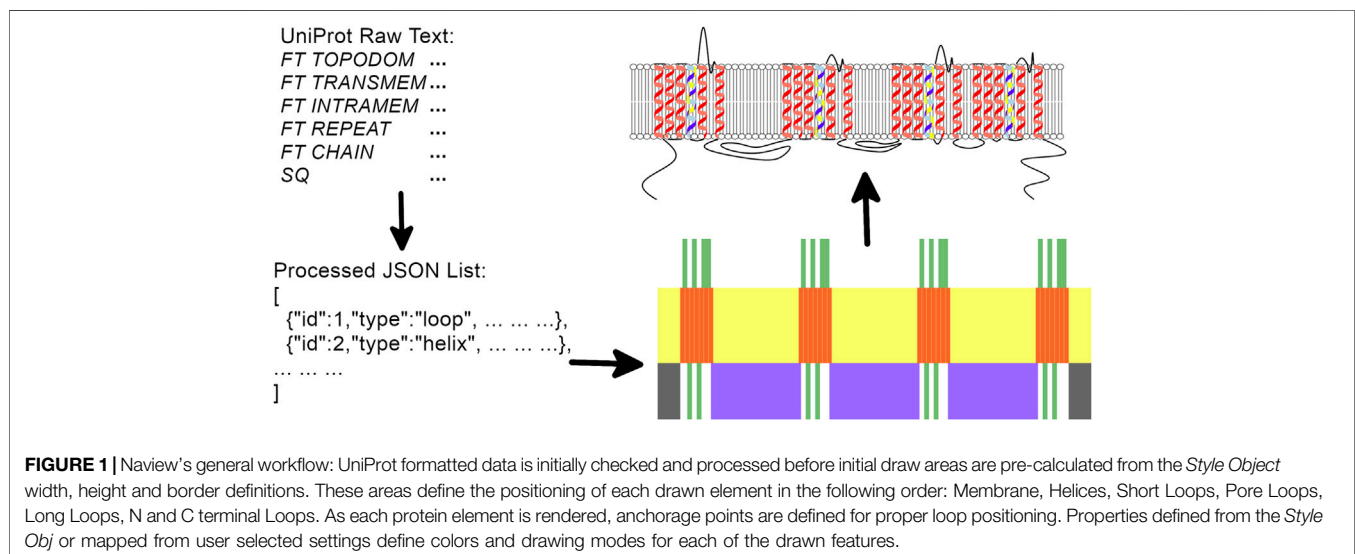
Source	Target	Raw_weight	Type
776	660	0.6944505517	Resids
86	469	0.7383026986	Resids
1,308	318	0.4949883823	Resids
305	510	0.9651479396	Resids
1,621	123	0.3030461658	Resids
DomainI; Helix4	DomainII; Loop4	0.08937180957	Elements
DomainII; Helix4	DomainIV; Helix4	0.9300459795	Elements
InterDomain5; Loop	InterDomain1; Loop	0.1476849439	Elements

fetched from the UniProtKb, requiring only the sequence identifier;

- 2) An optional JavaScript Object Notation (JSON) object, hereafter named *Style Object*, containing information related to the elements plot disposition such as their drawing types, widths, heights, scales, and colors (http://bioinfo.icb.ufmg.br/naview/public/docs/symbols/style_obj.html on the documentation for further information on the *Style Object*). When not supplied, a default representation of the *Style Object* is automatically applied. Any drawing options of the *Style Object* can be modified by Naview Style Editor (**Figure 2**).

Additionally, other inputs related to plotting text, color, and relationship annotations can be supplied. Each of them is described alongside their specific syntax in their dedicated sections.

The UniProt formatted *Raw Text* supplied by the user is then processed for the definition of drawing areas for three possible element types: membrane, helices, and loops which are further



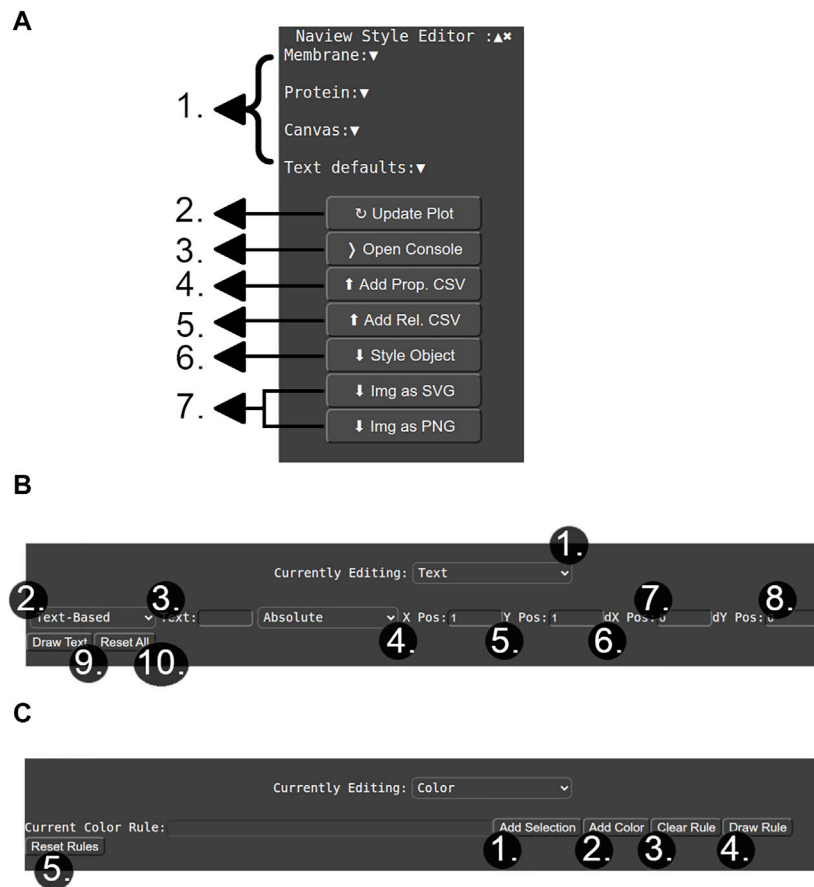


FIGURE 2 | Naview's style editor. **(A)** Main options of the styling menu including: 1) Dropdowns for options related to each of the drawn diagrams features such as colors, sizes and proportions. 2) Button for refreshing the currently drawn plot. 3) Button for opening the console that allows entering specific text annotation or color rules. 4) and 5) are buttons for adding property and relationship related data to the plot. 6) button for exporting a *Style Object* with the currently selected configurations. 7) Buttons for exporting the plot image in the SVG and PNG formats. **(B)** Console for adding a text annotation or color rule. 1) Dropdown for selecting between the text annotation or color rule modes. 2) Dropdown for selecting the input of free/property based text. 3) Input box for typing the desired text annotation. 4) Text positioning scheme: "absolute" defines text position by the given "x" and "y" 5) and 6) parameters; "relative" defines text position according to a selected element. "dx" and "dy" 7) and 8) shift the text to be drawn in the informed horizontal ("dx") and vertical direction ("dy"), being especially helpful in the "relative" positioning scheme. 9) Button for appending the currently defined text to the figure. 10) Removes all added text annotations. **(C)** Color rule addition console. 1) Opens a window for allowing specific residue/elements selections. 2) Opens a window for selecting a specific color/property-based color mapping. 3) Clears the currently selected color rule. 4) Updates plot with the currently selected color rule. 5) Removes all previously added color rules.

sub-divided as short loops, long loops, pore loops, N-terminal loop and C-terminal loop.

Membrane, Helix and Loop Descriptions

The Membrane element can be depicted as a "box" (SVG "rect" element) or as a lipid bilayer (multiple SVG "path" elements). The Style Object controls all specifications of coloring and drawing aspects of these two membrane representations, such as their opacity and relative sizes. Likewise, helices elements can be plotted according to three possible *Style Object* draw types: "box", "cylinder" and "cartoon".

Loops can be drawn by different curves whose rendering depends on their classification. Two aspects are considered for the rendering of these curves: their curve type function and their curve scaling method. Curve type functions describe the shape of a given loop by generating points to be interpolated by the

d3.curveNatural function. Distances between these points have fixed or user-selected bounded proportions such that each curve type drawing aspect is scaled according to the Curve Scaling methods defined in the *Style Object*. Curve types common to the short and long loops include the "Simple", "Bulb" and "Mushroom" curves. The "swirl" curve type is specific to short Loops. Pore loops are generated by the "pore" curve type and N- and C- terminal by the "N Curves" curve type. The availability of multiple curve customization options allows users to customize plot aspects to their preferred style (Figure 3).

Design decisions for the representation of membranes, helices and loops attempted to cover most previously published Nav diagrams (Tanabe et al., 1988; Trimmer et al., 1989; Chiamvimonvat et al., 1996; Marban et al., 1998; Yu and Catterall, 2003; Yamaoka et al., 2006; Wood and Iseppon, 2018; Zybur et al., 2021). The usage of individual elements

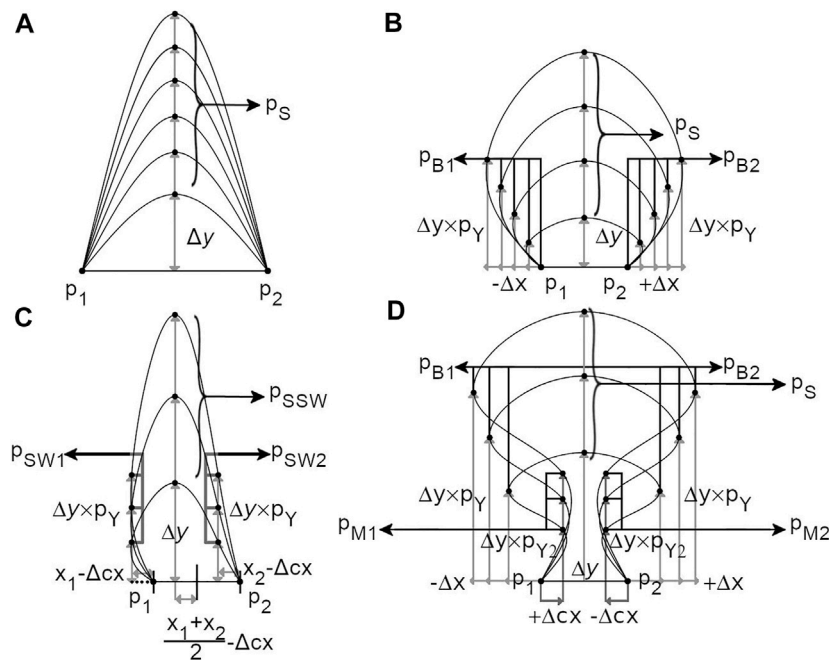


FIGURE 3 | Naview's curve drawing logic. **(A)** "Simple" curve function: a new point p_S is generated in the center of two anchoring points drawing functions and scaled by a Δy parameter according to the selected loop length scales. **(B)** The "Bulb" curve function in which two new points are generated in relation to the "Simple" curve type: p_{B1} and p_{B2} whose vertical growth is controlled by p_Y , a proportion of the total Δy . The horizontal position of these points is given by the Δx parameter in the opposite direction of their closest anchoring points. **(C)** "Swirl" curve function is a variation of the "Bulb" curve type whose horizontal position is defined in a symmetrical direction by a Δcx parameter, defined as a proportion of the distance of the anchoring points to their centroid. **(D)** The "Mushroom" curve type includes two new points in relation to the "Bulb" curve type: p_{M1} and p_{M2} . The vertical position of these points is defined by the p_{Y2} parameter as a proportion of the total Δy , and their horizontal position is defined from the anchoring points positions towards their centroid by the Δcx parameter.

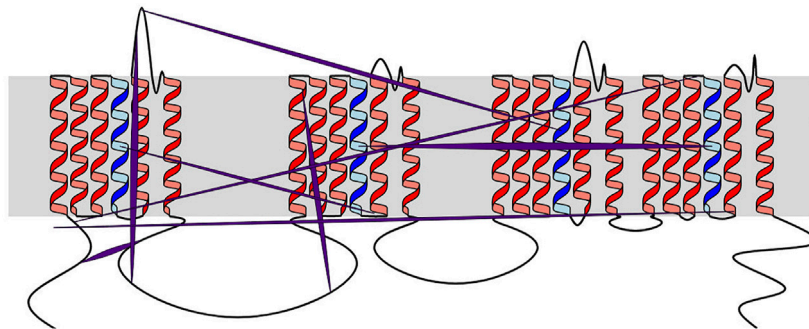


FIGURE 4 | Example of Naview's relationship drawing. Edges are colored in purple, with their central widths scaled according to the "raw_weight" column weights. This scaling allows the visual perception of stronger (larger width) and weaker (thinner width) relationships within the user inputted data. The membrane is shown as a grey box. All helices are shown as red cartoons except for the voltage-sensing helix 4, colored in blue.

inside a SVG document for each of the single Nav main secondary elements allowed the attribution of precise cartesian coordinates for each individual residue in this document. This enables the proper assignment of any text, color or edge annotations on the plot by the user.

Naview includes four scales to determine the loop length, depending on each loop type:

- "Fixed" in which a box of fixed height (and possibly width for "Bulb" and "Mushroom" curves) is set for determining the interpolating points of all loops of a given type (Short, Long, Pore or N/C terminus Loops).
- "Scaled" in which the height (and possibly width as above) of the boxes set for determining the interpolating points of all loops of a given type (Short, Long or Pore Loops) are set

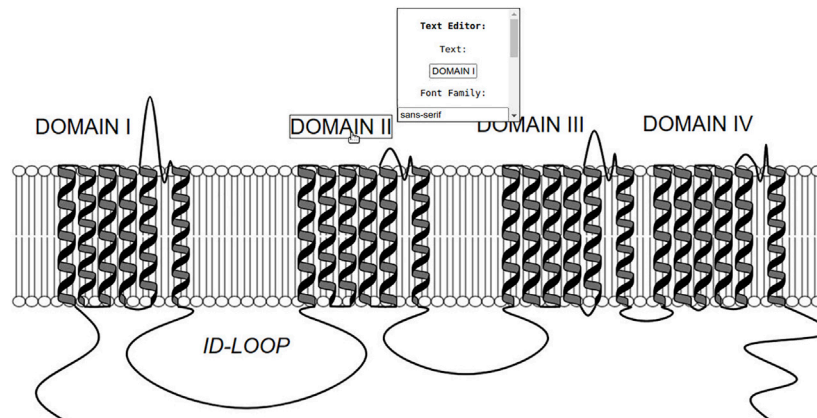


FIGURE 5 | Example of Naview's text annotations. All domain-indicating texts were added by using the Naview Style Editor console in the text edition mode. Text position can be adjusted by clicking and dragging any added text element. A single click highlights the selected text annotation and allows the editing of its current text and font characteristics. In this example such annotations were used to indicate specific domains (I-IV) and the first intracellular loop (ID-LOOP). Helices are shown as black cartoons and the membrane as a lipid bilayer. All loop residues are scaled to two pixels.

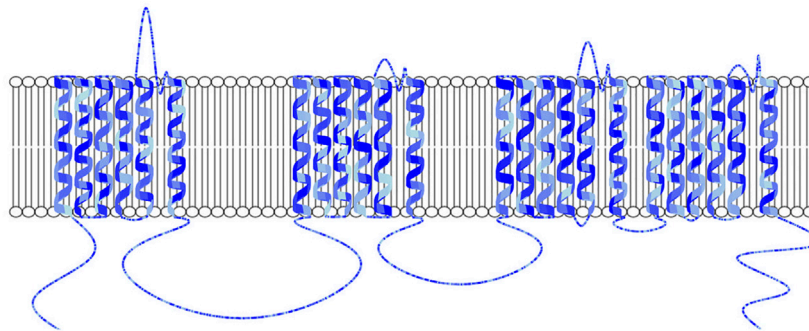


FIGURE 6 | Example of Naview's property-based color map from lightblue to blue after loading a CSV containing a randomly valued property named "Conservation" ranging from 0 to 1 for each of the protein's residues. Used color rule: "ALL, by:Conservation,#ADD8E6;#0000FF, min;max". As such residues with a higher "Conservation" value are colored in a darker tone of blue. Helices are shown as cartoons and the membrane as a lipid bilayer. All loop residues are scaled to two pixels.

from a linear, power or logarithmic scale of their amino acid numbers up to a maximum box height (and possibly width).

- "Reslen" in which the height (and possibly width) of each box of a loop-type (Short, Long, Pore or N/C terminus Loops) is defined by a specific pixel value.
- "Custom" in which boxes of fixed specific height (and possibly width for "Bulb" and "Mushroom" curves) are set for determining the interpolating points of each loop of a given type (Short, Long Loops).

All helix and loop coloring, opacity, stroke and scaling settings are controlled by properties of the *Style Obj*.

Input of Residue/Element Mapped Properties and Relationships

User-inputted residue properties and residue, helix and loop relationships can also be rendered as text annotations, specific

coloring rules and edges between residues or elements (**Figures 4–6**). The possibility of including properties and relationships in the plot differentiate Naview from drawing-only methods, by allowing the ability of the direct inclusion and visualization of experimental data. Both types of data can be either preloaded alongside the *Raw Text* (Examples in **Tables 1** and **2**) or included by the Naview Style Editor.

Specific property values mapped for a set of residues can be loaded and used to generate color scales for differential residue coloring or element mapped text annotations. These properties should be loaded as a JSON object in which each Nav alpha subunit residue index (Example 1,2,3... 2005 for a Nav containing 2005 residues) is used as a key for another dictionary, whose keys are strings describing a given property and whose values are those of the given properties for the selected residue (Example: 1:{*"Conservation"*:0.1},2:{*"Conservation"*:0.3},3:{*"Conservation"*:0.5} and henceforth).

Data representing relationships or interactions between different residues/elements present in the plot can be included

as a list of JSON inputs in the following format. Example: `{“source”: 1, “target”: DomainI;Helix6, raw_weight:0.5, “type”: “Residue Importance”}`.

Color Rules and Text Annotations

A list containing multiple color-filling text rules can be loaded as an input for generating a property-based residue color map. Accepted strings for color rules are any residue or element string keys followed by a comma-separated hex or string formatted color. Additionally, when properties have been mapped for a given Nav, they can be used for generating property-specific color maps.

Text annotations can be added as a list of JSON objects containing information about where a specific text should be drawn. This information can either be coded as absolute horizontal and vertical coordinates or as relative coordinates according to the positioning of a given residue or helix/loop element.

Alternatively, both color rules and text annotations can be added by the Naview Style Editor graphical interface (Figures 2, 5, 6).

RESULTS AND DISCUSSION

The existence of a diagram for displaying the alpha-subunit architecture of Nav for over 30 years highlights their usefulness in depicting important properties of these proteins. The Naview d3.js based JavaScript library described in this publication is the first automated method focused on generating these diagrams. Examples and the full documentation for this library can be found at: <http://bioinfo.icb.ufmg.br/naview/use> and <http://bioinfo.icb.ufmg.br/naview/public/docs/index.html>.

The construction of transparent, information-rich and thought-provoking visual narratives is an intrinsic challenge in bioinformatics data visualization which requires the management of different graphical elements for efficient communication (Tao et al., 2004; O'Donoghue, 2021). This challenge is addressed by Naview's through its high customization and data integrative potential and facilitated by the inclusion of a dynamic graphical interface. Since Naview is formatted as a fully documented JavaScript library, its inclusion in web data resources focused

on these channels can also be done simply and straightforwardly. By allowing the inclusion of residue mapped properties and relationships, Naview can be used for data exploration and integration purposes beyond the generation of publication-ready Nav figures.

In this publication, we demonstrate Naview and describe the logic of its implementation along with many of its features for plotting text, interactions and color mapped properties of sodium channels. Future updates should be focused on expanding the text annotation syntax to include drawing of polygons, arrows, backgrounds and other symbols, as well as reconfiguring the JavaScript library for drawing schemes and displaying data for any transmembrane/membrane-anchored protein.

DATA AVAILABILITY STATEMENT

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found in the article/Supplementary Material.

AUTHOR CONTRIBUTIONS

MA implemented most of the JavaScript code in the library. NdF tested, hosted the online version and wrote some additional JavaScript code and all PHP code. TM and LB tested the code and all authors contributed to the writing of the manuscript and gave ideas regarding the implemented features.

FUNDING

This work was supported by CNPq (Grant 457851/2014-7) and CAPES (Grant 051/2013 and MA scholarship). LB is a fellow researcher of CNPq.

ACKNOWLEDGMENTS

The authors would like to thank Lucas Carrijo de Oliveira for his helpful advice and discussions during writing.

REFERENCES

- Ahuja, S., Mukund, S., Deng, L., Khakh, K., Chang, E., Ho, H., et al. (2015). Structural Basis of Nav1.7 Inhibition by an Isoform-Selective Small-Molecule Antagonist. *Science* 350, aac5464. doi:10.1126/science.aac5464
- Beitz, E. (2000). T(E)Xtopo: Shaded Membrane Protein Topology Plots in LAT(E) X2epsilon. *Bioinformatics* 16, 1050–1051. doi:10.1093/bioinformatics/16.11.1050
- Bergwerf, H. (2015). MolView : an Attempt to Get the Cloud into Chemistry Classrooms. *ACS CHED CCCCE Newsl.* 2015, 1–9.
- Bond, C. S. (2003). TopDraw: a Sketchpad for Protein Structure Topology Cartoons. *Bioinformatics* 19, 311–312. doi:10.1093/bioinformatics/19.2.311
- Capes, D. L., Arcisio-Miranda, M., Jarecki, B. W., French, R. J., and Chanda, B. (2012). Gating Transitions in the Selectivity Filter Region of a Sodium Channel

- Are Coupled to the Domain IV Voltage Sensor. *Proc. Natl. Acad. Sci. U.S.A.* 109, 2648–2653. doi:10.1073/pnas.1115575109
- Cardoso, F. C., and Lewis, R. J. (2018). Sodium Channels and Pain: from Toxins to Therapies. *Br. J. Pharmacol.* 175, 2138–2157. doi:10.1111/bph.13962
- Chiamvimonvat, N., Pérez-García, M. T., Ranjan, R., Marban, E., and Tomaselli, G. F. (1996). Depth Asymmetries of the Pore-Lining Segments of the Na⁺ Channel Revealed by Cysteine Mutagenesis. *Neuron* 16, 1037–1047. doi:10.1016/S0896-6273(00)80127-0
- Chowdhury, S., and Chanda, B. (2019). Sodium Channels Caught in the Act. *Science* 363, 1278–1279. doi:10.1126/science.aaw8645
- Cousins, K. R. (2005). ChemDraw Ultra 9.0. CambridgeSoft, 100 CambridgePark Drive, Cambridge, MA 02140. www.cambridge.com. See Web Site for Pricing Options. *J. Am. Chem. Soc.* 127, 4115–4116. doi:10.1021/ja0410237

- Erickson, A., Deiteren, A., Harrington, A. M., Garcia-Caraballo, S., Castro, J., Caldwell, A., et al. (2018). Voltage-gated Sodium Channels: (NaV) Jigating the Field to Determine Their Contribution to Visceral Nociception. *J. Physiol.* 596, 785–807. doi:10.1113/JP273461
- Humphrey, W., Dalke, A., and Schulten, K. (1996). VMD: Visual Molecular Dynamics. *J. Mol. Graph.* 14, 33–38. doi:10.1016/0263-7855(96)00018-5
- Hutchinson, E. G., and Thornton, J. M. (1990). HERA--a Program to Draw Schematic Diagrams of Protein Secondary Structures. *Proteins* 8, 203–212. doi:10.1002/prot.340080303
- Jiang, D., Tonggu, L., Gamal El-Din, T. M., Banh, R., Pomès, R., Zheng, N., et al. (2021). Structural Basis for Voltage-Sensor Trapping of the Cardiac Sodium Channel by a Deathstalker Scorpion Toxin. *Nat. Commun.* 12, 128. doi:10.1038/s41467-020-20078-3
- Jmol development team (2016). *Jmol*.
- Johns, S. J. (2010). TOPO2, Transmembrane Protein Display Software. Available at: <http://www.sacs.ucsf.edu/TOPO2/>.
- Krause, S., Willighagen, E., and Steinbeck, C. (2000). JChemPaint - Using the Collaborative Forces of the Internet to Develop a Free Editor for 2D Chemical Structures. *Molecules* 5, 93–98. doi:10.3390/50100093
- Kubota, T., Durek, T., Dang, B., Finol-Urdaneta, R. K., Craik, D. J., Kent, S. B., et al. (2017). Mapping of Voltage Sensor Positions in Resting and Inactivated Mammalian Sodium Channels by LRET. *Proc. Natl. Acad. Sci. U S A.* 114, E1857–E1865. doi:10.1073/pnas.1700453114
- Laedermann, C. J., Abriel, H., and Decosterd, I. (2015). Post-translational Modifications of Voltage-Gated Sodium Channels in Chronic Pain Syndromes. *Front. Pharmacol.* 6, 263. doi:10.3389/fphar.2015.00263
- Marban, E., Yamagishi, T., and Tomaselli, G. F. (1998). Structure and Function of Voltage-Gated Sodium Channels. *J. Physiol.* 508 (Pt 3), 647–657. doi:10.1111/j.1469-7793.1998.647bp.x
- O'Donoghue, S. I. (2021). Grand Challenges in Bioinformatics Data Visualization. *Front. Bioinform.* 1, 669186. doi:10.3389/fbinf.2021.669186
- Omasits, U., Ahrens, C. H., Müller, S., and Wollscheid, B. (2014). Protter: Interactive Protein Feature Visualization and Integration with Experimental Proteomic Data. *Bioinformatics* 30, 884–886. doi:10.1093/bioinformatics/btt607
- Pan, X., Li, Z., Zhou, Q., Shen, H., Wu, K., Huang, X., et al. (2018). Structure of the Human Voltage-Gated Sodium Channel Nav1.4 in Complex with $\beta 1$. *Science* 362, 362. doi:10.1126/science.aau2486
- Pettersen, E. F., Goddard, T. D., Huang, C. C., Couch, G. S., Greenblatt, D. M., Meng, E. C., et al. (2004). UCSF Chimera--A Visualization System for Exploratory Research and Analysis. *J. Comput. Chem.* 25, 1605–1612. doi:10.1002/jcc.20084
- Rego, N., and Koes, D. (2015). 3Dmol.js: Molecular Visualization with WebGL. *Bioinformatics* 31, 1322–1324. doi:10.1093/bioinformatics/btu829
- Rose, A. S., and Hildebrand, P. W. (2015). NGL Viewer: a Web Application for Molecular Visualization. *Nucleic Acids Res.* 43, W576–W579. doi:10.1093/nar/gkv402
- Scheuer, T. (2011). Regulation of Sodium Channel Activity by Phosphorylation. *Semin. Cel Dev. Biol.* 22, 160–165. doi:10.1016/j.semcdb.2010.10.002
- Schrödinger, L. L. C. (2015). *The PyMOL Molecular Graphics System, Version~1.8*.
- Sehnal, D., Bittrich, S., Deshpande, M., Svobodová, R., Berka, K., Bazgier, V., et al. (2021). Mol* Viewer: Modern Web App for 3D Visualization and Analysis of Large Biomolecular Structures. *Nucleic Acids Res.* 49, W431–W437. doi:10.1093/nar/gkab314
- Sehnal, D., Deshpande, M., Vařeková, R. S., Mir, S., Berka, K., Midlik, A., et al. (2017). LiteMol Suite: Interactive Web-Based Visualization of Large-Scale Macromolecular Structure Data. *Nat. Methods* 14, 1121–1122. doi:10.1038/nmeth.4499
- Spyropoulos, I. C., Liakopoulos, T. D., Bagos, P. G., and Hamodrakas, S. J. (2004). TMRPres2D: High Quality Visual Representation of Transmembrane Protein Models. *Bioinformatics* 20, 3258–3260. doi:10.1093/bioinformatics/bth358
- Tanabe, T., Takeshima, H., Mikami, A., Flockerzi, V., Takahashi, H., Kangawa, K., et al. (1988). Primary Structure of the Receptor for Calcium Channel Blockers from Skeletal Muscle. *Nature* 328, 313–318. doi:10.1038/328313a0
- Tao, Y., Liu, Y., Friedman, C., and Lussier, Y. A. (2004). Information Visualization Techniques in Bioinformatics during the Postgenomic Era. *Drug Discov. Today BIOSILICO* 2, 237–245. doi:10.1016/S1741-8364(04)02423-0
- Todsén, W. L. (2014). ChemDoodle 6.0. *J. Chem. Inf. Model.* 54, 2391–2393. doi:10.1021/ci500438j
- Trimmer, J. S., Cooperman, S. S., Tomiko, S. A., Zhou, J. Y., Crean, S. M., Boyle, M. B., et al. (1989). Primary Structure and Functional Expression of a Mammalian Skeletal Muscle Sodium Channel. *Neuron* 3, 33–49. doi:10.1016/0896-6273(89)90113-X
- Wang, J., Youkharibache, P., Zhang, D., Lanczycki, C. J., Geer, R. C., Madej, T., et al. (2020). iCn3D, a Web-Based 3D Viewer for Sharing 1D/2D/3D Representations of Biomolecular Structures. *Bioinformatics* 36, 131–135. doi:10.1093/bioinformatics/btz502
- Widmark, J., Sundström, G., Ocampo Daza, D., and Larhammar, D. (2011). Differential Evolution of Voltage-Gated Sodium Channels in Tetrapods and Teleost Fishes. *Mol. Biol. Evol.* 28, 859–871. doi:10.1093/molbev/msq257
- Wood, J. N., and Iseppon, F. (2018). Sodium Channels. *Brain Neurosci. Adv.* 2, 2398212818810684. doi:10.1177/2398212818810684
- Xia, M., Liu, H., Li, Y., Yan, N., and Gong, H. (2013). The Mechanism of Na⁺/K⁺ Selectivity in Mammalian Voltage-Gated Sodium Channels Based on Molecular Dynamics Simulation. *Biophys. J.* 104, 2401–2409. doi:10.1016/j.bpj.2013.04.035
- Xu, H., Li, T., Rohou, A., Arthur, C. P., Tzakoniati, F., Wong, E., et al. (2019). Structural Basis of Nav1.7 Inhibition by a Gating-Modifier Spider Toxin. *Cell* 176, 702–e14. doi:10.1016/j.cell.2018.12.018
- Yamaoka, K., Vogel, S. M., and Seyama, I. (2006). Na⁺ Channel Pharmacology and Molecular Mechanisms of Gating. *Curr. Pharm. Des.* 12, 429–442. doi:10.2174/138161206775474468
- Yu, F. H., and Catterall, W. A. (2003/2003). Overview of the Voltage-Gated Sodium Channel Family. *Genome Biol.* 4 (4), 207–7. doi:10.1186/GB-2003-4-3-207
- Zhang, Y., Du, Y., Jiang, D., Behnke, C., Nomura, Y., Zhorov, B. S., et al. (2016). The Receptor Site and Mechanism of Action of Sodium Channel Blocker Insecticides. *J. Biol. Chem.* 291, 20113–20124. doi:10.1074/jbc.M116.742056
- Zybur, A., Hudmon, A., and Cummins, T. R. (2021). Distinctive Properties and Powerful Neuromodulation of Nav1.6 Sodium Channels Regulates Neuronal Excitability. *Cells* 10, 1595. doi:10.3390/cells10071595

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Afonso, da Fonseca Júnior, Miranda and Bleicher. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Development and Application of Automatized Routines for Optical Analysis of Synaptic Activity Evoked by Chemical and Electrical Stimulation

Debarpan Guhathakurta, Enes Yağız Akdaş, Anna Fejtová*[†] and Eva-Maria Weiss*[†]

Department of Psychiatry and Psychotherapy, Universitätsklinikum Erlangen, Friedrich-Alexander Universität Erlangen-Nürnberg, Erlangen, Germany

OPEN ACCESS

Edited by:

Lucy Collinson,
Francis Crick Institute,
United Kingdom

Reviewed by:

Yuan Shang,
University of Arizona, United States
Stephan Daetwyler,
University of Texas Southwestern
Medical Center, United States

*Correspondence:

Anna Fejtová
Anna.Fejtova@uk-erlangen.de
Eva-Maria Weiss
Eva-Maria.Weiss@uk-erlangen.de

[†]These authors share last authorship

Specialty section:

This article was submitted to
Data Visualization,
a section of the journal
Frontiers in Bioinformatics

Received: 12 November 2021

Accepted: 24 January 2022

Published: 15 February 2022

Citation:

Guhathakurta D, Akdaş EY, Fejtová A
and Weiss E-M (2022) Development
and Application of Automatized
Routines for Optical Analysis of
Synaptic Activity Evoked by Chemical
and Electrical Stimulation.
Front. Bioinform. 2:814081.
doi: 10.3389/fbinf.2022.814081

The recent development of cellular imaging techniques and the application of genetically encoded sensors of neuronal activity led to significant methodological progress in neurobiological studies. These methods often result in complex and large data sets consisting of image stacks or sets of multichannel fluorescent images. The detection of synapses, visualized by fluorescence labeling, is one major challenge in the analysis of these datasets, due to variations in synapse shape, size, and fluorescence intensity across the images. For their detection, most labs use manual or semi-manual techniques that are time-consuming and error-prone. We developed SynEdgeWs, a MATLAB-based segmentation algorithm that combines the application of an edge filter, morphological operators, and marker-controlled watershed segmentation. SynEdgeWs does not need training data and works with low user intervention. It was superior to methods based on cutoff thresholds and local maximum guided approaches in a realistic set of data. We implemented SynEdgeWs in two automatized routines that allow accurate, direct, and unbiased identification of fluorescently labeled synaptic puncta and their consecutive analysis. SynEval routine enables the analysis of three-channel images, and ImgSegRout routine processes image stacks. We tested the feasibility of ImgSegRout on a realistic live-cell imaging data set from experiments designed to monitor neurotransmitter release using synaptic phluorins. Finally, we applied SynEval to compare synaptic vesicle recycling evoked by electrical field stimulation and chemical depolarization in dissociated cortical cultures. Our data indicate that while the proportion of active synapses does not differ between stimulation modes, significantly more vesicles are mobilized upon chemical depolarization.

Keywords: segmentation algorithm, synapse detection, synaptic vesicle recycling, electrical stimulation, chemical depolarization, cultured neurons, image processing

INTRODUCTION

Neurotransmission is crucial for brain development, cognition, learning, and memory processes. In neuronal synapses, neurotransmitters are stored in synaptic vesicles (SVs). Upon stimulation of neurons, these vesicles fuse with the presynaptic plasma membrane to release neurotransmitter into the synaptic cleft, which is the key step in synaptic transmission. To preserve the presynaptic

structure and to ensure effective vesicular release during repetitive stimulations, SVs are retrieved from the presynaptic membrane and subsequently refilled with neurotransmitters. To study their properties, synapses in neurons can be visualized as synaptic puncta in neurons *in vitro*, *ex vivo*, or *in vivo* with fluorescence microscopy utilizing antibodies against pre- and postsynaptic proteins (Ivanova et al., 2020; Anni et al., 2021) or using genetically encoded reporter constructs (Ng et al., 2002; Welzel et al., 2011). Reliable detection of synaptic puncta is crucial for proper quantification of synaptic properties. In the past, automatized segmentation algorithms emerged as tools to reduce time need and human bias (Ippolito and Eroglu, 2010; Danielson and Lee, 2015; Kulikov et al., 2019). Nowadays, sophisticated segmentation algorithms based on machine learning are able to segment synapses precisely and comprise approaches working with very small sets of training data (Berg et al., 2019; Stringer et al., 2021). However, downstream postprocessing of bulk images and merging of received data are difficult. In fact, most labs still rely on human experts carrying out detection of synaptic puncta manually or semi-manually (Abraira et al., 2017; Ippolito and Eroglu, 2010; O'Neil et al., 2021). This procedure is time-consuming and error-prone and relies on reduced data amount. We think that routines, enabling a full analysis that includes preprocessing steps and postprocessing calculations, can improve this. Hence, we developed the segmentation algorithm SynEdgeWs that we implemented in frameworks to realize fully automatized routines performing image preprocessing, precise and robust puncta segmentation, and postprocessing of data. SynEval routine allows the analysis of three-channel images and embeds the readout of synaptic puncta features such as number, fraction, and emitted mean fluorescence intensity (MFI). ImgSegRout routine processes image stacks such as time-lapse imaging sequences. We applied ImgSegRout on a realistic live-cell imaging data set from experiments where SV release was monitored using genetically encoded markers, the so-called synaptic phluorins (Royle et al., 2008). Finally, as a proof of concept, we tested SynEval routine on a realistic data set intended to compare different approaches to induce neurotransmitter release in cultured neurons, namely electrical stimulation via field electrodes and chemical depolarization.

METHODS

Preprocessing

Efficient preprocessing of images is crucial for proper segmentation of synaptic puncta. In the first step, convolution of the original image creates a background image that is subtracted from the original image afterward (**Supplementary Methods S1.1**) (Sternberg, 1983). Negative values are set to zero and linear normalization enhances the contrast of acquired images. The preprocessing routine is additionally equipped with a retouching function for very bright regions that may disturb proper segmentation. This is an optional function, selectable via graphical user interface

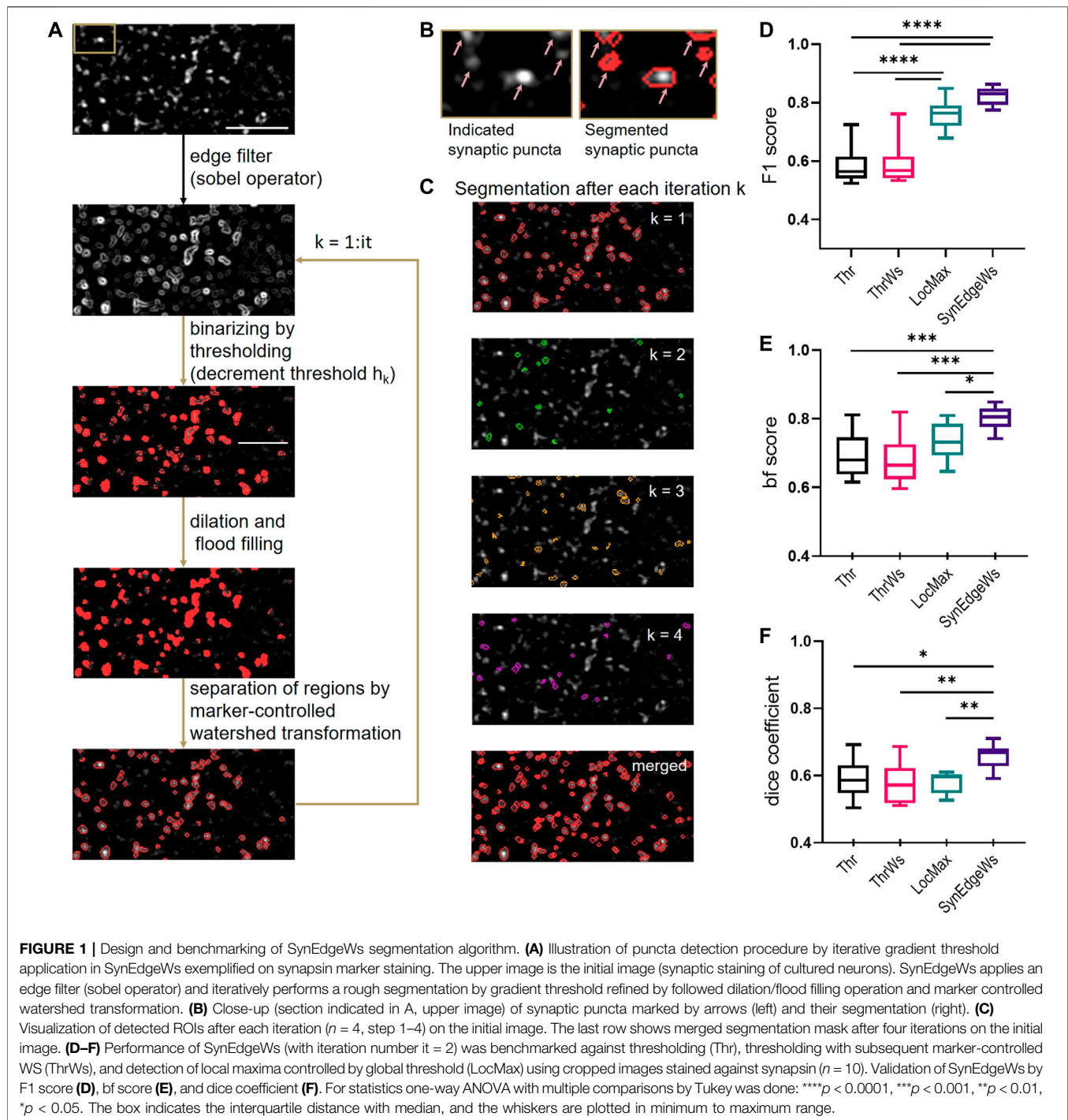
(GUI). Thereby, based on its characteristic bimodal shape, intensity histogram of the original image enables determining of a cutoff threshold value in-between the maxima to outline bright regions (**Supplementary Methods S1.1, Supplementary Figure S1**). Subsequent dilation and flood filling were implemented with MATLAB built-in functions. The resulting binary image masks the original image and the values of pixels within the mask are replaced with the corresponding pixel values from the background image. Subsequently, the background is subtracted from the whole image.

Segmentation Algorithm SynEdgeWs

We developed SynEdgeWs to detect automatically fluorescently labeled synaptic puncta without user intervention (detailed flowchart in **Supplementary Material Figure S2, Figure 1A**). While customized to work within the presented routines, SynEdgeWs implementation in new or modified routines is easy. In brief, an edge filter using sobel operator (Kanopoulos et al., 1988) calculates the image gradient (**Figure 1B**). Determined on an image gradient histogram, the application of the gradient threshold outlines the edges of synaptic puncta as a rough segmentation that is followed by dilation and flood-filling operations. To separate potentially connected puncta, marker-controlled watershed transformation operates within each section originating from intensity centroids. Afterward morphological operators (dilation/erosion) discard potential artifacts. To refine contour of regions of interest (ROI), thresholding checks border pixel values. Regions with a size beyond a certain range are discarded. Therefore, in the frameworks, minimum and maximum pixel numbers are calculated from expected synaptic puncta size in micrometer, camera pixel size, magnification, and binning adjustable via the GUI. The algorithm works with an iteratively decreasing image gradient threshold to overcome heterogeneous fluorescence intensity emitted by puncta (**Figure 1C, Supplementary Material—Methods S1.2**). For each iteration, the coordinates of detected synaptic puncta were stored in order to merge them finally. ROI detected during one iteration was excluded for the following iterations. This procedure avoids the detection of large regions that would be difficult to separate consecutively by watershed transformation. The user can determine the number of iterations *via* the GUI.

Routine SynEval for Segmentation of Antibody-Stained Synapses in a Multichannel Approach

The routine SynEval analyzes three-channel data in a batch process (**Figures 2A,B**). A GUI enables selecting images as TIFF files for each channel and configuring settings for the determination of the valid synaptic puncta size range in pixel counts (**Supplementary Material Table S1**). All images undergo preprocessing. The image recorded in channel 1 is set as a template. A segmentation mask and the corresponding list of ROI coordinates arise from running SynEdgeWs on this

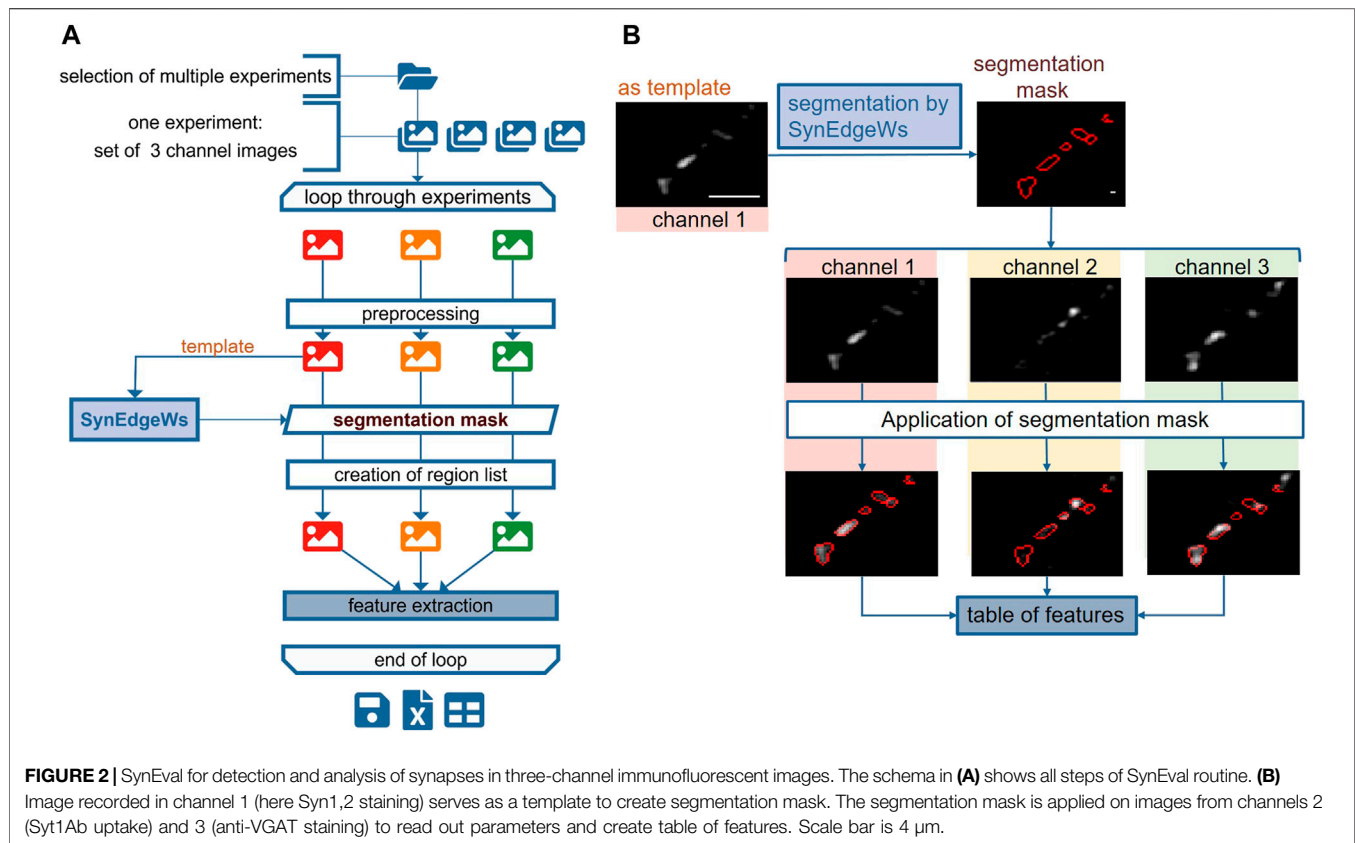


template. The ROI coordinates are transferred to channel 2 and 3 images and the MFI of each ROI from all the channels is obtained. To evaluate signal colocalization, the program determines a threshold for the signal in channels 2 and 3. Therefore, the application of edge filter, dilation, and flood filling results in a rough segmentation. Within this segmented region, the median of the lowest 1% fluorescence intensity is calculated and defines the threshold. Further postprocessing calculations provide a feature table

(Supplementary Material Table S2), which is exported as an MS excel file.

Routine ImgSegRout for Monitoring Fluorescence Signals Derived From Puncta

ImgSegRout processes time-lapse recordings saved as image stacks. It works in a batch mode and allows the operator to select several image stack files at once (Figure 3A). The routine is



based on our routine described in Anni et al. (2021) modified by herein-introduced segmentation algorithm SynEdgeWs and preprocessing procedure (process flow in **Figure 3A**). In brief, similar to SynEval, a GUI prompts to adjust settings for puncta size calculation, to select preprocessing features such as retouching and to insert the iteration number using SynEdgeWs (**Supplementary Material Table S1**). Moreover, postprocessing steps such as bleaching correction are selectable. Additionally, either by selecting a single frame or by selecting a sequence of frames consecutively averaged, the user determines a template for the segmentation process in SynEdgeWs. Subsequently, a multiple TIFF file is loaded. SynEdgeWs detects ROI on the template and returns a list of coordinates. ROI coordinates are transferred to each frame of the whole stack and MFI is read out. Additionally, the read-out process returns a background trace containing one background value per frame. Postprocessing includes subtraction of background values from individual fluorescent signal traces as well as smoothing and optional bleaching correction described in Anni et al. (2021). ImgSegRout exports all results as a MS excel file.

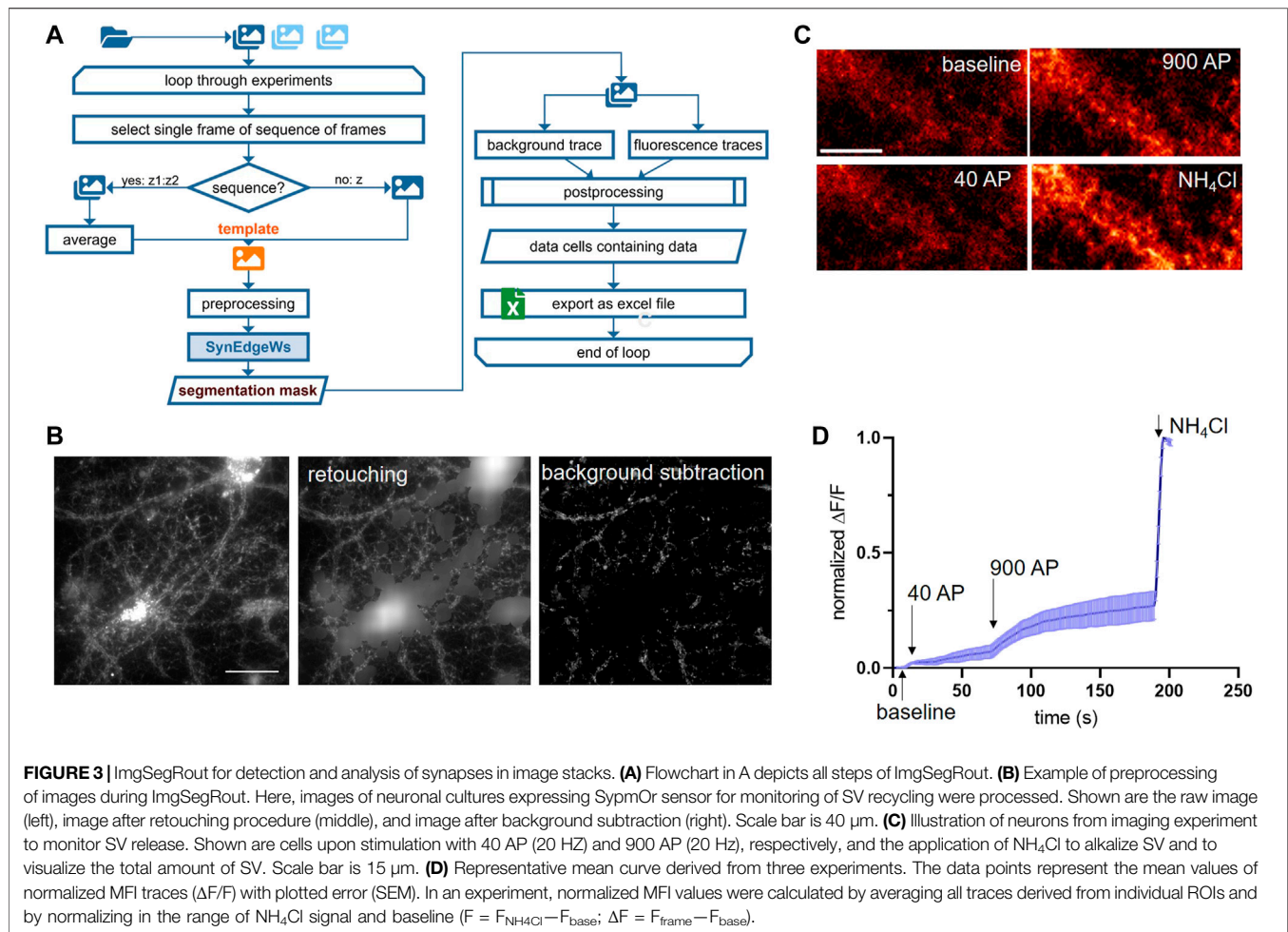
Primary Neuronal Cultures

Dissociated primary rat neuronal cultures were prepared exactly as described previously (Anni et al., 2021). The experiments involving animals in this study were approved by local animal

welfare officer (FAU: TS12/2016 and TS13/2016), in accordance with the European Directive 2010/63/EU and German animal welfare law. Briefly, cortices from E18 rat embryo were collected and cell suspension was obtained after trypsinization and mechanical trituration. Cells were plated in DMEM containing 10% (v:v) fetal calf serum, L-glutamine, and antibiotics on poly-L-lysine coated 18 mm Menzel glass coverslips at density of 120,000 cells/ml and kept at 37°C in 5% CO₂ atmosphere. 1 h later media was replaced to Neurobasal growth medium supplemented with B27, L-Glutamine, and antibiotics. Neurons were grown for 18–21 days *in vitro* (DIV) prior to all experiments (**Supplementary Material Table S3**).

Immunocytochemistry and Synaptotagmin1 Antibody Uptake Assay

Synaptotagmin1 antibody (Syt1Ab) uptake assay was carried out using Syt1Ab as described previously with slight modifications (Anni et al., 2021). For chemical stimulation, high KCl-Tyrode's buffer (TB) containing in mM: 69 NaCl, 50 KCl, 2 CaCl₂, 2 MgCl₂, 30 glucose, 25 HEPES, pH 7.4 and Syt1Ab (1:250 dilution) was applied to coverslips with DIV 18–21 neurons for 4 min at room temperature (RT). Thereafter, neurons were shortly washed and fixed in 4% (w:v) paraformaldehyde. For electrical stimulation, neurons were placed in a stimulation chamber and immersed in physiological TB, containing in mM: 119 NaCl, 2.5 KCl, 2 CaCl₂, 2 MgCl₂, 30 glucose, 25 HEPES, pH



7.4 and Syt1Ab. A train of 900 pulses (90 mA, 1 ms each) was delivered at 20 Hz using submersed electrodes. After 1 min, neurons were shortly washed and fixed in 4% (w:v) paraformaldehyde. The following steps were identical for electrically and chemically stimulated samples. For blocking and permeabilization coverslips were incubated in 10% (v:v) FCS, 0.1% (w:v) glycine, and 0.3% (v:v) TritonX 100 in PBS for 40 min. Primary antibody against VGLUT1 (1:1,000), VGAT (1:1,000), and synapsin 1,2 were applied overnight at 4°C in 1:1,000 dilution. The fluorescently labeled secondary antibodies were applied for 1 h at RT. All antibodies were diluted in PBS containing 3% (v:v) FCS. Coverslips were mounted in Mowiol. Images of immunofluorescence for all channels were acquired exactly as described previously (Anni et al., 2021) (Supplementary Material Table S3).

Preparation of Lentiviral Construct

To express the pH-sensitive synaptophysin-mOrange [SypmOr (Egashira et al., 2015)] in neuronal cultures, the SypmOr sequence was cloned into a FULW lentiviral vector (i.e., FUW with a modified multiple cloning site) using NEBuilder® HiFi DNA Assembly (NEB) through EcoRI and BamHI restriction sites. The production of virus in

HEK293T cells was done exactly as described in Anni et al. (2021). To transduce neurons, 100 μL of lentivirus containing medium was applied per coverslip at DIV 2 (Supplementary Material Table S3).

Live Imaging of SV Recycling Using SypmOr

Imaging was performed as in Anni et al. (2021) with minor modifications. Coverslips with neurons (DIV18–21) were placed in an electrical field stimulation chamber and imaged at RT in physiological TB containing 10 μM CNQX, 50 μM APV, pH 7.4, and 1 μM bafilomycin A1 on an epifluorescence microscope, using an automated perfect focus system (PFS) and 60X/NA1.2 water-immersion objective. Stimulus was generated using A 385 stimulus isolator connected to STG-4008 stimulus generator (Multi Channel Systems, Reutlingen, Germany). Subsequent to stimulations, TB containing 60 mM NH_4Cl was applied to achieve alkalization across all membranes. SypmOr fluorescent dye was excited at 543/22 with a Led-HUB lamp and time-lapse images were acquired using a Cy3 filter (emitter 593/40) at the frequency of 1 Hz using iXon EM + 885 EMCCD Andor camera controlled by VisiView software in 2 * 2 binning mode. Data were exported as stack files (.stk) containing frames with 502 × 501 pixels of 16-bit

monochromatic intensity values (**Supplementary Material Table S3, S4**). For further processing, stack files are converted into multiple TIFF files.

RESULTS

Performance of Segmentation and Routines

The aim of both presented routines is the fast, unbiased, and reproducible identification of synaptic puncta from images obtained by fluorescence microscopy and consecutive calculation returning a table of results as an Excel file. The in-house developed segmentation tool SynEdgeWs is the essential core algorithm of both routines and is imbedded in a framework of pre- and postprocessing procedures to allow direct usage on data with the purpose of saving time and reducing error potential.

Benchmarking of Segmentation Tool SynEdgeWs

Binarization of images by automatically determined cutoff threshold (Thr) (Sezgin and Sankur, 2004; Glebov, 2019) and local maxima determination, controlled by global threshold (LocMax) (Sbalzarini and Koumoutsakos, 2005; Xu et al., 2011) are still commonly used methods for image segmentation that run without user intervention and training data render them capable to run within the presented routines. To test SynEdgeWs algorithm, we benchmarked its performance against these methods by implementing them into the same environment in MATLAB. Since watershed transformation is a common method to separate connected puncta, we additionally implemented that to Thr (ThrWs) (Richter et al., 2018; Guo et al., 2019) (**Supplementary Material Methods S1.3**). To generate a reference segmentation as ground truth (ROI_{ref}), a human expert carried out manual segmentation of synaptic puncta on ten cropped images using Image Segmenter App (MATLAB). The same images were subsequently segmented by the four automatic segmentation methods resulting in respective ROI_{auto}. To compare all tested algorithms, F1 score was calculated. F1 is an established parameter to benchmark accuracy calculated as the harmonic mean of the performance metrics precision (positive predictive value) and recall (sensitivity) (Dice, 1945; Sørensen, 1948; Fawcett, 2006). Here, the calculation of F1 score underlies the comparison of individual ROIs (**Supplementary Material Method S1.4**). Additionally, we used built-in functions in MATLAB to measure further parameters to quantify segmentation quality. These are the F1 score, which compares the binary segmentation masks at pixel level, hereinafter referred to as dice coefficient (dice) (The MathWorks, 2017a) and the contour-matching score, also called boundary F1 score (bf score) (Csurka et al., 2004; The MathWorks, 2017b).

Benchmarking SynEdgeWs against LocMax yielded in significantly higher values for the measure dice (SynEdgeWs: 0.658 ± 0.011 , LocMax: 0.580 ± 0.010 , $p = 0.0040$) as well as bf

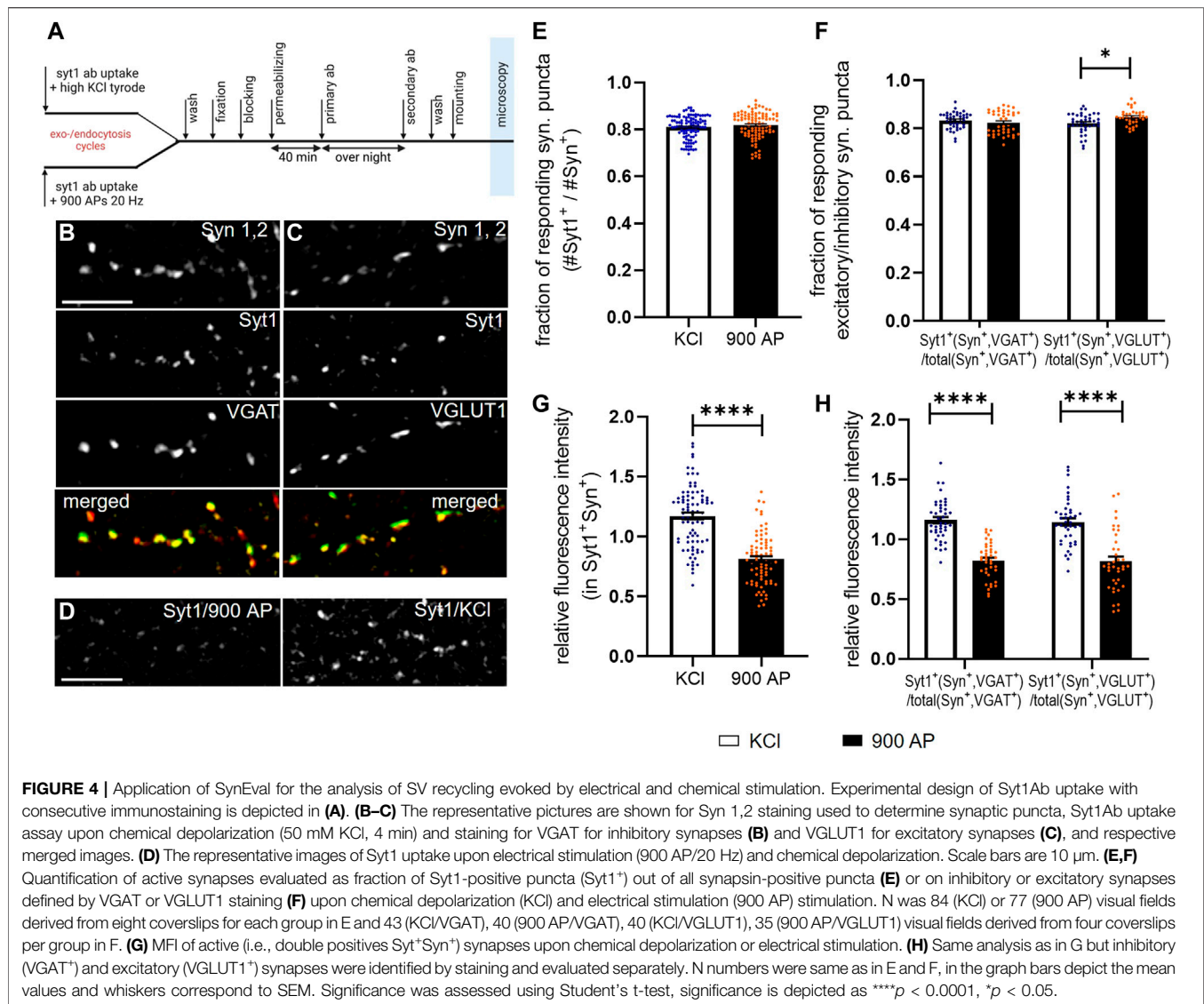
score (SynEdgeWs: 0.802 ± 0.011 , LocMax: 0.733 ± 0.016 , $p = 0.0461$) and higher values for F1 score (SynEdgeWs: 0.822 ± 0.010 , LocMax: 0.761 ± 0.016). For all measures, SynEdgeWs significantly outperforms Thr (F1 score: 0.582 ± 0.019 , $p < 0.0001$; bf score: 0.693 ± 0.021 , $p = 0.0007$; dice: 0.591 ± 0.018 , $p = 0.0158$) and ThrWs (F1 score: 0.590 ± 0.022 , $p < 0.0001$; bf score: 0.681 ± 0.021 , $p = 0.002$; dice: 0.577 ± 0.018 , $p = 0.0025$) in all measures (**Figures 1D–F**).

SynEval for Detection and Analysis of Synapses in Three-Channel Immunofluorescence Images

The MATLAB-based routine SynEval facilitates analysis of three-channel recordings in a batch process. The image recorded in channel 1 is set as a template to create a segmentation mask and a list of coordinates of detected ROI (**Figures 2A,B**). The ROI coordinates are transferred to the images recorded in channels 2 and 3 to read out parameters (**Supplementary Material Table S2**). To test SynEval on a realistic dataset, probes immunostained against synapsin 1,2 (Syn 1,2), a synaptic marker, were recorded in channel 1. The signal in channel 2 corresponded to Syt1 antibody labeling. This labeling had been previously performed in living cells to mark active synapses undergoing neurotransmitter release during antibody incubation (Kraszewski et al., 1995). The signal in channel 3 corresponded to staining for vesicular glutamate transporter 1 (VGLUT1), a marker for excitatory synapses (**Figure 4B**). Compared with manual analysis, this routine enables faster analysis. We tested running time ($n = 150$ images split into 15 runs) of our routine by using a built-in function stopwatch timer by MATLAB resulting in a mean value of 39.1 ± 4.8 s (**Supplementary Material Table S5**). Analysis of the same data, performed by a skilled experimenter using an optimized Fiji plugin (Wang et al., 2020), needed about 240 s per experiment. Additional 240 s were needed for postprocessing data carried out in MS Excel (**Supplementary Material Methods S1.5**). Thus, the time advantage gained by SynEval is around one order of magnitude compared with the semi-manual method. Time requirement for the user is further reduced courtesy of the batch mode.

ImgSegRout for Detection and Analysis of Synapses in Image Stacks

Time-lapse fluorescence imaging of optical probes targeted toward the lumen of SVs (synapto-pHluorins) is a common method to investigate release and recycling of SVs at the level of individual synapses, which is a proxy for neurotransmission (Sankaranarayanan et al., 2000). We developed ImgSegRout to monitor synapto-pHluorin fluorescence signals from time-lapse recordings, but, in general, the routine is capable of extracting fluorescence traces derived from any fluorescent puncta recorded as an image stack. It processes data in a batch mode (**Figure 3A**). To test ImgSegRout, we generated a realistic dataset by live-imaging neurons expressing the SypmOr reporter for monitoring SV fusion and retrieval (Egashira et al., 2015).



In this case, we implemented an approach described earlier by Burrone and colleagues (Burrone et al., 2006). Specifically, imaging was performed in the presence of bafilomycin to prevent vesicle reacidification, which allows visualization of cumulative release of SVs of different physiological properties. Release of a readily releasable pool of vesicles was induced by electrical field stimulation with 40 APs (pulses) at 20 Hz, release of all releasable vesicles was achieved by delivery of 900 APs at 20 Hz (Figures 3C,D). In these experiments, expression of SympOr reporter resulted in a strong fluorescence signal in neuronal cell bodies, which hampered the reliable segmentation process. Therefore, we applied a function integrated in preprocessing to retouch these very bright areas and to render images suitable for the segmentation process (Figure 3B). Testing performance of ImgSegRout by analyzing several real data experiments ($n = 12$ split in 3 runs) using the built-in stopwatch time function in MATLAB yielded in an averaged running time of 29.99 s per image stack with 260 images, 502×501 pixels. We

switched off bleaching correction, because the bleaching was minimal in these experiments.

Application of SynEval to Compare SV Recycling Induced by Chemical or Electrical Stimulation

Finally, we employed SynEval on realistic data with the aim of comparing SV release induced by chemical depolarization and electrical field stimulation. These two methods are broadly used in the field, but direct comparison of the data obtained by these alternative approaches was not yet performed. To close this gap, we labeled recycling vesicles evoked 1) by brief chemical depolarization with 50 mM KCl or 2) by electrical field stimulation with 900 APs at 20 Hz applied via submerged parallel field electrodes (Figure 4A). We used an antibody against luminal domain of SV protein Syt1 (Syt1Ab). This antibody binds its epitope only upon fusion to SV with plasma membrane (i.e., during depolarization/stimulation),

internalized during compensatory endocytosis and thus labels vesicles that have undergone exo- and endocytosis cycle during time of experiment. Following stimulation, cells were fixed and processed for immunostaining with antibodies for presynaptic marker Syn 1,2, as well as for marker of inhibitory (vesicular GABA transporter, VGAT) (**Figure 4B**) or excitatory (VGLUT1) (**Figure 4C**) synapses. Images were analyzed, using SynEval routine (**Figure 2A**). Syn 1,2 staining had been recorded in channel 1 to create segmentation mask with SynEdgeWs (**Figure 2B**). The segmentation mask determined ROIs on images from channel 2 (Syt1Ab uptake) and channel 3 (VGAT and VGLUT1, respectively) and application of threshold identified ROIs as positive for respective marker. To compare stimulation methods, proportion of synapses positive for Syt1Ab uptake as well as MFI of Syt1Ab uptake signal were analyzed (**Figures 4F–H**). While the first parameter reveals proportion of presynaptically silent synapses, the second relates to the relative number of SVs, which underwent exocytosis upon the respective stimulation at individual synapses and is a good proxy for presynaptic efficacy. The overall number of active (i.e., responding) synapses in relation to the total amount of synapses was similar upon both types of stimulation (**Figure 4E**, KCl: 0.829 ± 0.004 ; AP 900: 0.835 ± 0.005). In the next step, we analyzed proportion of active inhibitory and excitatory synapses. No difference was obvious in the proportion of inhibitory synapses, minor but significant increase was detected in the proportion of excitatory synapses upon electrical stimulation (**Figure 4F**, VGLUT1⁺, KCl: 0.813 ± 0.007 ; AP 900: 0.844 ± 0.007 /VGAT⁺, KCl: 0.935 ± 0.003 ; AP 900: 0.919 ± 0.004). In contrast, analyzing FI of Syt1Ab, depicted as relative FI related to overall mean, showed increased labeling upon depolarization with KCl compared with electrical stimulation (**Figure 4G**, KCl: 1.172 ± 0.028 ; AP 900: 0.8118 ± 0.024). This was true for both inhibitory and excitatory synapses (**Figure 4H**, VGLUT1⁺, KCl: 2.070 ± 0.031 ; AP 900: 0.835 ± 0.040 /VGAT⁺, KCl: 1.160 ± 0.026 ; AP 900: 0.823 ± 0.024). These data indicate that while the proportion of synapses that respond to chemical and electrical stimulation remains the same, the number of SV that are released upon chemical depolarization at excitatory and inhibitory synapses is significantly higher in comparison with neurons undergoing electrical field stimulation. This needs to be considered when interpreting the experimental outcomes using both stimulation regimes.

CONCLUSION

In this study, we implemented newly developed segmentation algorithm SynEdgeWs in fully automatized frameworks to combine precise, reliable, and fast identification of objects on fluorescently visible and acquired synaptic puncta images with complete pre- and postprocessing. The emerging routines SynEval and ImgSegRout are user-friendly turnkey solutions with the purpose of saving time and reducing human bias.

SynEdgeWs relies on gradient intensity. Since it does not rely on a cutoff intensity threshold to create a binary image, it is less affected by low signal-to-noise ratio or uneven illumination. We have proven SynEdgeWs to outperform algorithms based on threshold application and maxima-guided approaches, as determined by assessment of

accurate synapses localization (F1 score) and other measures. Since SynEdgeWs operates iteratively and applies decreasing thresholds for image gradient for each iteration, trade-off between specificity and sensitivity is adjustable depending on image data quality. Due to preservation of shape, this algorithm is potentially suitable to recognize virtually any other cellular structure defined by fluorescent signal that we aim to realize in future routines.

The routines SynEval and ImgSegRout were significantly faster than semi-manual methods. Moreover, the automatic routines are less prone to human error or individual variability, since they hardly involve any steps requiring manual intervention and therefore allow comparison of data obtained by different experimentations or laboratories. Both routines are applicable and adaptable to a wide range of experimental setups. We prepare all software packages for execution in MATLAB runtime enabling the use of software without installing MATLAB and provide routines with a GUI.

The GUI allows specifying further settings such as camera pixel size, magnification, binning, expected diameter of puncta in micrometer to define expected puncta dimensions in pixel counts and to exclude structures out of scope and reasoning. Both routines are equipped with pre- and postprocessing computations partly selectable via the GUI, like bleaching correction in the postprocessing of ImgSegRout or retouching of bright artifact in preprocessing.

Finally, the application of SynEval allowed us to answer a relevant biological question on comparing two different techniques broadly used to induce, monitor, and quantify SV release. Both electrical stimulation and chemical depolarization with KCl have their advantages depending on the experimental system. But without detailed knowledge about their relative potential to evoke SV release, the comparison of experiments using either of them is difficult. In our setting, the proportion of synapses, which are activated, does not differ between both methods. However, a direct comparison revealed that significantly more SV are mobilized upon chemical depolarization compared with electrical stimulations. We conclude that both electrical stimulation and chemical depolarization merit their place in different experimental settings, but chemical depolarization tends to mobilize vesicles that are not releasable upon intense electrical stimulation. It will be interesting to approach the molecular determinants of the observed difference in future experiments.

RESOURCE IDENTIFICATION INITIATIVE

All catalog numbers and RRID used in the study are given in **Supplementary Material Table S3, S4**.

DATA AVAILABILITY STATEMENT

The datasets presented in this study can be found in online repositories. Generated code and datasets for this study can be found here: <https://github.com/EvaMWe/Synapse-quantification>.

ETHICS STATEMENT

Ethical review and approval was not required for the animal study because the experiments involving animals in this study were approved by the local animal welfare officer (FAU:TS12/2016 and TS13/2016) and in accordance with the European Directive 2010/63/EU and German animal welfare law.

AUTHOR CONTRIBUTIONS

E-MW, DG, and AF conceptualized study; E-MW developed algorithm, programmed all routines, and wrote the first draft of the manuscript; DG: performed all experiments; and EA prepared lentiviral construct. All authors edited the manuscript and approved the final version.

REFERENCES

- Abraira, V. E., Kuehn, E. D., Chirila, A. M., Springel, M. W., Toliver, A. A., Zimmerman, A. L., et al. (2017). The Cellular and Synaptic Architecture of the Mechanosensory Dorsal Horn. *Cell* 168, 295–e19. doi:10.1016/j.cell.2016.12.010
- Anni, D., Weiss, E. M., Guhathakurta, D., Akdas, Y. E., Klueva, J., Zeitler, S., et al. (2021). Aβ1-16 Controls Synaptic Vesicle Pools at Excitatory Synapses via Cholinergic Modulation of Synapsin Phosphorylation. *Cell Mol Life Sci* 78, 4973–4992. doi:10.1007/s00018-021-03835-5
- Berg, S., Kutra, D., Kroeger, T., Straehle, C. N., Kausler, B. X., Haubold, C., et al. (2019). Ilastik: Interactive Machine Learning for (Bio)image Analysis. *Nat. Methods* 16, 1226–1232. doi:10.1038/s41592-019-0582-9
- Burrone, J., Li, Z., and Murthy, V. N. (2006). Studying Vesicle Cycling in Presynaptic Terminals Using the Genetically Encoded Probe synaptopHluorin. *Nat. Protoc.* 1, 2970–2978. doi:10.1038/nprot.2006.449
- Csurka, G., Larlus, D., Perronnin, F., and Meylan, F. (2004). What Is a Good Evaluation Measure for Semantic Segmentation. *IEEE PAMI* 26. doi:10.5244/c.27.32
- Danielson, E., and Lee, S. H. (2015). SynPAnal: Software for Rapid Quantification of the Density and Intensity of Protein Puncta from Fluorescence Microscopy Images of Neurons (vol 9, e115298, 2014). *Plos One* 10, e115298. doi:10.1371/journal.pone.0115298
- Dice, L. R. (1945). Measures of the Amount of Ecologic Association between Species. *Ecology* 26, 297–302. doi:10.2307/1932409
- Egashira, Y., Takase, M., and Takamori, S. (2015). Monitoring of Vacuolar-type H⁺ ATPase-Mediated Proton Influx into Synaptic Vesicles. *J. Neurosci.* 35, 3701–3710. doi:10.1523/JNEUROSCI.4160-14.2015
- Fawcett, T. (2006). An Introduction to ROC Analysis. *Pattern Recognition Lett.* 27, 861–874. doi:10.1016/j.patrec.2005.10.010
- Glebov, O. O. (2019). Distinct Molecular Mechanisms Control Levels of Synaptic F-Actin. *Cell Biol Int* 44 (1), 336–342. doi:10.1002/cbin.11226
- Guo, S. M., Veneziano, R., Gordonov, S., Li, L., Danielson, E., Perez de Arce, K., et al. (2019). Multiplexed and High-Throughput Neuronal Fluorescence Imaging with Diffusible Probes. *Nat. Commun.* 10, 4377. doi:10.1038/s41467-019-12372-6
- Ippolito, D. M., and Eroglu, C. (2010). Quantifying Synapses: an Immunocytochemistry-Based Assay to Quantify Synapse Number. *J. Vis. Exp.* 45. doi:10.3791/2270
- Ivanova, D., Imig, C., Camacho, M., Reinhold, A., Guhathakurta, D., Montenegro-Venegas, C., et al. (2020). CtBP1-Mediated Membrane Fission Contributes to Effective Recycling of Synaptic Vesicles. *Cell Rep* 30, 2444–e7. doi:10.1016/j.celrep.2020.01.079
- Kanopoulos, N., Vasanthavada, N., and Baker, R. L. (1988). Design of an Image Edge Detection Filter Using the Sobel Operator. *IEEE J. Solid-state Circuits* 23, 358–367. doi:10.1109/4.996

FUNDING

This work was supported by BMBF GeNeRARE (FZ 01GM1902B) and funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - (FE1335/3). We acknowledge financial support by Deutsche Forschungsgemeinschaft and Friedrich-Alexander-Universität Erlangen-Nürnberg within the funding programme. Open Access Publication Funding.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fbinf.2022.814081/full#supplementary-material>

- Kraszewski, K., Mundigl, O., Daniell, L., Verderio, C., Matteoli, M., and De Camilli, P. (1995). Synaptic Vesicle Dynamics in Living Cultured Hippocampal Neurons Visualized with CY3-Conjugated Antibodies Directed against the Lumenal Domain of Synaptotagmin. *J. Neurosci.* 15, 4328–4342. doi:10.1523/jneurosci.15-06-04328.1995
- Kulikov, V., Guo, S. M., Stone, M., Goodman, A., Carpenter, A., Bathe, M., et al. (2019). DoGNet: A Deep Architecture for Synapse Detection in Multiplexed Fluorescence Images. *Plos Comput. Biol.* 15, e1007012. doi:10.1371/journal.pcbi.1007012
- Ng, M., Roorda, R. D., Lima, S. Q., Zemelman, B. V., Morcillo, P., and Miesenböck, G. (2002). Transmission of Olfactory Information between Three Populations of Neurons in the Antennal Lobe of the Fly. *Neuron* 36, 463–474. doi:10.1016/s0896-6273(02)00975-3
- O'Neil, S. D., Rácz, B., Brown, W. E., Gao, Y., Soderblom, E. J., Yasuda, R., et al. (2021). Action Potential-Coupled Rho GTPase Signaling Drives Presynaptic Plasticity. *Elife* 10. doi:10.7554/eLife.63756
- Richter, K. N., Revelo, N. H., Seitz, K. J., Helm, M. S., Sarkar, D., Saleeb, R. S., et al. (2018). Glyoxal as an Alternative Fixative to Formaldehyde in Immunostaining and Super-resolution Microscopy. *EMBO J.* 37, 139–159. doi:10.15252/embj.201695709
- Royle, S. J., Granseth, B., Odermatt, B., Derevier, A., and Lagnado, L. (2008). Imaging Phluorin-Based Probes at Hippocampal Synapses. *Methods Mol. Biol.* 457, 293–303. doi:10.1007/978-1-59745-261-8_22
- Sankaranarayanan, S., De Angelis, D., Rothman, J. E., and Ryan, T. A. (2000). The Use of pHluorins for Optical Measurements of Presynaptic Activity. *Biophys. J.* 79, 2199–2208. doi:10.1016/S0006-3495(00)76468-X
- Sbalzarini, I. F., and Koumoutsakos, P. (2005). Feature point Tracking and Trajectory Analysis for Video Imaging in Cell Biology. *J. Struct. Biol.* 151, 182–195. doi:10.1016/j.jsb.2005.06.002
- Sezgin, M., and Sankur, B. (2004). Survey over Image Thresholding Techniques and Quantitative Performance Evaluation. *J. Electron. Imaging* 13, 146–168.
- Sørensen, T. (1948). A Method of Establishing Groups of Equal Amplitude in Plant Sociology Based on Similarity of Species and its Application to Analyses of the Vegetation on Danish Commons. *Kongelige Danske Videnskabernes Selskab* 5, 1–34.
- Sternberg, S. R. (1983). Biomedical Image Processing. *Computer* 16, 22–34. doi:10.1109/mc.1983.1654163
- Stringer, C., Wang, T., Michaelos, M., and Pachitariu, M. (2021). Cellpose: a Generalist Algorithm for Cellular Segmentation. *Nat. Methods* 18, 100–106. doi:10.1038/s41592-020-01018-x
- The MathWorks (2017b). *Bfscore*.
- The MathWorks (2017a). *Dice*.
- Wang, Y., Wang, C., Ranefall, P., Broussard, G. J., Wang, Y., Shi, G., et al. (2020). SynQuant: an Automatic Tool to Quantify Synapses from

- Microscopy Images. *Bioinformatics* 36, 1599–1606. doi:10.1093/bioinformatics/btz760
- Welzel, O., Henkel, A. W., Stroebel, A. M., Jung, J., Tischbirek, C. H., Ebert, K., et al. (2011). Systematic Heterogeneity of Fractional Vesicle Pool Sizes and Release Rates of Hippocampal Synapses. *Biophys. J.* 100, 593–601. doi:10.1016/j.bpj.2010.12.3706
- Xu, Y., Rubin, B. R., Orme, C. M., Karpikov, A., Yu, C., Bogan, J. S., et al. (2011). Dual-mode of Insulin Action Controls GLUT4 Vesicle Exocytosis. *J. Cel Biol* 193, 643–653. doi:10.1083/jcb.201008135

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Guhathakurta, Akdaş, Fejtová and Weiss. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Uncertainty Visualization: Concepts, Methods, and Applications in Biological Data Visualization

Daniel Weiskopf*

Visualization Research Center (VISUS), University of Stuttgart, Stuttgart, Germany

OPEN ACCESS

Edited by:

Sean O'Donoghue,
Garvan Institute of Medical Research,
Australia

Reviewed by:

Pere-Pau Vázquez,
Universitat Politècnica de Catalunya,
Spain
Daniel Haehn,
University of Massachusetts Boston,
United States

*Correspondence:

Daniel Weiskopf
weiskopf@visus.uni-stuttgart.de

Specialty section:

This article was submitted to
Data Visualization,
a section of the journal
Frontiers in Bioinformatics

Received: 12 October 2021

Accepted: 14 January 2022

Published: 17 February 2022

Citation:

Weiskopf D (2022) Uncertainty
Visualization: Concepts, Methods, and
Applications in Biological
Data Visualization.
Front. Bioinform. 2:793819.
doi: 10.3389/fbinf.2022.793819

This paper provides an overview of uncertainty visualization in general, along with specific examples of applications in bioinformatics. Starting from a processing and interaction pipeline of visualization, components are discussed that are relevant for handling and visualizing uncertainty introduced with the original data and at later stages in the pipeline, which shows the importance of making the stages of the pipeline aware of uncertainty and allowing them to propagate uncertainty. We detail concepts and methods for visual mappings of uncertainty, distinguishing between explicit and implicit representations of distributions, different ways to show summary statistics, and combined or hybrid visualizations. The basic concepts are illustrated for several examples of graph visualization under uncertainty. Finally, this review paper discusses implications for the visualization of biological data and future research directions.

Keywords: visualization, uncertainty, layout, visual mapping, sampling, graph visualization

1 INTRODUCTION

Data uncertainty can seriously affect its analysis and subsequent decision-making. Therefore, uncertainty should be considered in the context of visual data analysis and communication. This is well understood in many disciplines that deal with measured data. For example, error bars are widely used to indicate the uncertainty that comes with measurements, indicating standard mean of error or related descriptions of variability or uncertainty. However, uncertainty is not restricted to measurements but can also originate from numerical error in simulations, uncertainty in devising models, or many other sources.

In this paper, we discuss approaches to uncertainty visualization that do not restrict themselves to error bars. We address the problem of uncertainty visualization from a broader perspective, going beyond traditional statistical graphics and supporting more complex data than individual univariate distributions of data values, and therefore, linking to advanced visualization techniques. For many reasons, uncertainty visualization is difficult and considered one of the top research problems in visualization (Johnson, 2004). We will discuss some of the reasons and show strategies to address the problems.

There are already a number of survey papers on uncertainty visualization (see Section 2). We aim to complement them by adding some new perspectives: 1) We focus on presenting general concepts of uncertainty visualization, with an emphasis on strategies for visual mappings. Here, we will use a categorization that partially differs from existing ones, focusing on structuring the design space. 2) We build a bridge between sampling for visualizing uncertainty and modeling probability distributions, emphasizing the need for appropriate layout methods. 3) The general concepts are illustrated with examples in biological data visualization, and implications for visualization in bioinformatics are discussed.

This paper is written from the perspective of visualization research, as for example, presented in conferences like *IEEE VIS*, *EuroVis*, or *IEEE PacificVis* and journals like *IEEE Transactions on Visualization and Computer Graphics* or *Computer Graphics Forum*. Therefore, we want to build a connection between visualization research in general and applications in bioinformatics. Although this paper has some characteristics of a survey, it is not meant to be a systematic survey of (biological) uncertainty visualization techniques. Instead, we often use examples from our own previous work to illustrate concepts. The main goal is broad coverage of principles, concepts, and approaches.

We see the following benefits: This paper provides an overview of general strategies that can be useful to visualize uncertainty in biological data. We also discuss practical aspects of integration into biological data analysis and visual communication, as well as future directions.

This paper is based on and extends a talk from VIZBI 2021.¹

2 RELATED WORK

There are many survey papers on uncertainty visualization that cover the topic from different perspectives. The seminal paper by Pang et al. (1997) adopts a general classification of visualization techniques and applies it to uncertainty visualization. Their classification is based on: the value of the input data and its corresponding value uncertainty; the position of the data within the domain, along with its positional uncertainty; the extent of location and value; the visualization extent (discrete vs. continuous); and axes mappings. This kind of classification or variants thereof are good because they bring order into the large collection of visualization techniques in general, and uncertainty visualization techniques in particular. They also facilitate choosing a visualization based on data characteristics. However, this taxonomy is less suited to understand how uncertainty visualization works and how we can use the design space to come up with new uncertainty visualizations. Therefore, Pang et al. (1997) also characterize uncertainty visualization techniques according to the following categories: adding glyphs, adding geometry, modifying geometry, modifying attributes, animation, sonification, and psychovisual approaches.

Griethe and Schumann (2006) base their survey on categories that can be associated with the visualization design space, similar to Pang et al.'s latter characterization: using free graphical variables, including additional graphical objects, animation, interaction, or leveraging other human senses. Later papers by Potter et al. (2011) and Brodlie et al. (2012) primarily structure their surveys according to data type, in particular, the dimensionality of the domain and the attached data values and uncertainties. Bonneau et al. (2014) organize their survey according to traditional representations (in 1D, 2D, and for probability density functions), visual comparison techniques,

modification of attributes, glyphs, and image discontinuity. Ristovski et al. (2014) present a taxonomy focused on types of uncertainty and corresponding visualization challenges, concentrating on medical visualization. Siddiqui et al. (2021) summarize uncertainty visualization techniques for diffusion tensor imaging (DTI), considering the whole DTI visualization pipeline.

The above survey papers not only report on existing uncertainty visualization techniques, but also provide some background information: for example, on modeling uncertainty, how uncertainty data is acquired, and how uncertainty can be included in visualization processes or the visualization pipeline.

Jena et al. (2020) use a categorization with respect to publication type, publication venue, application domain, target user, and evaluation type. Their survey paper is accompanied by a web page² that can be queried and browsed according to the categorization and that comes with consistent descriptions and representative images for each visualization technique. Especially the thumbnail images facilitate quick browsing for potential solutions to uncertainty visualization problems.

Padilla et al. (2020) start from the design space of uncertainty visualization, distinguishing graphical annotations of distributional properties (showing intervals and ratios, or distributions), visual encodings of uncertainty, and hybrid approaches. They also summarize some theories for uncertainty visualization, bringing in a perspective from psychology.

A recent survey article is by Kamal et al. (2021). They use the following categories to structure uncertainty visualizations: geometry, attributes, animation, visual variables, graphical techniques, and glyphs. They also summarize the conceptual basis of uncertainty visualization, sources and models of uncertainty, evaluation approaches, and future research directions.

As pointed out by Griethe and Schumann (2006), not all taxonomies are necessarily useful in structuring existing uncertainty visualizations because they might result in very uneven distributions of papers to categories. Therefore, our categorization of visual mappings is inspired by the design-space-oriented classifications from Pang et al. (1997), Griethe and Schumann (2006), Bonneau et al. (2014), Padilla et al. (2020), and Kamal et al. (2021). Our structure of visual mappings in **Section 4** synthesizes a categorization based on variants from the above previous work, targeting strategies that can be used to develop new uncertainty visualization techniques.

The above survey papers are primarily based in the visualization research community. It should be noted that there is relevant related research in other fields as well. One prominent example is geography, geospatial science, and cartography; see the survey by MacEachren et al. (2005).

Related to perceptual and cognitive theories, Zuk and Carpendale (2006) applied principles by Bertin, Tufte, and Ware to examples of uncertainty visualizations to illustrate

¹D. Weiskopf: Uncertainty Visualization. Keynote presentation at the 11th International Meeting on Visualising Biological Data (VIZBI 2021)

²<https://namastevis.github.io/uncertaintyVizBrowser/>

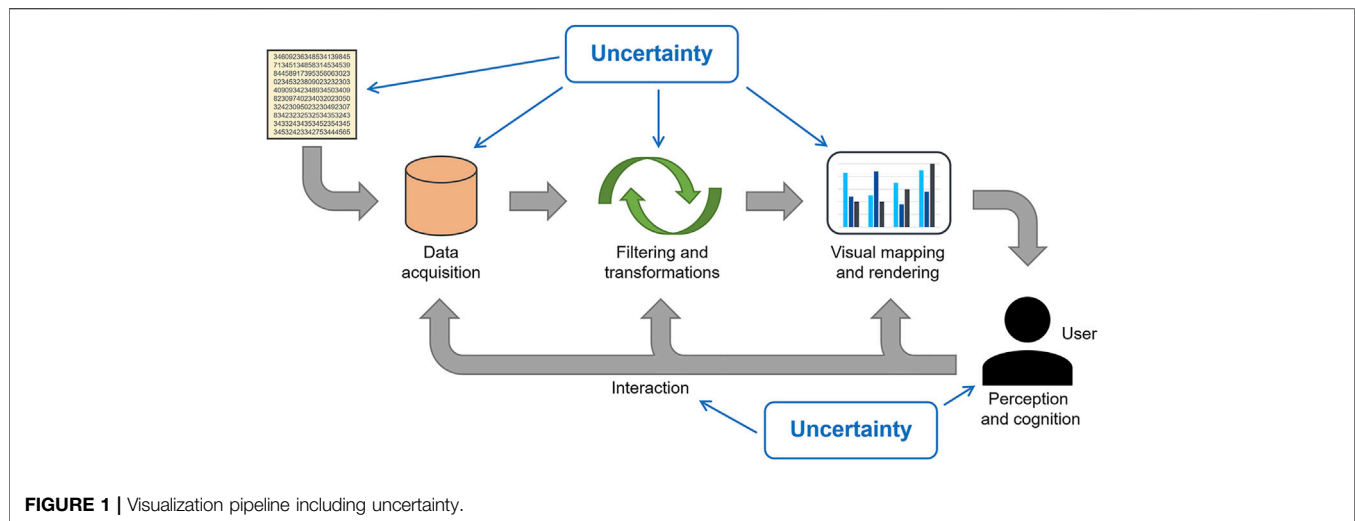


FIGURE 1 | Visualization pipeline including uncertainty.

and better understand these and assess them. While theirs is not a survey paper, it provides a theoretical underpinning that is useful in understanding uncertainty visualization. The survey paper by Hullman et al. (2019) focuses on one aspect of uncertainty visualization: its evaluation. Examples of evaluation papers include the ones by Deitrick and Edsall (2006) or Sanyal et al. (2009), but many more are reviewed by Hullman et al.

Skeels et al. (2010) pick out another important aspect: what are relevant models and types of uncertainty for visualization? Furthermore, visualization in general has to consider the analysis tasks that should be supported. Murray et al. (2017) provide a task taxonomy for the analysis of biological pathway data that includes identifying uncertainty. Also in the context of bioinformatics, Hamada (2014) summarizes several approaches to handle uncertainty, in particular, recommending visual representations.

It should also be noted that there are other concepts that are related to uncertainty and have some overlap. For example, ensemble visualization aims to show members from an ensemble, which can be viewed as a special case of describing variability. Therefore, uncertainty and ensemble visualization techniques show substantial overlap. Wang et al. (2019) provide a survey of ensemble visualizations. Other related concepts comprise human trust building or data provenance, as integrated into the framework by Sacha et al. (2016).

Some of the example visualizations that we demonstrate in this paper are based on (joint) research that went into the doctoral theses by Görtler (2021) and Schulz (2021). These theses also provide overviews on quantification for uncertainty visualization and approaches to making visualizations aware of uncertainty. In particular, they discuss sampling and layout methods for uncertainty visualization.

In summary, we do not want to replace the aforementioned surveys that come with a broad coverage of previous literature. Instead, our goal is to provide some additional perspective on the problem of uncertainty visualization. In contrast to most of the previous survey papers, we use many examples from biological data visualization to illustrate uncertainty visualization. Furthermore, we present a slightly different categorization of

visual mappings and point out specific issues that were not the focus of previous papers: the role and challenges of sampling for the implicit visualization of distributions, and the relevance of layouts for advanced uncertainty visualization.

3 OVERVIEW OF UNCERTAINTY VISUALIZATION

This section provides an overview of where and how uncertainty plays a role in visualization. We use the visualization pipeline to organize and structure the effects of uncertainty, see Figure 1.

Many of the previous survey papers employ the visualization pipeline as well (Pang et al., 1997; Griethe and Schumann, 2006; Brodlie et al., 2012; Ristovski et al., 2014; Kamal et al., 2021; Siddiqui et al., 2021). Our description is based on a pipeline for scientific visualization by Haber and McNabb (1990) and the related one for information visualization by Chi and Riedl (1998). However, we extend it slightly by including the human user (with their perceptual and cognitive aspects) and the interaction of the user with different stages of the pipeline. All of these need to consider uncertainty as well.

Following Brodlie et al. (2012), we can distinguish between *visualization of uncertainty* and *uncertainty of visualization*. The former is the typical focus when we address uncertainty visualization: showing the uncertainty that comes with the data. The latter term describes the additional uncertainty introduced by visualization—on top of the uncertainty associated with the data. Often, these two terms are treated in a combined fashion because they form the overall uncertainty in the final visualization.

There is an important point that comes with the visualization pipeline: The different stages have to be made uncertainty-aware and they have to be able to propagate uncertainty through the pipeline.

3.1 Uncertainty Modeling and Acquisition

One difficulty is that the term *uncertainty* is not well defined in the field of uncertainty visualization. In particular, there is not a

unique model of uncertainty. In some vagueness, it may refer to error, variability, or other aspects that may degrade the quality of data and visualization. Therefore, a typical challenge in using uncertainty visualization is to first understand the type of uncertainty that is to be shown. This is one of the critical elements in linking visualization to the specific application at hand.

There are a number of different taxonomies to describe various types of uncertainty. For example, we can distinguish between accuracy/error, precision, completeness, consistency, lineage, currency, credibility, subjectivity, and interrelatedness (MacEachren et al., 2005, 2012). Skeels et al. (2010) provide a classification in the form of measurement precision, completeness (covering missing values, sampling, aggregation), inferences (covering predictions, modeling, and descriptions of past events), disagreement, and credibility.

These models of uncertainty are determined by the sources of uncertainty and how it is used in the visualization and analysis. For example, there might be measurement errors, numerical errors from simulations, missing or corrupted data, variability from statistical observations, or from aggregating larger chunks of data into a compressed form.

Despite this vagueness, many uncertainty visualization techniques are based on some kind of probabilistic modeling of data uncertainty, i.e., in the form of probabilities or probability density functions. Furthermore, such uncertainty is often acquired by aggregation or computing summary statistics such as mean, median, standard error, percentiles, etc. Therefore, unless stated otherwise, we assume such probabilistic modeling and that uncertainty is described by summary statistics, by parameters of probability models (like parameters of probability density functions), or by providing original data samples (from which statistical descriptions could be computed).

3.2 Filtering and Transformations

Usually, the input data is not directly mapped to a visual representation. In particular, for large or complex data, it might be necessary to reduce the amount of data shown. Therefore, filtering and transformations of the input data are required to obtain data that is more informative: it might be reduced in amount or complexity, or important features might be extracted for highlighting. Therefore, this stage of the visualization pipeline is critical for avoiding or reducing information overload.

Filtering can be as simple as selecting data items based on allowed ranges of data, which might be specified by the user or driven by the distribution of the input data. Clustering is a common transformation approach in visualization because it facilitates structuring and grouping data, supporting summarized and compact representations; see, for example the survey paper by Xu and Wunsch (2005). Another typical example is the use of dimensionality reduction methods (or multidimensional projection) that allow one to transform high-dimensional input data to 2D or 3D data, leading to an easy mapping to visualization space. For background reading, see, for example, the book on nonlinear dimensionality reduction by Lee and Verleysen (2007). Modeling in high-dimensional space is

very generic and can be used for manifold applications. One bioinformatics example is the representation of phylogenetic trees that lends itself to multidimensional projection and uncertainty visualization (Willis and Bell, 2018).

Complex types of transformations can introduce additional uncertainty, i.e., they can lead to increasing visualization uncertainty. For example, multidimensional projections cannot fully guarantee the preservation of the original characteristics of the input data. The introduced distortions from projections can be identified and visualized, as summarized in a survey paper by Nonato and Aupetit (2019). Or, as in fuzzy clustering (Baraldi and Blonda, 1999), transformations might provide gradual or fuzzy assignments to clusters on purpose, again resulting in uncertainty that only originates at this stage of the visualization pipeline.

However, transformations do not only contribute to visualization uncertainty, they also have to be able to propagate incoming uncertainty downstream the pipeline. In this case, the transformation stage does not add errors during the process, but it has to pass them through appropriately. Since transformations can be highly nonlinear, this propagation might be hard to compute and it might distort the uncertainty substantially.

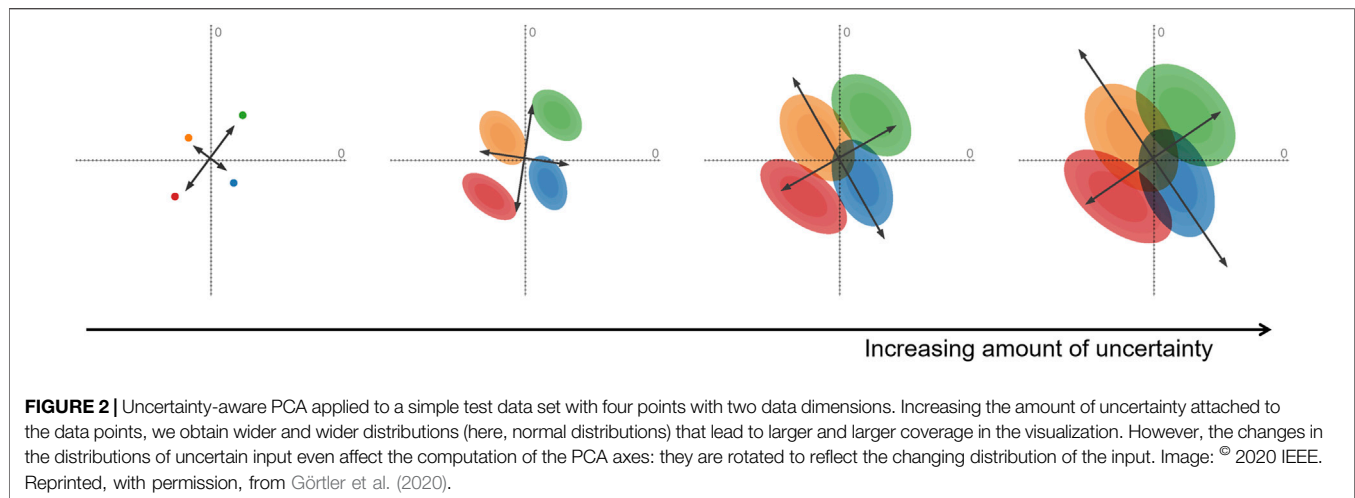
For example, uncertainty-aware principal component analysis (PCA) (Görtler et al., 2020) incorporates the uncertainty in high-dimensional data points to adapt the computation of the projection operator. **Figure 2** illustrates the effect of uncertainty on PCA. Uncertainty not only affects the display of the data points (which get wider with increasing uncertainty), but it even impacts the projection directions as indicated by the rotation of the PCA axes.

This example demonstrates the importance of making transformations aware of uncertainty. While there are uncertainty-aware variants already for some of the typical filtering and transformation techniques, there is still much room for future work in this direction. This is a research question not just for visualization but any field where numerical analysis of uncertain data is performed. Therefore, related methods may be developed in a range of different research fields.

3.3 Mapping and Rendering

The mapping stage of the visualization pipeline takes the transformed data and produces a renderable representation, for example, in the form of geometry together with attributes like color or opacity. Such geometry could be the set of points to be shown in a scatterplot, or a triangle mesh for an isosurface. This representation is then rendered to generate the final visualization image. The actual rendering is mostly well understood, with manifold techniques available from computer graphics.

In contrast, the mapping stage is in the center of visualization because it is the critical link between data and image. Developing appropriate visual mappings can already be hard for visualization without uncertainty, and it becomes even more challenging for uncertainty visualization. Visual mapping is a focal point of this paper, with a detailed discussion of mapping strategies in a dedicated later section (see **Section 4**).



3.4 Perception and Cognition

Visualization only works in combination with a human that uses imagery to understand the data or communicate with others. Therefore, visual perception and cognition play a critical role in visualization in general (Ware, 2021). In this context, user-oriented evaluation of visualization techniques is relevant and challenging at the same time (Lam et al., 2012); there is even a specialized series of workshops addressing evaluation methods for visualization.³

Including uncertainty makes understanding and assessing perception and cognition even harder. In particular, we have to be careful in designing uncertainty visualization so that it is correctly understood by the recipient. For example, even researchers have problems understanding and correctly judging the information encoded in the, at first sight quite simple, visualizations in the form of confidence intervals and error bars (Belia et al., 2005). These findings led to recommending alternatives to error bars (Correll and Gleicher, 2014).

Error bars are quite simple and very common; therefore, it is conceivable that more complex uncertainty visualizations could be affected even more from difficulties with perceiving and understanding them (Boukhelifa and Duke, 2009). Assessing cognitive aspects is particularly hard when complex decision-making has to be done under uncertainty (Padilla et al., 2021). It can also make a difference whether experts or non-experts use and read uncertainty visualizations. For example, Tak et al. (2014) study how non-experts perceive and understand typical examples of uncertainty visualizations. Some theories and further examples of perceptual and cognitive considerations are summarized by Padilla et al. (2020). Similarly, special attention needs to be paid to perform a proper evaluation of uncertainty visualization; see the survey paper by Hullman et al. (2019).

³BELIV: Evaluation and Beyond – Methodological Approaches for Visualization, <https://beliv-workshop.github.io>

3.5 Interaction

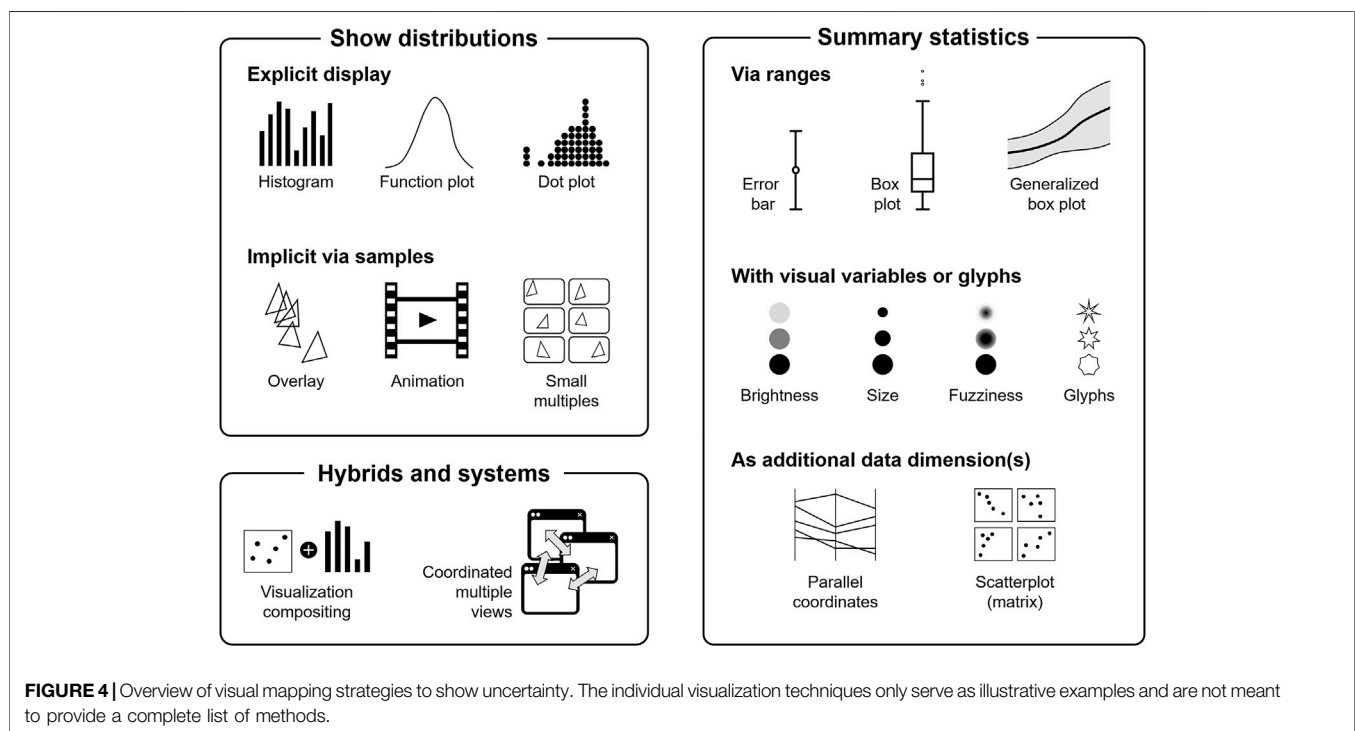
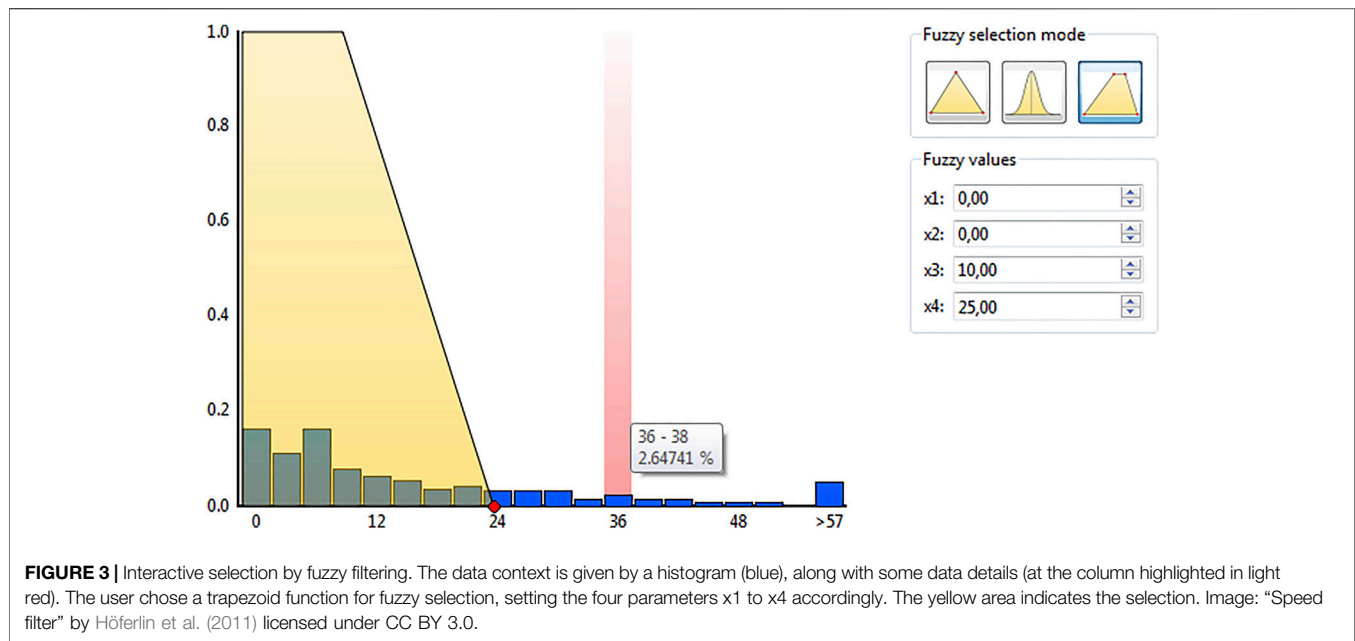
While uncertainty visualization sometimes targets passive consumption, for example, in the form of an illustration for visual communication, it is often employed in an interactive environment. Interactive visualization or visual analytics are typically used to facilitate visual data analysis.

Therefore, the interaction needs to be made aware of uncertainty as well. This includes how data serves as the basis for the interaction technique. However, uncertainty can also be present in the interaction itself. The user may not be sure about what they want to exactly specify with their input. For example, the input may serve as a threshold for interactive filtering. Here, uncertain input may be specified by sliders that are connected to uncertainty in the form of probability density functions (Greis et al., 2017). Another example is fuzzy selection facilitated by several selection modes, including triangle and trapezoidal shapes (Höferlin et al., 2011); see Figure 3.

Overall, the topic of uncertainty-aware interaction has not received much attention in visualization research. Therefore, we see the need for more work in this direction. One challenge is that this is directly linked to the difficult problem of understanding cognition and mental models of uncertainty—related to the previous subsection. Another challenge is that uncertainty-aware interaction has to be adapted to the different steps of the visualization pipeline. For example, specifying uncertain value ranges (as in the two examples above) is appropriate for defining value-oriented filtering, but different inputs are needed for other filters, transformations, or visual mappings.

3.6 Integration

So far, we have discussed the stages of the visualization pipeline one after another. However, uncertainty needs to be propagated through the whole process (Wu et al., 2012). Unfortunately, it can be hard to accurately compute uncertainty propagation because the various stages of the visualization pipeline can be quite complex and highly nonlinear. In particular, it is challenging to include human perception, cognition, and interaction in this



propagation. Another problem is that typical uncertainty propagation methods tend to increase uncertainty substantially, especially, when transformations are highly sensitive or when there is a sequence of transformations. The uncertainty estimates are often too conservative and, therefore, unrealistically large if uncertainty is passed on without a full model of the data and visualization process. By including

additional information, more accurate and tighter descriptions of uncertainty might be possible.

Overall, the whole visualization process should be made aware of uncertainty (Correa et al., 2009). Since this might not be fully possible, we recommended assessing the visualization workflow and identifying the most substantial contributors to uncertainty, along with the intended visualization goals and tasks. Based on

this, efforts in incorporating uncertainty can be directed to the most relevant components.

4 VISUAL MAPPING

In this section, we discuss visual mappings of uncertainty in more depth. Visual mapping strategies are summarized in **Figure 4**. This figure is inspired by the visual summary used by Padilla et al. (2020). However, our categorization partially differs from theirs and also from the other taxonomies reviewed in **Section 2**. Please note that the icons in **Figure 4** illustrate typical representatives for the respective strategy, but they are not meant to be comprehensive, i.e., it is to be understood that there are more visualization approaches for the respective strategy.

If not stated otherwise, we assume a probabilistic model of uncertainty—typically in the form of probability density functions (PDFs) describing distributions of data values. These may be reduced to concise characteristic descriptions, for example, by summary statistics. Or the raw samples might be available before computing summary statistics or constructing PDFs.

4.1 Explicit Visualization of Distributions

Let us start with the first kind of visual mappings: these aim to show distributions explicitly and fully. For example, a PDF can be seen just as a function and, therefore, a function plot displays the uncertainty distribution comprehensively. If the uncertainty data is provided as “raw” sampled data, traditional histograms in the form of bar charts can be employed. An alternative is the dot plot (Wilkinson, 1999), or the nonlinear dot plot (Rodrigues and Weiskopf, 2018) for higher dynamic range. The sample-based visualization can even be used if only a PDF is available: just by drawing samples from the given PDF.

The advantage of the explicit visualization of distributions is that they provide full disclosure of uncertainty information. A disadvantage is the extra visualization space needed: frequency or probability (density) are plotted along an axis (usually, the vertical axis) that is perpendicular to the axis that carries the data values (usually, the horizontal axis), i.e., we require 2D space instead of 1D space just for the data axis.

A related characteristic is that the 2D visualization axes carry different meanings: data values vs frequency or probability (density). This difference can have benefits if we want to clearly separate the two meanings. At the same time, it can lead to problems if the visualization space is taken as one 2D space.

Overall, the explicit visualization of distributions is typically employed for rather small data sets, or for data drill down to show detailed views on large data sets.

4.2 Implicit Visualization of Distributions Via Samples

Some problems of the above explicit visualization can be addressed by showing distributions implicitly *via* samples drawn from the distribution. The basic process is as follows: In the first step, the distribution is sampled to produce potential

realizations of the data, compatible with the uncertainty representation. Each sample is treated as if it was not affected by uncertainty. In the second step, each sample is visualized. The last step is responsible for showing the visualizations of all samples in some combined fashion.

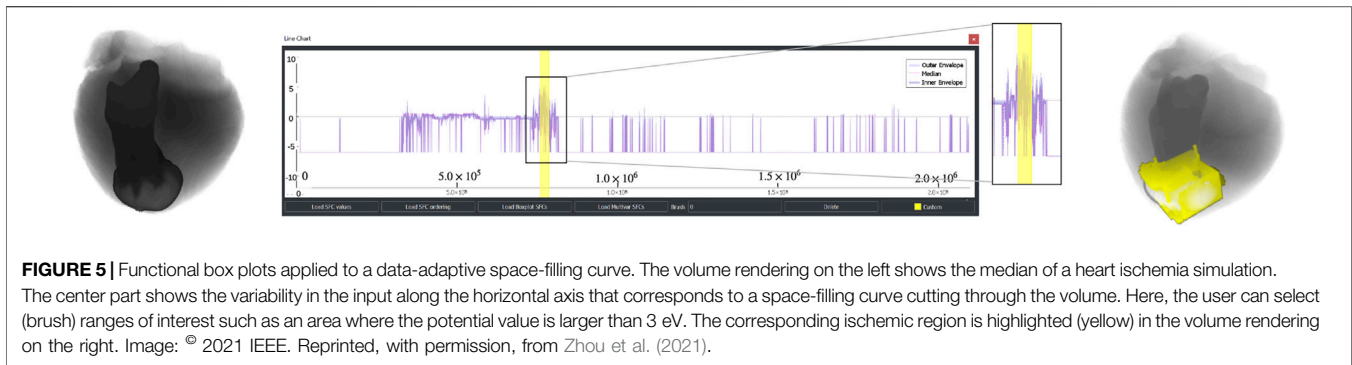
Variants of this uncertainty visualization approach mostly differ in the way they implement the last step. One option is to overlay or composite the individual visualizations of the samples, for example, by additive blending or alpha blending (Schulz et al., 2017). Another option is the use of animation, showing individual visualizations one after another, e.g., in the form of the animation of potential realizations (here, surfaces) by Ehlschlaeger et al. (1997) or in the form of Hypothetical Outcome Plots (Kale et al., 2019). Yet another option places individual visualizations next to each other in one large image, in the form of small multiples (Tufte, 1990).

All of these implicit visualizations have the advantage that they just use the regular visualization space, i.e., there is no need for extra space with other semantics, as for the explicit visualization of distributions. Therefore, the uncertainty visualization should be understandable by the user if they are familiar with the original, non-uncertainty-affected visualization. The variants for the last step have specific advantages and disadvantages. The overlay approach has the advantage that it essentially needs just the visualization space that a single visualization would need. Another advantage is that it results in a static image, i.e., it can be flexibly used in visual communication, and it gives the user enough time to carefully inspect the visualization. The main disadvantages are overplotting, clutter, and ambiguities that can arise from compositing many visualization samples.

The animation approach avoids this overplotting and provides some advantages in interpreting uncertainty (Kale et al., 2019). However, this approach comes with typical problems of animated visualization that can be difficult for analysis tasks (Robertson et al., 2008). Animation also has some issues with scalability with the number of samples shown: it is hard to get a quick overview, which in contrast is possible with the single and static image in the overlay approach.

Small multiples are similar to the animated display because they show individual visualizations independently. The main difference is that animation puts the individual images one after another along time, whereas small multiples place them next to each other in an enlarged visualization space. Similarly to animation, this approach avoids overplotting. However, it needs much visual space and, again, has issues with the scalability regarding the number of samples. Also, it might be hard to perceive and interpret differences between the individual visualizations.

While the visual representation is quite different in the three approaches, they all share the need for appropriate registration or alignment between the individual images—whether these are the images that go into the blending, animation, or as part of the small multiples. The potential problem is that individual images may look very different even if the sampling from the distribution leads to similar data. In other words, some visualization techniques can be very sensitive to slight changes in the input data. For example,



many graph drawing algorithms can lead to quite different outputs even if the input is similar (e.g., in the form of rotated images). **Section 5** discusses the registration problem and visualization approaches for the example of graph drawing in more detail.

4.3 Summary Statistics as Range Plots

The above explicit and implicit visualizations aim to show the full characteristics of the underlying distributions. However, it is often sufficient to convey just some aggregated or concise representation of the distributions. For example, summary statistics may rely on some indicator of central tendency (such as mean or median) and variability (like standard deviation, standard error, or percentiles). Statistical graphics then maps these summarizations to visual representations such as error bars or box plots.

From the perspective of visualization, these mappings lead to a representation of ranges. For example, a typical box plot shows the range from the 25 percentile to the median and then to the 75 percentile, where each of the boundaries is indicated by a line in the box plot. Another observation is that these range plots need additional visualization space to make room to show the ranges. Therefore, they work fine for traditional statistical plots where one has just a few data items that are enriched by statistical graphics. However, it becomes harder to fit the range plots into a visualization that already needs a lot of space on the image to show data without uncertainty.

One strategy maps the original data to a lower-dimensional visual representation that supports adding ranges. For example, 3D volume data can first be reduced to a 1D curve by letting a space-filling curve cut through the volume; afterward, we can apply bands or range representations around the curve (Demir et al., 2014). **Figure 5** shows an example that uses a data-adaptive space-filling curve to perform the reduction to 1D (Zhou et al., 2021). Here, the data comes from a heart ischemia simulation; see Rosen et al. (2016) for background reading.

Another strategy relies on a generalization of the idea of a box plot, utilizing the concept of statistical depth, which can be seen as the generalization of medians or percentiles in complex data. For example, contour box plots indicate parts or ranges in a spatial domain that correspond to certain values or ranges of depth (Whitaker et al., 2013). Another example shows variability in functions by function box plots (Mirzargar et al., 2014).

Yet another strategy places small glyphs on the domain to indicate data ranges at respective locations. For example, radial

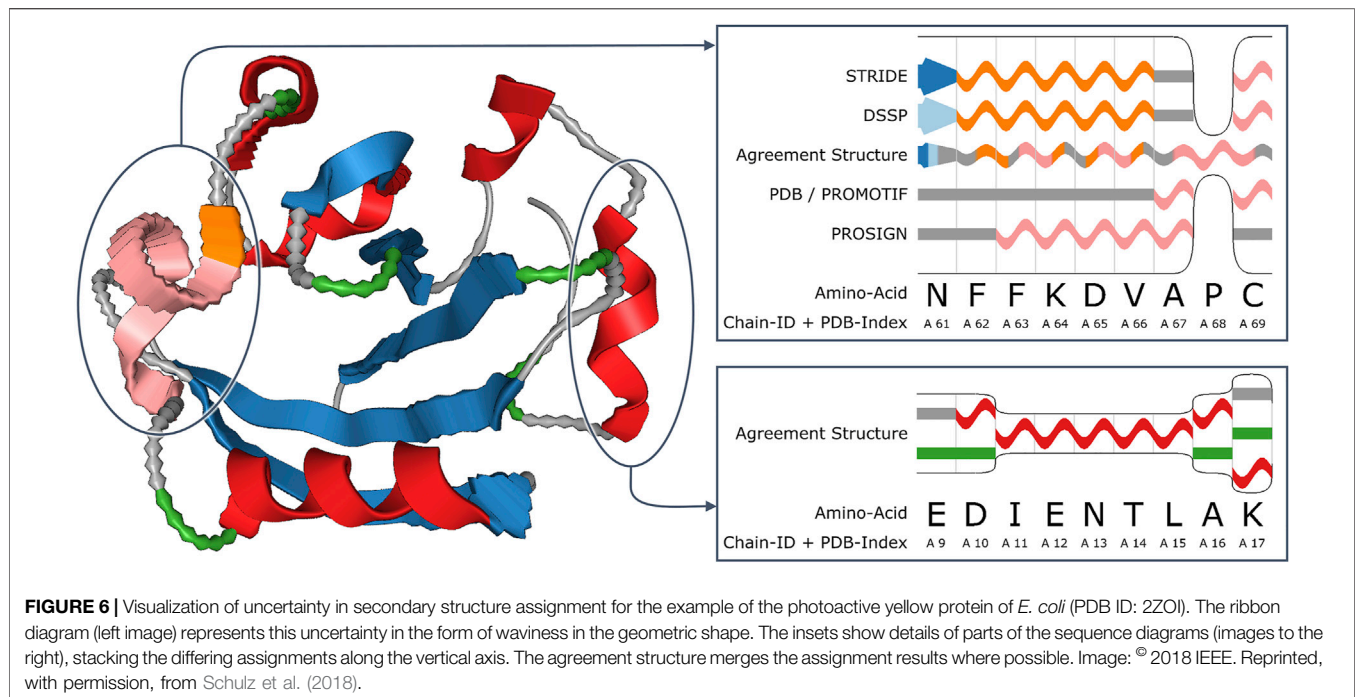
glyphs can be used to represent the range of vector quantities at respective locations in a vector field (Hlawatsch et al., 2011). Furthermore, the concept of displaying ranges can be extended to rather complex geometric representations, for example, in order to visualize confidence intervals for fiber tracking for showing 3D brain structures (Brecheisen et al., 2013).

In general, range plots provide a representation of summarizing characteristics of uncertainty and are rooted in well-known visual representations from statistical graphics. Therefore, they can be used without much learning required by recipients of the visualization. Another advantage is that ranges show quantitative information about summary statistics. However, there is a caveat: as mentioned before, even traditional error bars might be misinterpreted (Belia et al., 2005). Furthermore, the principle of showing distinct ranges can lead to the wrong interpretations because they might lead to introducing false categorical boundaries, e.g., inside vs outside regions (Padilla et al., 2020). Finally, range-based visualizations tend to need substantial extra space on the visualization image that might not be available.

4.4 Summary Statistics in Visual Variables and Glyphs

We can still use characteristic quantities from summary statistics, but now map them to visual channels, such as color, brightness, texture characteristics, etc. There are many different design choices for this mapping, with different characteristics and effectiveness for uncertainty visualization. Most of these mappings focus on including the variability of the input data into the visual representation.

For example, MacEachren et al. (2012) link visual channels for uncertainty representation to the semiology of graphics by Bertin (1983). Visual variables (also called retinal variables by Bertin) describe a set of visual primitives from which we can construct a visualization. MacEachren et al. (2012) investigate the following visual variables according to their usefulness for uncertainty visualization in terms of intuitiveness and task performance (focusing on map reading): location, size, color hue, color value, color saturation, orientation, grain, arrangement, shape, fuzziness, and transparency. These exhibit different adeptness for uncertainty visualization, for example, fuzziness shows a high level of intuitiveness in their study.



These visual variables are only one approach to structure the design space. Boukhelifa et al. (2012) provide a grouping into three main categories: color-oriented approaches (hue, saturation, or brightness), focus-based methods (mapping uncertainty to contour crispness, transparency, or resolution), and geometric mapping (e.g., sketchiness in rendering, distorting line marks). Animation (for example, oscillating displays) can also be used to represent uncertainty (Pang et al., 1997).

In particular, if such mappings are used to modify larger graphical elements such as icons or glyphs, we have a quite large design space that allows us to represent uncertainty. For example, Vehlow et al. (2013) modify attributes of (larger) nodes to show uncertainty: by color gradients or alternatively by star-shaped icons. In another application, glyphs are designed to represent the distribution of fibers (Schultz et al., 2013).

Figure 6 shows an example of a 3D visualization using waviness to represent uncertainty, here for the uncertainty that comes from disagreement in secondary structure assignments (Schulz et al., 2018). Alternative visualization methods for the uncertainty in secondary structure assignments are discussed by Hamada (2014). Another example of uncertainty visualization for proteins is by Maack et al. (2021), who address the visual representation of uncertainty in the conformation of proteins.

Uncertainty encoding in visual variables has the advantage that it can, if done appropriately, provide an intuitive visualization of uncertainty that integrates well in existing non-uncertainty visualization techniques because the original visualization technique might not be changed substantially. However, these visualization techniques tend to focus on rather qualitative representations; it is usually hard to read off

accurate uncertainty information. Another issue is that there can be conflicts in choosing the visual variables: one has to balance between the need for a good visual representation of uncertainty and the other kinds of information that should be shown in the visualization. Also, one has to be careful that there might be (negative) interactions between visual variables that can make it hard to include uncertainty information in an existing visualization.

In summary, this mapping approach needs careful design but can lead to good qualitative overview visualizations.

4.5 Uncertainty as Additional Data Dimension

The above approaches to including summary statistics essentially use different visual mappings to integrate the additional information that comes with summary statistics. To this end, they employ different variants of visual mappings.

However, we can also cast the problem of uncertainty visualization into the problem of multivariate visualization. For example, let us consider the case of data with n data attributes or dimensions. And let us assume that each data dimension comes with uncertainty described by one measure of variability (e.g., standard error of means). Then, we just increase the dimensionality of the data from n to $2n$ to represent, for example, both the means and the standard error of means. From this perspective, we have transformed the problem of n -D visualization (for precise data) to the problem of $2n$ -D visualization (for uncertain data). Therefore, we can apply standard visualization techniques that can deal with multiple data dimensions (Wong and Bergeron, 1994), such as

parallel coordinates (Inselberg, 1985; Heinrich and Weiskopf, 2013) or scatterplot matrices.

The advantage of this approach is that it can readily use existing visualization techniques and, thus, there is no or only little extra effort required. Another advantage is that many of these visualization techniques support accurate visualization. For example, parallel coordinates or scatterplots let us read off quantitative information accurately from the diagrams, which is in contrast to the more qualitative visualizations in the previous subsection. The important disadvantage is that we lose the nature of uncertainty in the visualization: there is no intuitive connection to variability. Therefore, this approach is less useful for conveying uncertainty in visual communication, and it can be prone to misinterpretations even by expert analysts.

4.6 Hybrid Visualizations and Systems

The above visualization techniques can be used in combination or together with other non-uncertainty visualizations, leading to hybrid visualizations. One strategy is to build a composition of a larger visualization that combines different visual representations. Often, the uncertainty visualization is placed next to the usual, non-uncertainty visualization. For example, Holzhüter et al. (2012) use an explicit representation of uncertainty with additional bar charts placed next to the actual visualization to show uncertainty from the visualization of biological expression data. Typical strategies use juxtaposition of visualizations or overlays to perform the composition. The summary plot (Potter et al., 2010), for example, integrates a box plot, histogram, a display of statistical moments, and a plot of the distribution.

Another common strategy employs multiple coordinated views (Baldonado et al., 2000) to link separate visualization views, often in connection with brushing and linking (Becker and Cleveland, 1987). Multiple coordinated views are popular in larger visualization or visual analytics systems because they allow us to represent data from different angles.

Hybrid visualizations, in particular, multiple coordinated views, are quite common and useful for uncertainty visualization because they allow us to reduce the complexity of each individual visualization, which is especially important for the increased difficulty that comes with including uncertainty in the visualization. However, we have to be careful that we do not overload the user with too complex combinations and hard-to-handle interactions. Therefore, attention needs to be paid to an appropriate design of the visualization and interaction.

5 EXAMPLE: GRAPH VISUALIZATION

We want to illustrate the aforementioned concepts for the example of graph visualization, with a focus on node-link diagrams. There are several reasons for choosing this example: 1) It is a rather complex kind of visualization already for the traditional non-uncertainty case. Therefore, it serves to show what challenges and opportunities arise with advanced uncertainty visualization. 2) It is an example of visualization of abstract data (often referred to as information visualization),

which is less well explored than uncertainty visualization for scalar or tensor fields (as in scientific visualization). Therefore, this example illustrates the current developments in uncertainty visualization. 3) Graphs are a versatile form of data representation with manifold uses in bioinformatics and beyond. Therefore, there is direct relevance for applications in biological data visualization.

Graph visualization is a large subfield of visualization, with many techniques available; *see*, for example, Battista et al. (1998), von Landesberger et al. (2011), and Beck et al. (2017) for background information.

Our first example (Vehlow et al., 2012) aims at the visualization of biochemical reaction networks. Such networks play a role in understanding certain cell functions or diseases. Our first step is to interface with the underlying modeling of the system and data acquisition (the early steps of the visualization pipeline; *see* Section 3). In this example, the modeling is circled around ordinary differential equations (ODEs) that are connected in the form of a directed graph. Vertices of the graph represent species and edges correspond to reactions. Besides regular edges, there might be hyper-edges representing regulatory interactions. Uncertainty is introduced by noise in measurements and, subsequently, by the uncertainty that comes with Bayesian parameter estimation.

From the visualization perspective, we are dealing with data in the form of a graph with uncertain and time-dependent attributes on the graph's vertices and edges, where time dependency comes from the temporal evolution of the reactions. Figure 7 shows a snapshot from a visualization system that facilitates the uncertainty-aware visual analysis of such kind of data. It takes the general approach of multiple coordinated views with brushing-and-linking (Section 4.6) to present the data from different angles and with different levels of detail. The node-link graph visualization (Figure 7 (1)) shows the topological structure of the graph and includes the visualization of uncertainty for edge and vertex attributes *via* color-coding of respective standard deviations; therefore, the uncertainty visualization uses a visual variable (here, color) to represent summary statistics (here, standard deviation); *see* Section 4.4. The same color-coding is used to show uncertainty in a detail view (Figure 7 (4)).

The visualization system also includes bands around temporal function plots (Figure 7 (6), (7)), implementing a range visualization of summary statistics; *see* Section 4.3. Furthermore, there is an explicit display of value distributions in the form of histograms (Figure 7 (3)), again focusing on selected details; *see* Section 4.1. Value distributions are also shown in an overlay of sample points in a scatterplot (Figure 7 (2)); *see* Section 4.2. Finally, there is additional data processing and extraction of information that is aligned with uncertainty-affected input: fitting of axes due to principal component analysis (Figure 7 (2)) and correlation according to Pearson coefficients (Figure 7 (5)).

This example demonstrates that multiple different perspectives are often required to obtain a comprehensive view and analysis of uncertain data. The different views are also needed to support a variety of analysis tasks. In this example, the system

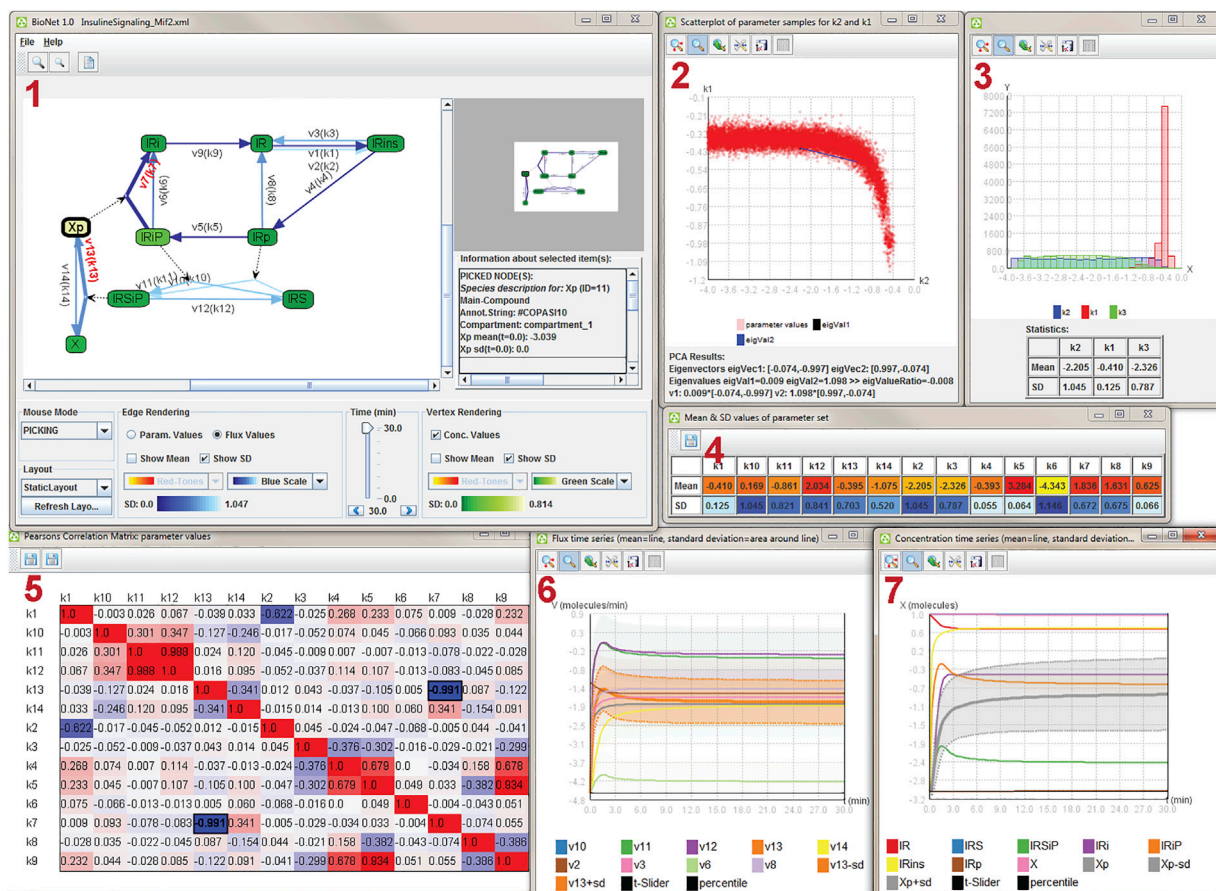


FIGURE 7 | Coordinated multiple views for uncertainty visualization, analyzing an insulin signaling model. Image: © 2012 IEEE. Reprinted, with permission, from Vehlow et al. (2012).

was developed and evaluated in collaboration with domain experts.

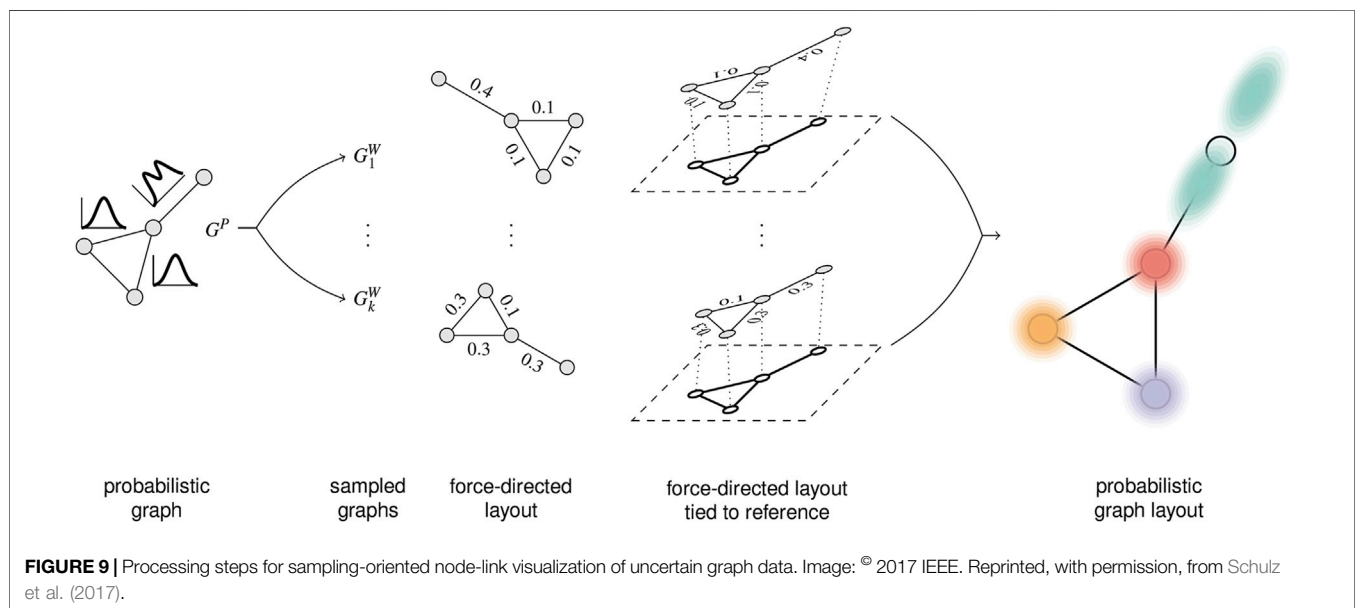
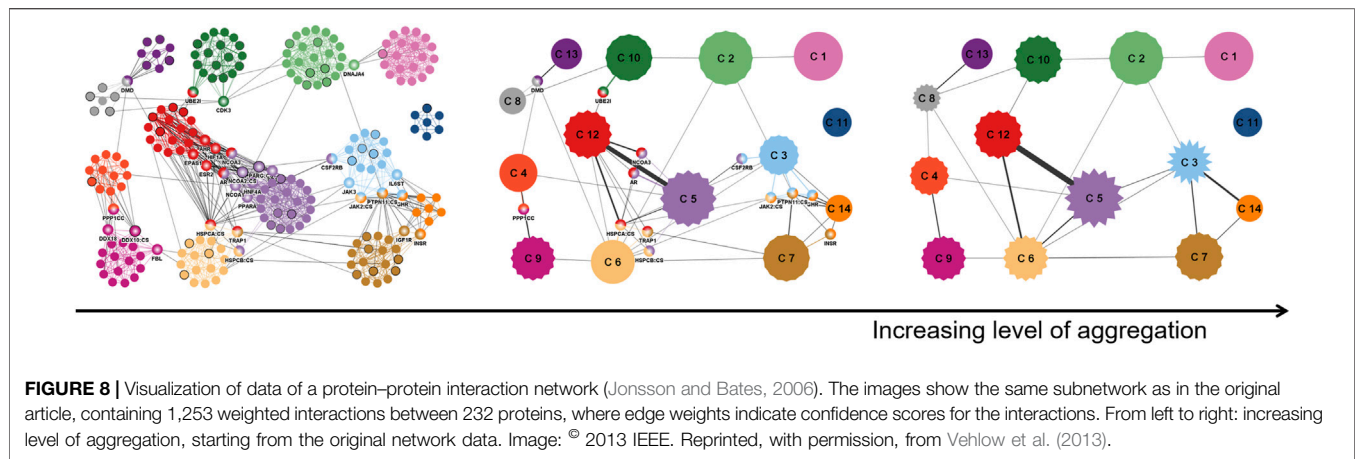
The next example shows uncertainty visualization for the case where uncertainty is introduced not at the data acquisition stage, but only later during the visualization pipeline in the transformation stage (Vehlow et al., 2013). Here, graph clustering (i.e., community detection) is applied to facilitate data analysis of a protein–protein interaction network on different levels of granularity: graph nodes are combined in groups that can be then shown by meta-nodes representing groups of nodes. Uncertainty is introduced by applying fuzzy clustering, which can lead to the gradual membership of a node in several groups.

In this example, the amount of uncertainty associated with grouping in a meta-node is represented by the amplitude of spikes in star-shaped icons (see the middle and right image in Figure 8), i.e., summary statistics is represented in a visual variable of the icon. In addition, original nodes may belong to several fuzzy clusters; here, the certainty of membership is shown again by a visual variable, now in the form of a color gradient within a node (several examples in the left image in Figure 8). Besides the visual mapping to visual variables, the layout of the network has to incorporate the information from fuzzy clustering, i.e., the mapping stage of the visualization pipeline has to be aware of the uncertainty model.

The previous two examples have focused the graph visualization aspect on showing summary statistics *via* visual variables. Our third example shifts the focus: how does uncertainty in edge attributes affect the geometry of the node-link diagram? The uncertainty model assumes distributions of weights on edges. Differing edge weights should influence the length of the edge. Therefore, the layout has to incorporate the variability of the weights.

A probabilistic graph layout (Schulz et al., 2017) achieves uncertainty visualization by showing distributions implicitly *via* overlay. Figure 9 illustrates the processing steps. First, we need a model of the probabilistic graph. Here, one has to consider whether there are dependencies between the probability density functions for the weights on the different edges. With this uncertainty model, we can then draw samples: these samples are complete graphs with edge weights, albeit each weight is now a fixed value that comes from drawing the sample. The next step produces a graph layout independently for each of the graph samples, here *via* a force-directed graph layout.

As already discussed in Section 4.2, registration or alignment is needed if the individual visualizations do not fit together. This is the case with many graph layout results. Therefore, we need an alignment step, here implemented by tying the individual layouts



to a reference layout. In other words, an appropriate layout is a key component in this kind of uncertainty visualization.

The final step renders the overlay of the individual graph visualizations. The basic idea is to perform blending of the individual images. However, this approach would lead to problems caused by visual clutter. Therefore, a combination of splatting nodes, curve bundling for the edges, and adapted node coloring and clustering is used.

Figure 10 shows an example of probabilistic graph visualization for protein-protein interactions. The edge weights are derived from scores computed from data from the STRING database.⁴ The comparison between the traditional visualization without uncertainty (left image in **Figure 10**) and the one that incorporates uncertainty (right

image in **Figure 10**) demonstrates varying levels of (un-)certainty associated with the different interactions.

This example is based on an overlay resulting in a static image. By exchanging the last part of the processing pipeline, one could also use small multiple or animation to show the individual graph visualizations coming from the sampling process. For example, Zhang et al. (2022) present and discuss a method based on animation.

The sampling approach essentially reduces the problem of uncertainty visualization to the visualization of many individual samples. **Figure 11** illustrates the process.

We start with the uncertainty model in the form of probability density functions or similar probabilistic descriptions. From these, points—in a potentially abstract and complex space—are produced by sampling (e.g., Monte-Carlo random sampling, quasi-Monte-Carlo sampling, etc.) and mapped to intermediate images by applying regular non-uncertainty-oriented visualization. In the last step, the images are overlaid to generate the final visualization. As in the example of probabilistic graph

⁴<https://string-db.org/>

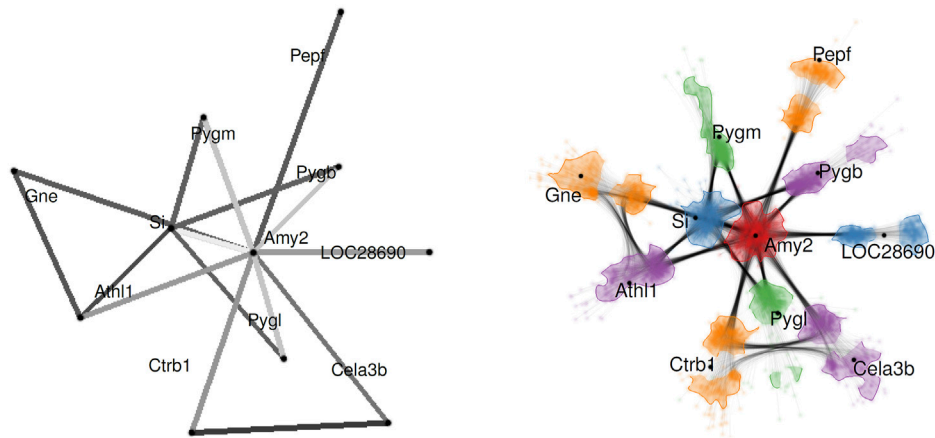


FIGURE 10 | Probabilistic graph layout for visualizing protein-protein interactions for pancreatic alpha-amylase (Amy2). The left image shows the expected (average) graph, i.e., traditional non-uncertainty visualization. The right image shows the uncertainty visualization. Image: © 2017 IEEE. Reprinted, with permission, from Schulz et al. (2017).

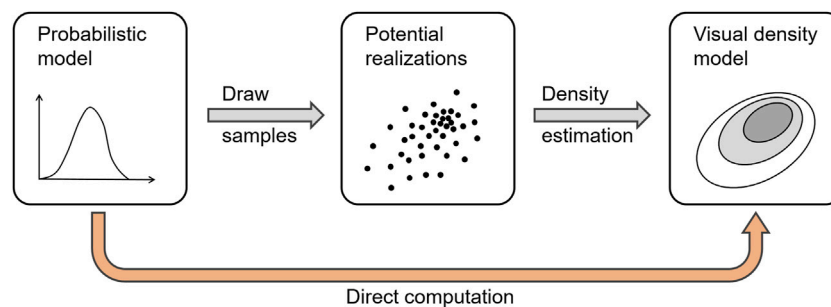


FIGURE 11 | Process of sampling and density estimation for the implicit visualization of distributions.

visualization, this last step aims to generate a density representation (here, of nodes and edges), e.g., by employing kernel density estimation. Therefore, the process essentially performs a discretization into points and then a reconstruction of a density field, i.e., a numerical approximation with several potential sources for errors and required parameter choices.

Ideally, we would avoid the construction of in-between samples and, instead, directly go from the probability description of the data to the density model of the visual output. This can be readily done when there is no registration needed, such as for typical cases of scientific visualization with given spatial embedding. For example, the probability where an isosurface cuts through the volume can then be mapped to density, which can be rendered by color-coding (Pöthkow and Hege, 2011). However, when the visual mapping implies more complex transformations, the density computation becomes more difficult. For certain scenarios of multidimensional data, there are techniques that construct density plots for parallel coordinates and scatterplots (Bachthaler and Weiskopf, 2008; Heinrich and Weiskopf, 2009; Heinrich et al., 2011) that carry over to respective uncertainty plots (Zheng and Sadlo, 2021). However, developing similar techniques for other advanced

examples of uncertainty visualization remains a largely unsolved problem so far.

6 DISCUSSION

We have surveyed concepts, strategies, and methods for uncertainty visualization—mostly from the perspective of visualization research. This section discusses general observations, open questions, and directions for future research. In addition, we link this discussion to recommendations geared toward use in applications of biological data visualization.

6.1 Open Questions and Future Directions in Visualization Research

We have seen that there has been quite some progress in uncertainty visualization, leading to a large variety of available techniques. However, we have also discussed that uncertainty visualization is challenging due to the difficult, yet relevant interplay of many different components in the visualization process. Therefore, there are a number of directions for future research.

Layout is key to advanced visual mappings. One issue with integrating uncertainty information in an already complex visualization is the lack of space, for example, to place glyphs, integrate range representations, use waviness or sketchiness of larger visual marks, etc. Here, the layout process essentially needs to balance the different and conflicting requirements from showing complex data and its uncertainties. Visualization space is a scarce resource in this respect. The example of probabilistic graph visualization exhibits another layout problem: the one of aligning or registering individual visualization images. Therefore, future progress in the visual mapping of uncertainty is related to developing appropriate layout methods that optimize for potentially conflicting goals.

Perception, cognition, and evaluation. Understanding how we perceive visualization and reason with it is a central problem in visualization in general; and this problem is even harder when we include uncertainty. Therefore, this topic will continue to play a highly relevant role in uncertainty visualization, and it is tightly connected to ways of how uncertainty visualization is evaluated, e.g., from the user perspective.

Uncertainty visualization literacy. There is the general issue of visualization literacy, i.e., dealing with how people can generate and read visualizations. With the progress in uncertainty visualization techniques comes the opportunity of working on improving respective literacy. Due to the difficulties that users have with many visual representations of uncertainty, there is a great potential from the interplay between improving visualization techniques and teaching skillsets.

Interacting with uncertainty visualization. We have touched on some examples of interaction techniques geared toward the process of uncertainty visualization. However, this topic is largely untapped so far. We see great potential for future research on interaction methods that will have to include the perceptual and cognitive aspects discussed above.

Integration with machine learning and explainable AI. The major trend toward including machine learning also manifests itself in uncertainty visualization. Here, the special interest is in assessing and visually communicating the uncertainty associated with automatic data analysis and machine learning, which also links to visualization as a means to support explainable artificial intelligence (AI).

Frameworks and software integration. A message from the consideration of the complete visualization pipeline is: it is not sufficient to just look at stages of the pipeline separately. For example, it is not enough to only consider visual mappings of uncertainty. Instead, there is a need for frameworks that provide a unified perspective. There is already some work on frameworks and integration (e.g., Correa et al. (2009), Wu et al. (2012), and Sacha et al. (2016)), but with the progress coming from the other topics listed above, the frameworks will need to be adapted and extended. In particular, there is the challenge of including the user in the combined process of human-machine visual data analysis. A practical problem is the lack of uncertainty visualization techniques in many existing software systems. Available implementations of uncertainty visualization are often

restricted to individual and separate research prototypes. Therefore, there is the need for extended software systems supporting uncertainty visualization.

6.2 Recommendations

The lack of widespread implementations of uncertainty visualization is one issue that makes it hard to include it in applications of biological data visualization. Still, there are opportunities for practical impact of uncertainty visualization on bioinformatics applications. Some of the following recommendations might facilitate the integration of uncertainty visualization in such applications.

Think about data modeling and the context of the visualization process. An important early step is to understand the data and uncertainty model, which naturally has to be deeply rooted in the application at hand. The next step is to consider the tasks that should be solved with visualization and how they might be affected by data uncertainty. To this end, interdependencies between the components for data acquisition, processing, and visualization should be taken into account, including propagation of uncertainty. Here, rough estimates or models might be sufficient for a coarse description of the interdependencies, and these might be done completely outside of visualization software systems.

Focus on main players for uncertainty. Although we argued for the importance of considering the whole visualization process, it is clear that not all stages are equally important for each application. Instead, it is better to focus the attention on the main sources and effects of uncertainty. Then, only these parts of the whole process might have to be extended from regular non-uncertainty processing to an uncertainty-aware counterpart. This approach can reduce the effort substantially, especially when there is no comprehensive uncertainty visualization system available.

Choose appropriate visualization techniques. In general, visualizations should be chosen to match data characteristics, tasks, and intended audience. Usually, there is not a single-best method. This statement is especially true for uncertainty visualization. For example, existing multivariate data visualization might be enough for your own internal processes of data analysis, but not for effective communication to a broader outside audience. The choice of visualization technique might also be related to the availability of implementations (or lack thereof). Some visual mappings are easier to integrate into existing non-uncertainty-oriented visualization techniques than others. For example, per-pixel visual variables like color or others tend to be easy to integrate into existing non-uncertainty-oriented visualization systems; it might be as simple as modifying the color map or extending multivariate visualization in parallel coordinates with additional data axes. Other uncertainty mappings require much more work, for example, when there is a serious impact on the layout or when comprehensive systems have to be changed for a full visual analytics framework for uncertainty. Such efforts in modifying or implementing visualization techniques should play a role in choosing appropriate techniques.

Need for integration in existing software. In general, there is a lack of comprehensive uncertainty support in existing visualization software in many bioinformatics applications. Therefore, some community effort could help with including more of the uncertainty-aware stages of the visualization pipeline.

Uncertainty awareness. Due to the complexity of uncertainty visualization, there might not be a single and comprehensive solution. Instead, the main goal of this paper is increased awareness of issues that come with uncertainty in visualization.

AUTHOR CONTRIBUTIONS

The author confirms being the sole contributor of this work.

REFERENCES

- Bachthaler, S., and Weiskopf, D. (2008). Continuous Scatterplots. *IEEE Trans. Vis. Comput. Graph.* 14, 1428–1435. doi:10.1109/TVCG.2008.119
- Baldonado, M. Q. W., Woodruff, A., and Kuchinsky, A. (2000). “Guidelines for Using Multiple Views in Information Visualization,” in Working Conference on Advanced Visual Interfaces (AVI 2000) (New York, NY: Association for Computing Machinery), 110–119. doi:10.1145/345513.345271
- Baraldi, A., and Blonda, P. (1999). A Survey of Fuzzy Clustering Algorithms for Pattern Recognition. I. *IEEE Trans. Syst. Man. Cybern. B Cybern.* 29, 778–785. doi:10.1109/3477.809032
- Battista, G. D., Eades, P., Tamassia, R., and Tollis, I. G. (1998). *Graph Drawing: Algorithms for the Visualization of Graphs*. Hoboken, NJ: Prentice-Hall.
- Beck, F., Burch, M., Diehl, S., and Weiskopf, D. (2017). A Taxonomy and Survey of Dynamic Graph Visualization. *Comput. Graph. Forum* 36, 133–159. doi:10.1111/cgf.12791
- Becker, R. A., and Cleveland, W. S. (1987). Brushing Scatterplots. *Technometrics* 29, 127–142. doi:10.1080/00401706.1987.10488204
- Belia, S., Fidler, F., Williams, J., and Cumming, G. (2005). Researchers Misunderstand Confidence Intervals and Standard Error Bars. *Psychol. Methods* 10, 389–396. doi:10.1037/1082-989X.10.4.389
- Bertin, J. (1983). *Semiology of Graphics: Diagrams, Networks, Maps*. Madison, WI: University of Wisconsin Press.
- Bonneau, G.-P., Hege, H.-C., Johnson, C. R., Oliveira, M. M., Potter, K., Rheingans, P., et al. (2014). “Overview and State-of-the-Art of Uncertainty Visualization,” in *Scientific Visualization*. Editors C. D. Hansen, M. Chen, C. R. Johnson, A. E. Kaufman, and H. Hagen (Berlin, New York: Springer), 3–27. doi:10.1007/978-1-4471-6497-5_1
- Boukhelifa, N., Bezerianos, A., Isenberg, T., and Fekete, J. (2012). Evaluating Sketchiness as a Visual Variable for the Depiction of Qualitative Uncertainty. *IEEE Trans. Vis. Comput. Graph.* 18, 2769–2778. doi:10.1109/TVCG.2012.220
- Boukhelifa, N., and Duke, D. J. (2009). “Uncertainty Visualization: Why Might It Fail?,” in International Conference on Human Factors in Computing Systems (CHI 2009), Extended Abstracts Volume, Boston, MA, April 4–9, 2009. Editors D. R. Olsen Jr., R. B. Arthur, K. Hinckley, M. R. Morris, S. E. Hudson, and S. Greenberg (ACM), 4051–4056. doi:10.1145/1520340.1520616
- Brecheisen, R., Platel, B., ter Haar Romeny, B. M., and Vilanova, A. (2013). Illustrative Uncertainty Visualization of DTI Fiber Pathways. *Vis. Comput.* 29, 297–309. doi:10.1007/s00371-012-0733-9
- Brodie, K., Allendes Osorio, R., and Lopes, A. (2012). “A Review of Uncertainty in Data Visualization,” in *Expanding the Frontiers of Visual Analytics and Visualization*. Editors J. Dill, R. A. Earnshaw, D. J. Kasik, J. A. Vince, and P. C. Wong (London: Springer), 81–109. doi:10.1007/978-1-4471-2804-5_6
- Chi, E. H., and Riedl, J. T. (1998). “An Operator Interaction Framework for Visualization Systems,” in IEEE Symposium on Information Visualization (InfoVis ’98), Research Triangle, CA, USA, October 19–20, 1998 (IEEE Computer Society), 63–70. doi:10.1109/INFVIS.1998.729560
- Correa, C. D., Chan, Y.-H., and Ma, K.-L. (2009). “A Framework for Uncertainty-Aware Visual Analytics,” in IEEE Symposium on Visual Analytics Science and Technology (IEEE VAST 2009), Atlantic City, NJ, October 12–13, 2009 (IEEE Computer Society), 51–58. doi:10.1109/VAST.2009.5332611
- Correll, M., and Gleicher, M. (2014). Error Bars Considered Harmful: Exploring Alternate Encodings for Mean and Error. *IEEE Trans. Vis. Comput. Graph.* 20, 2142–2151. doi:10.1109/TVCG.2014.2346298
- Deitrick, S., and Edsall, R. (2006). “The Influence of Uncertainty Visualization on Decision Making: An Empirical Evaluation,” in *Progress in Spatial Data Handling: 12th International Symposium on Spatial Data Handling*. Editors A. Riedl, W. Kainz, and G. A. Elmes (Berlin, Heidelberg: Springer), 719–738. doi:10.1007/3-540-35589-8_45
- Demir, I., Dick, C., and Westermann, R. (2014). Multi-charts for Comparative 3D Ensemble Visualization. *IEEE Trans. Vis. Comput. Graph.* 20, 2694–2703. doi:10.1109/TVCG.2014.2346448
- Ehlschlaeger, C. R., Shortridge, A. M., and Goodchild, M. F. (1997). Visualizing Spatial Data Uncertainty Using Animation. *Comput. Geosciences* 23, 387–395. doi:10.1016/S0098-3004(97)00005-8
- Görtler, J., Spinner, T., Streeb, D., Weiskopf, D., and Deussen, O. (2020). Uncertainty-aware Principal Component Analysis. *IEEE Trans. Vis. Comput. Graph.* 26, 822–831. doi:10.1109/TVCG.2019.2934812
- Görtler, J. (2021). *Quantitative Methods for Uncertainty Visualization*. PhD thesis. Konstanz (Germany): University of Konstanz.
- Greis, M., Schuff, H., Kleiner, M., Henze, N., and Schmidt, A. (2017). Input Controls for Entering Uncertain Data. *Proc. ACM Hum.-Comput. Interact.* 1, 3:1–3:17. doi:10.1145/3095805
- Griethel, H., and Schumann, H. (2006). “The Visualization of Uncertain Data: Methods and Problems,” in *Simulation und Visualisierung (SimVis 2006)*, 143–156.
- Haber, R. B., and McNabb, D. A. (1990). “Visualization Idioms: A Conceptual Model for Visualization Systems,” in *Visualization in Scientific Computing*. Editors G. M. Nielson, B. D. Shriver, and L. J. Rosenblum (Los Alamitos, CA: IEEE Computer Society Press), 74–93.
- Hamada, M. (2014). Fighting Against Uncertainty: An Essential Issue in Bioinformatics. *Brief. Bioinform.* 15, 748–767. doi:10.1093/bib/bbt038
- Heinrich, J., and Weiskopf, D. (2009). Continuous Parallel Coordinates. *IEEE Trans. Vis. Comput. Graph.* 15, 1531–1538. doi:10.1109/TVCG.2009.131
- Heinrich, J., Bachthaler, S., and Weiskopf, D. (2011). Progressive Splatting of Continuous Scatterplots and Parallel Coordinates. *Comput. Graph. Forum* 30, 653–662. doi:10.1111/j.1467-8659.2011.01914.x
- Heinrich, J., and Weiskopf, D. (2013). “State of the Art of Parallel Coordinates,” in *Eurographics 2013 – State of the Art Reports*, 95–116. doi:10.2312/conf/EG2013/stars/095-116
- Hlawatsch, M., Leube, P., Nowak, W., and Weiskopf, D. (2011). Flow Radar Glyphs—Static Visualization of Unsteady Flow with Uncertainty. *IEEE Trans. Vis. Comput. Graph.* 17, 1949–1958. doi:10.1109/TVCG.2011.203
- Höferlin, M., Höferlin, B., Weiskopf, D., and Heidemann, G. (2011). Uncertainty-aware Video Visual Analytics of Tracked Moving Objects. *J. Spat. Inf. Sci.* 2, 87–117. doi:10.5311/JOSIS.2010.2.1

FUNDING

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 251654672 – TRR 161 (Project A01). The publication was supported by the Open Access Publishing Fund of the University of Stuttgart.

ACKNOWLEDGMENTS

I would like to thank my collaborators in Project A01 of the TRR 161—Oliver Deussen, Christoph Schulz, and Jochen Görtler—for our joint work and discussions on uncertainty visualization that built the basis for many of the papers reviewed in this manuscript.

- Holzhüter, C., Lex, A., Schmalstieg, D., Schulz, H.-J., Schumann, H., and Streit, M. (2012). "Visualizing Uncertainty in Biological Expression Data," in *Visualization and Data Analysis (VDA 2012)* (Bellingham, WA: SPIE), 82940O. doi:10.1117/12.908516
- Hullman, J., Qiao, X., Correll, M., Kale, A., and Kay, M. (2019). In Pursuit of Error: A Survey of Uncertainty Visualization Evaluation. *IEEE Trans. Vis. Comput. Graph.* 25, 903–913. doi:10.1109/TVCG.2018.2864889
- Inselberg, A. (1985). The Plane with Parallel Coordinates. *Vis. Comput.* 1, 69–91. doi:10.1007/BF01898350
- Jena, A., Engelke, U., Dwyer, T., Raiamanickam, V., and Paris, C. (2020). "Uncertainty Visualisation: An Interactive Visual Survey," in IEEE Pacific Visualization Symposium (PacificVis 2020), Tianjin, China, June 3–5, 2020 (IEEE), 201–205. doi:10.1109/PacificVis48177.2020.1014
- Johnson, C. (2004). Top Scientific Visualization Research Problems. *IEEE Comput. Graph. Appl.* 24, 13–17. doi:10.1109/MCG.2004.20
- Jonsson, P. F., and Bates, P. A. (2006). Global Topological Features of Cancer Proteins in the Human Interactome. *Bioinform.* 22, 2291–2297. doi:10.1093/bioinformatics/btl390
- Kale, A., Nguyen, F., Kay, M., and Hullman, J. (2019). Hypothetical Outcome Plots Help Untrained Observers Judge Trends in Ambiguous Data. *IEEE Trans. Vis. Comput. Graph.* 25, 892–902. doi:10.1109/TVCG.2018.2864909
- Kamal, A., Dhakal, P., Javaid, A. Y., Devabhaktuni, V. K., Kaur, D., Zaients, J., et al. (2021). Recent Advances and Challenges in Uncertainty Visualization: A Survey. *J. Vis.* 24, 861–890. doi:10.1007/s12650-021-00755-1
- Lam, H., Bertini, E., Isenberg, P., Plaisant, C., and Carpendale, S. (2012). Empirical Studies in Information Visualization: Seven Scenarios. *IEEE Trans. Vis. Comput. Graph.* 18, 1520–1536. doi:10.1109/TVCG.2011.279
- Lee, J. A., and Verleysen, M. (2007). *Nonlinear Dimensionality Reduction*. New York, NY: Springer-Verlag. doi:10.1007/978-0-387-39351-3
- Maack, R. G. C., Raymer, M. L., Wischgoll, T., Hagen, H., and Gillmann, C. (2021). A Framework for Uncertainty-Aware Visual Analytics of Proteins. *Comput. Graph.* 98, 293–305. doi:10.1016/j.cag.2021.05.011
- MacEachren, A. M., Roth, R. E., O'Brien, J., Li, B., Swingley, D., and Gahegan, M. (2012). Visual Semiotics & Uncertainty Visualization: An Empirical Study. *IEEE Trans. Vis. Comput. Graph.* 18, 2496–2505. doi:10.1109/TVCG.2012.279
- MacEachren, A. M., Robinson, A., Hopper, S., Gardner, S., Murray, R., Gahegan, M., et al. (2005). Visualizing Geospatial Information Uncertainty: What We Know and What We Need to Know. *Cartography Geogr. Inf. Sci.* 32, 139–160. doi:10.1559/1523040054738936
- Mirzargar, M., Whitaker, R. T., and Kirby, R. M. (2014). Curve Boxplot: Generalization of Boxplot for Ensembles of Curves. *IEEE Trans. Vis. Comput. Graph.* 20, 2654–2663. doi:10.1109/TVCG.2014.2346455
- Murray, P., McGee, F., and Forbes, A. G. (2017). A Taxonomy of Visualization Tasks for the Analysis of Biological Pathway Data. *BMC Bioinform.* 18, 1–21. doi:10.1186/s12859-016-1443-5
- Nonato, L. G., and Aupetit, M. (2019). Multidimensional Projection for Visual Analytics: Linking Techniques with Distortions, Tasks, and Layout Enrichment. *IEEE Trans. Vis. Comput. Graph.* 25, 2650–2673. doi:10.1109/TVCG.2018.2846735
- Padilla, L., Kay, M., and Hullman, J. (2020). Uncertainty Visualization. *PsyArXiv*. doi:10.31234/osf.io/ebd6r
- Padilla, L. M. K., Powell, M., Kay, M., and Hullman, J. (2021). Uncertain About Uncertainty: How Qualitative Expressions of Forecaster Confidence Impact Decision-Making with Uncertainty Visualizations. *Front. Psychol.* 11, 3747. doi:10.3389/fpsyg.2020.579267
- Pang, A. T., Wittenbrink, C. M., and Lodha, S. K. (1997). Approaches to Uncertainty Visualization. *Vis. Comput.* 13, 370–390. doi:10.1007/s003710050111
- Pöthkow, K., and Hege, H. C. (2011). Positional Uncertainty of Isocontours: Condition Analysis and Probabilistic Measures. *IEEE Trans. Vis. Comput. Graph.* 17, 1393–1406. doi:10.1109/TVCG.2010.247
- Potter, K., Kniss, J., Riesenfeld, R., and Johnson, C. R. (2010). Visualizing Summary Statistics and Uncertainty. *Comput. Graph. Forum* 29, 823–832. doi:10.1111/j.1467-8659.2009.01677.x
- Potter, K., Rosen, P., and Johnson, C. R. (2012). From Quantification to Visualization: A Taxonomy of Uncertainty Visualization Approaches. *IFIP Adv. Inf. Commun. Technol.* 377, 226–249. doi:10.1007/978-3-642-32677-6_15
- Ristovski, G., Preusser, T., Hahn, H. K., and Linsen, L. (2014). Uncertainty in Medical Visualization: Towards A Taxonomy. *Comput. Graph.* 39, 60–73. doi:10.1016/j.cag.2013.10.015
- Robertson, G., Fernandez, R., Fisher, D., Lee, B., and Stasko, J. (2008). Effectiveness of Animation in Trend Visualization. *IEEE Trans. Vis. Comput. Graph.* 14, 1325–1332. doi:10.1109/TVCG.2008.125
- Rodrigues, N., and Weiskopf, D. (2018). Nonlinear Dot Plots. *IEEE Trans. Vis. Comput. Graph.* 24, 616–625. doi:10.1109/TVCG.2017.2744018
- Rosen, P., Burton, B., Potter, K., and Johnson, C. R. (2016). "Muview: A Visual Analysis System for Exploring Uncertainty in Myocardial Ischemia Simulations," in *Visualization in Medicine and Life Sciences III, Towards Making an Impact*. Editors L. Linsen, B. Hamann, and H. Hege (Berlin, New York: Springer), 49–69. doi:10.1007/978-3-319-24523-2_3
- Sacha, D., Senaratne, H., Kwon, B. C., Ellis, G., and Keim, D. A. (2016). The Role of Uncertainty, Awareness, and Trust in Visual Analytics. *IEEE Trans. Vis. Comput. Graph.* 22, 240–249. doi:10.1109/TVCG.2015.2467591
- Sanyal, J., Zhang, S., Bhattacharya, G., Amburn, P., and Moorhead, R. J. (2009). A User Study to Compare Four Uncertainty Visualization Methods for 1D and 2D Datasets. *IEEE Trans. Vis. Comput. Graph.* 15, 1209–1218. doi:10.1109/TVCG.2009.114
- Schultz, T., Schlaffke, L., Schölkopf, B., and Schmidt-Wilcke, T. (2013). HiFiVE: A Hilbert Space Embedding of Fiber Variability Estimates for Uncertainty Modeling and Visualization. *Comput. Graph. Forum* 32, 121–130. doi:10.1111/cgf.12099
- Schulz, C., Nocaj, A., Goertler, J., Deussen, O., Brandes, U., and Weiskopf, D. (2017). Probabilistic Graph Layout for Uncertain Network Visualization. *IEEE Trans. Vis. Comput. Graph.* 23, 531–540. doi:10.1109/TVCG.2016.2598919
- Schulz, C., Schatz, K., Krone, M., Braun, M., Ertl, T., and Weiskopf, D. (2018). "Uncertainty Visualization for Secondary Structures of Proteins," in IEEE Pacific Visualization Symposium (PacificVis), 96–105. doi:10.1109/PacificVis.2018.00020
- Schulz, C. (2021). *Uncertainty-aware Visualization Techniques*. PhD thesis. Stuttgart (Germany): University of Stuttgart.
- Siddiqui, F., Höllt, T., and Vilanova, A. (2021). "Uncertainty in the DTI Visualization Pipeline," in *Anisotropy across Fields and Scales*. Editors E. Özarslan, T. Schultz, E. Zhang, and A. Fuster (Cham: Springer International Publishing), 125–148. doi:10.1007/978-3-030-56215-1_6
- Skeels, M., Lee, B., Smith, G., and Robertson, G. G. (2010). Revealing Uncertainty for Information Visualization. *Inf. Vis.* 9, 70–81. doi:10.1057/ivs.2009.1
- Tak, S., Toet, A., and van Erp, J. (2014). The Perception of Visual Uncertainty Representation by Non-experts. *IEEE Trans. Vis. Comput. Graph.* 20, 935–943. doi:10.1109/TVCG.2013.247
- Tufte, E. (1990). *Envisioning Information*. Cheshire, CT: Graphics Press.
- Vehlow, C., Reinhardt, T., and Weiskopf, D. (2013). Visualizing Fuzzy Overlapping Communities in Networks. *IEEE Trans. Vis. Comput. Graph.* 19, 2486–2495. doi:10.1109/TVCG.2013.232
- Vehlow, C., Hasenauer, J., Kramer, A., Heinrich, J., Radde, N., Allgöwer, F., et al. (2012). "Uncertainty-aware Visual Analysis of Biochemical Reaction Networks," in IEEE Symposium on Biological Data Visualization (BioVis), 91–98. doi:10.1109/BioVis.2012.6378598
- von Landesberger, T., Kuijper, A., Schreck, T., Kohlhammer, J., van Wijk, J. J., Fekete, J.-D., et al. (2011). Visual Analysis of Large Graphs: State-of-the-Art and Future Research Challenges. *Comput. Graph. Forum* 30, 1719–1749. doi:10.1111/j.1467-8659.2011.01898.x
- Wang, J., Hazarika, S., Li, C., and Shen, H.-W. (2019). Visualization and Visual Analysis of Ensemble Data: A Survey. *IEEE Trans. Vis. Comput. Graph.* 25, 2853–2872. doi:10.1109/TVCG.2018.2853721
- Ware, C. (2021). *Information Visualization: Perception for Design*. 4th Edn. Cambridge, MA: Morgan Kaufmann.
- Whitaker, R. T., Mirzargar, M., and Kirby, R. M. (2013). Contour Boxplots: A Method for Characterizing Uncertainty in Feature Sets from Simulation Ensembles. *IEEE Trans. Vis. Comput. Graph.* 19, 2713–2722. doi:10.1109/TVCG.2013.143
- Wilkinson, L. (1999). Dot Plots. *The Am. Statistician* 53, 276–281. doi:10.1080/00031305.1999.10474474

- Willis, A., and Bell, R. (2018). Uncertainty in Phylogenetic Tree Estimates. *J. Comput. Graph. Stat.* 27, 542–552. doi:10.1080/10618600.2017.1391697
- Wong, P. C., and Bergeron, R. D. (1994). “30 Years of Multidimensional Multivariate Visualization,” in *Scientific Visualization: Overviews, Methodologies, and Techniques*. Editors G. M. Nielson, H. Hagen, and H. Müller (Los Alamitos, CA: IEEE Computer Society), 3–33.
- Wu, Y., Yuan, G.-X., and Ma, K.-L. (2012). Visualizing Flow of Uncertainty through Analytical Processes. *IEEE Trans. Vis. Comput. Graph.* 18, 2526–2535. doi:10.1109/TVCG.2012.285
- Xu, R., and Wunsch II, D. (2005). Survey of Clustering Algorithms. *IEEE Trans. Neural Netw.* 16, 645–678. doi:10.1109/TNN.2005.845141
- Zhang, D., Adar, E., and Hullman, J. (2022). Visualizing Uncertainty in Probabilistic Graphs with Network Hypothetical Outcome Plots (NetHOPs). *IEEE Trans. Vis. Comput. Graph.* 28, 443–453. doi:10.1109/TVCG.2021.3114679
- Zheng, B., and Sadlo, F. (2021). Uncertainty in Continuous Scatterplots, Continuous Parallel Coordinates, and Fibers. *IEEE Trans. Vis. Comput. Graph.* 27, 1819–1828. doi:10.1109/TVCG.2020.3030466
- Zhou, L., Johnson, C. R., and Weiskopf, D. (2021). Data-driven Space-Filling Curves. *IEEE Trans. Vis. Comput. Graph.* 27, 1591–1600. doi:10.1109/TVCG.2020.3030473
- Zuk, T., and Carpendale, S. (2006). “Theoretical Analysis of Uncertainty Visualizations,” in *Proceedings of SPIE Visualization and Data Analysis (VDA 2006)*. Editors R. F. Erbacher, J. C. Roberts, M. T. Gröhn, and K. Börner (Bellingham, WA: SPIE), 6060, 606007. doi:10.1117/12.643631
- Conflict of Interest:** The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.
- Publisher’s Note:** All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.
- Copyright © 2022 Weiskopf. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



DJExpress: An Integrated Application for Differential Splicing Analysis and Visualization

Lina Marcela Gallego-Paez* and Jan Mauer*

BioMed X Institute (GmbH), Heidelberg, Germany

OPEN ACCESS

Edited by:

Sean O'Donoghue,
Garvan Institute of Medical Research,
Australia

Reviewed by:

Junfeng Xia,
Anhui University, China
Yoseph Barash,
University of Pennsylvania,
United States

*Correspondence:

Lina Marcela Gallego-Paez
linhiel@gmail.com
Jan Mauer
jan.mauer@gmail.com

Specialty section:

This article was submitted to
Data Visualization,
a section of the journal
Frontiers in Bioinformatics

Received: 30 September 2021

Accepted: 08 February 2022

Published: 24 February 2022

Citation:

Gallego-Paez LM and Mauer J (2022)
DJExpress: An Integrated Application
for Differential Splicing Analysis and
Visualization.
Front. Bioinform. 2:786898.
doi: 10.3389/fbinf.2022.786898

RNA-seq analysis of alternative pre-mRNA splicing has facilitated an unprecedented understanding of transcriptome complexity in health and disease. However, despite the availability of countless bioinformatic pipelines for transcriptome-wide splicing analysis, the use of these tools is often limited to expert bioinformaticians. The need for high computational power, combined with computational outputs that are complicated to visualize and interpret present obstacles to the broader research community. Here we introduce *DJExpress*, an R package for differential expression analysis of transcriptomic features and expression-trait associations. To determine gene-level differential junction usage as well as associations between junction expression and molecular/clinical features, *DJExpress* uses raw splice junction counts as input data. Importantly, *DJExpress* runs on an average laptop computer and provides a set of interactive and intuitive visualization formats. In contrast to most existing pipelines, *DJExpress* can handle both annotated and *de novo* identified splice junctions, thereby allowing the quantification of novel splice events. Moreover, *DJExpress* offers a web-compatible graphical interface allowing the analysis of user-provided data as well as the visualization of splice events within our custom database of differential junction expression in cancer (DJEC DB). DJEC DB includes not only healthy and tumor tissue junction expression data from TCGA and GTEx repositories but also cancer cell line data from the DepMap project. The integration of DepMap functional genomics data sets allows association of junction expression with molecular features such as gene dependencies and drug response profiles. This facilitates identification of cancer cell models for specific splicing alterations that can then be used for functional characterization in the lab. Thus, *DJExpress* represents a powerful and user-friendly tool for exploration of alternative splicing alterations in RNA-seq data, including multi-level data integration of alternative splicing signatures in healthy tissue, tumors and cancer cell lines.

Keywords: alternative splicing, splicing aberrations, differential splicing analysis, cancer splicing, The Cancer Genome Atlas Program (TCGA), GTEx database

INTRODUCTION

Splicing of pre-mRNA is a crucial process in eukaryotic gene expression regulation. In addition to canonical splicing, which leads to the inclusion of constitutive exons into the mature mRNA, the transcriptome is subjected to alternative splicing. Alternative splicing can give rise to multiple protein-coding isoforms from a single pre-mRNA and thus represents a major determinant for

TABLE 1 | Feature comparison between *DJExpress* and other existing splicing analysis tools.

Tool	GUI	User-selected alignment method	Non-annotated junctions supported	Splicing pattern visualization	Downstream trait association
DJExpress	Yes	Yes	Yes	Yes	Yes
MAJIQ	Yes	Yes	Yes	Yes	No
Psichomics	Yes	Yes	No	Yes	Yes
AltAnalyze	Yes	Yes	No	Yes	Yes
LeafCutter	Yes	No	Yes	Yes	Yes
SplAdder	No	Yes	Yes	Yes	No
rMATS	No	Yes	Yes	No	No
SpliceSeq	Yes	No	No	Yes	No
Whippet	No	No	Yes	Yes	No
JunctionSeq	No	No	Yes	Yes	No
MISO	No	No	No	Yes	No
SUPPA	No	Yes	No	No	No
Cufflinks	No	No	Yes	No	No
Salmon	No	Yes	No	No	No
RSEM	No	Yes	No	No	No
Sailfish	No	No	No	No	No
VAST-TOOLS	No	No	No	No	No
Kallisto	No	No	No	No	No

proteome diversity. Approximately 92%–94% of human genes generate alternatively spliced transcripts, often with tissue-specific regulation (Wang et al., 2008; Barbosa-Morais et al., 2012). Alternative splicing is involved in a variety of cellular processes, such as cell proliferation, differentiation, migration and survival (Paronetto et al., 2016; Gallego-Paez et al., 2017). Emerging data indicate that alternative splicing plays a critical role in the pathogenesis of many diseases, including several molecular subtypes of cancer (Oltean and Bates, 2014; Scotti and Swanson, 2016; Jiang and Chen, 2021). Interrogating such splicing abnormalities can facilitate identification of disease drivers, drug resistance mechanisms, and molecules capable of regulating pathological splicing events. Thus, exploration of alternative and aberrant splicing phenotypes promises to shed light on novel aspects of health and disease.

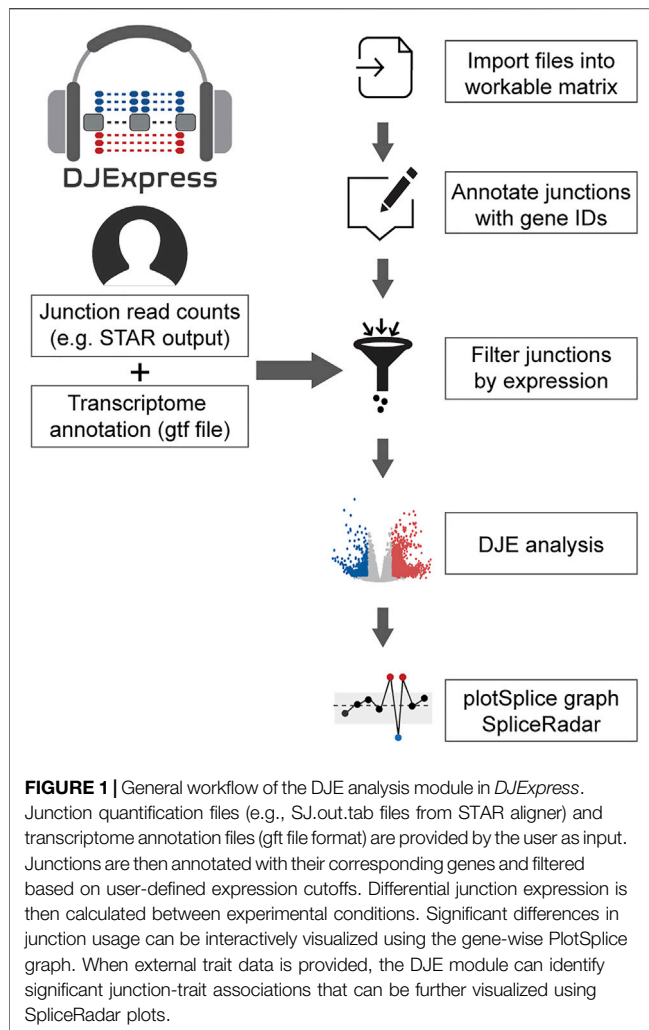
The recent release of transcriptome-wide RNA sequencing (RNA-seq) data repositories such as The Cancer Genome Atlas (TCGA) (Tomczak et al., 2015) and the Genotype-Tissue Expression (GTEx) project (Lonsdale et al., 2013) have lifted alternative splicing analysis opportunities to an unprecedented level. However, a unified and accessible analysis strategy for this data has largely been missing.

The gradual development of RNA-seq technologies and cost-effective alternative splicing studies at the transcriptome level has allowed the parallel evolution of bioinformatic tools for splicing quantification and visualization. Most of these tools rely on two main computational approaches: 1) quantification of the Percent Spliced-In (PSI) metric, which uses the ratio between exon-exon junction spanning sequencing reads that provide evidence for the inclusion or exclusion of an alternatively spliced region [e.g., rMATS (Shen et al., 2014), MISO (Katz et al., 2010), SUPPA (Alamancos et al., 2015), SplAdder (Kahles et al., 2016), psichomics (Saravia-Agostinho and Barbosa-Morais, 2019), AltAnalyze (Emig et al., 2010), SpliceSeq (Ryan et al., 2012), VAST-TOOLS (Irimia et al., 2014), MAJIQ (Vaquero-Garcia et al., 2016), LeafCutter (Li et al., 2018) and Whippet (Sterne-Weiler et al., 2018)], and 2)

quantification and de-convolution of the entire set of reads aligned to the gene to estimate transcript isoform abundance (e.g., Cufflinks (Trapnell et al., 2010), RSEM (Li and Dewey, 2011), Sailfish (Patro et al., 2014), Salmon (Patro et al., 2017) and Kallisto (Bray et al., 2016)) (see **Table 1** for a comparison of these tools). Although these bioinformatic tools have propelled transcriptome-wide alternative splicing analysis forward, they suffer from significant limitations. These include the need for high computational resources and bash-based operation, restrictions of input file formats, incomplete transcriptome annotation and consequently inaccurate transcript/PSI quantification. Furthermore, these tools suffer from complex static graphical outputs that are complicated to visualize and interpret or lack the option for association of splicing phenotypes to clinical or molecular data. These caveats are obstacles for a straight-forward interpretation of the biological and physiological relevance of alternative splicing in disease. Thus, despite the large variety of available tools, there is still a high demand for easy-to-use alternative splicing analysis strategies that can incorporate comprehensive data visualization and integration with external sample traits.

Here we introduce a novel differential junction expression analysis pipeline, *DJExpress*, which is an R package for analysis of transcriptomic features and expression-trait associations. *DJExpress* runs on an average laptop computer (**Supplementary Figure S1**) and provides a set of interactive and intuitive visualization formats. *DJExpress* uses raw splice junction counts—derived from STAR aligner (Dobin et al., 2013) or other junction quantification algorithms—as input data to determine gene-level differential junction usage. The statistical approaches implemented by *DJExpress* include empirical Bayesian procedures to assess differential junction expression between experimental conditions and junction-level t-statistics tests to determine differences between each junction and all other junctions within the same gene.

In contrast to the majority of existing pipelines, *DJExpress* can handle both annotated and *de novo* identified splice junctions, thereby allowing the characterization of novel splice events.



Moreover, through gene-level differential junction usage calculation, *DJExpress* identifies associations between junction expression and molecular/clinical features using large matrix operations. An additional more advanced feature of *DJExpress* involves weighted junction co-expression network analysis (JCNA). JCNA-derived junction expression modules can be correlated with phenotypes of interest, thereby allowing differential splicing analysis on a systemic scale. For downstream processing, JCNA outputs can be exported in a format compatible with network visualization tools such as VisANT and Cytoscape (Shannon et al., 2003; Hu et al., 2004).

In addition to these locally accessible features, *DJExpress* offers a web-compatible graphical interface for the analysis of user-provided data as well as the visualization of DJEC DB, a custom database of cancer-specific splicing profiles and their association to external traits from tumor samples and cancer cell lines. DJEC DB includes not only TCGA and GTEx data, but also cancer cell line data from the Cancer Dependency Map (DepMap¹) project. The integration of DepMap data allows association of junction expression with functional

genomics features such as gene dependencies and drug response profiles. This facilitates identification of cancer cell models for specific splicing alterations that can then be used for functional characterization in the lab.

Taken together, *DJExpress* represents a novel and versatile tool to analyze and explore alternative splicing phenotypes in health and disease.

METHODS

Differential Junction Expression Module

The data analysis workflow in the DJE module is depicted in **Figure 1**. For differential junction expression (DJE) and junction co-expression network analysis (JCNA), *DJExpress* uses quantified raw reads aligned to exon-exon junction loci and the transcriptome annotation as the primary input. Mapped and quantified junction reads are typically generated from FASTQ or BAM files using common RNA-seq alignment/quantification tools [e.g., STAR (Dobin et al., 2013), TopHat (Trapnell et al., 2009), MapSplice (Wang et al., 2010), Rsubread (Liao et al., 2019)] (**Figure 2A**). Following the statistical principles in limma Bioconductor package (Law et al., 2014; Ritchie et al., 2015), *DJExpress* first tests for differential expression of genomic features (here splice junction regions) using an initial input matrix of read count values as rows and sample ids as columns. Count data is then transformed to log₂-counts per million (logCPM), and observation-level weights based on mean-variance relationship are computed (using the *voom* function from *limma*). Users can decide at this point whether to keep the default expression threshold for filtering junctions prior to hypothesis testing (10 minimum of read count mean per junction) or to adjust the threshold based on the mean-variance trend. A linear model is then fit per junction using a provided experimental design, and empirical Bayes moderated *t*-statistics are implemented to assess the significance level of the observed expression changes.

The linear model framework of *limma* is also used in parallel to calculate differential junction usage, where significant differences in log-fold changes in the fit model between junctions from the same gene are tested (using the *diffSplice* function from *limma*). *DJExpress* thereby identifies alternatively spliced regions in transcripts based on two main features of splice junction expression: 1) Quantitative changes in the abundance of individual junctions between experimental groups, and 2) Differences in their expression levels compared to the average expression of other junctions in the gene.

Following these criteria, splice junctions are classified based on their absolute log-fold change (e.g., experimental condition A vs B) and their relative log-fold change (target junction vs all other junctions in the gene) in one of the following expression groups (**Figure 2B**):

Group 0: Junctions without differential expression or differential usage.

Group 1: Junctions with equal levels of differential expression and differential usage, reflecting changes in splicing patterns between experimental conditions (in this case, both absolute and relative log-fold change values are similar, if not the same).

¹<https://depmap.org/>.

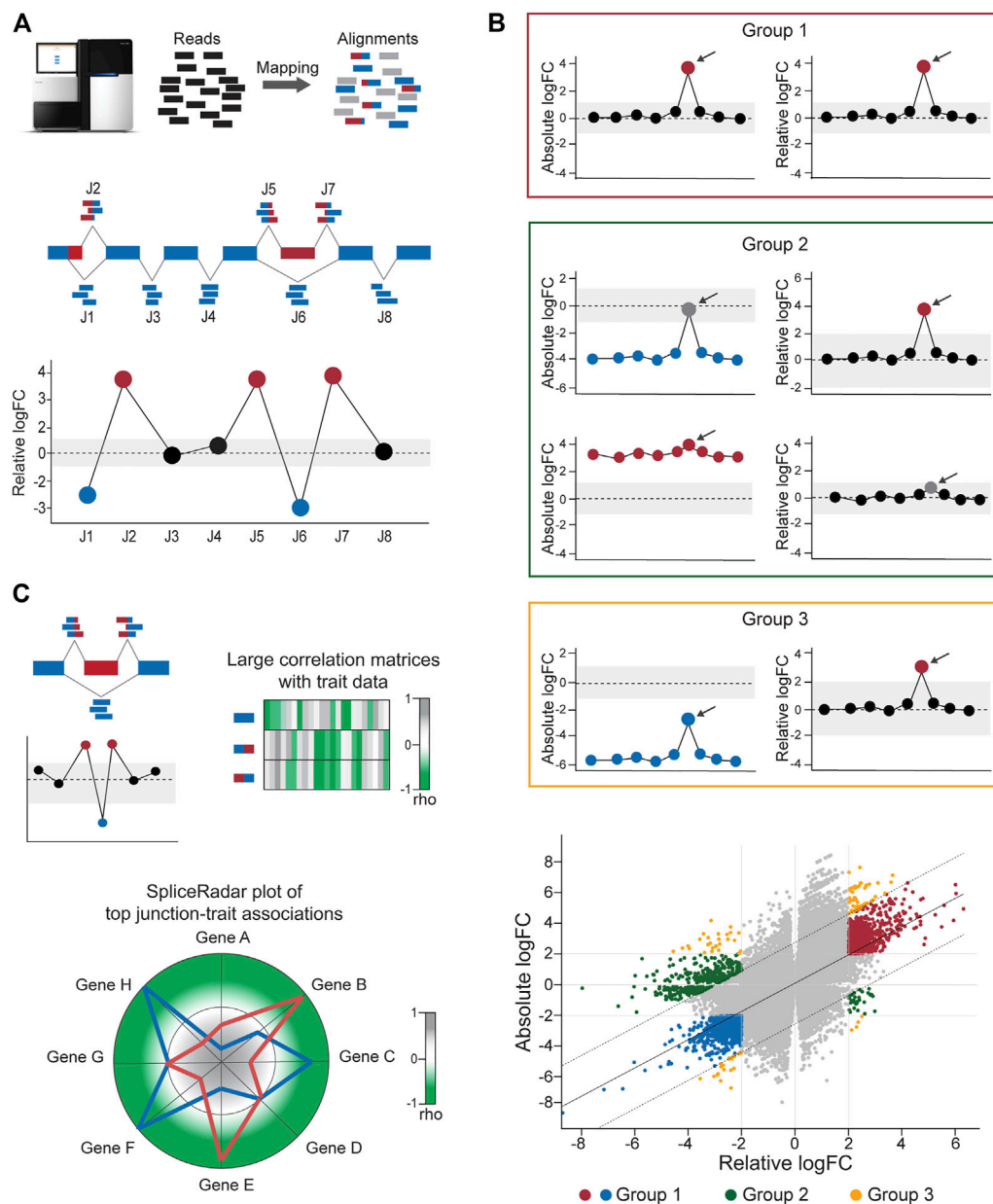


FIGURE 2 | Calculation of differential junction expression using the DJE module. **(A)** After alignment and quantification of RNA-seq reads supporting exon–exon junctions, differential junction expression is analyzed and depicted using the gene-wise splice plot visualization method. The schematic shows 8 junctions (J1–J8) in hypothetical gene, where each junction is plotted along the x-axis and ordered by genomic coordinate position. Relative log-fold change values (logFC), which indicate the difference between the expression of the target junction vs the average junction expression in the gene is shown in the y-axis. Junctions with logFC values above a user-defined threshold (absolute logFC of 1.0 in the example) are considered as differentially used and colored blue or red in case of downregulation and upregulation, respectively. **(B)** *DJExpress* determines alternatively spliced transcript regions based on both, alterations in their expression levels compared to the average expression of other junctions the same gene (differential usage, based on relative logFC) and alterations in junction abundance between experimental conditions (differential expression, based on absolute logFC). Junctions are then classified into four main groups. Group 0 corresponds to junctions without differential expression or differential usage and is visually represented as grey points in the scatter plot. Group 1 (red box and red/blue points in the scatter plot) comprises junctions with similar values of absolute and relative logFCs which reflects changes in splicing patterns between experimental conditions without confounding alterations in the total expression of the gene. Group 2 (green box and green points in the scatter plot) represents junctions with differential expression but no differential usage or vice-versa, which indicates the presence of altered total gene expression levels between conditions that explain observed differences. Group 3 (orange box and orange points in the scatter plot) designates junctions with significant but dissimilar levels of relative and absolute logFCs, indicating the presence of both, total gene expression and local splicing changes. Relative vs absolute logFC plots are produced within the output of the DJE module, where junctions are classified into specific groups according to the significance of their logFC values and their position inside or outside of the distribution by ≥ 2 standard deviations. Arrows indicate example target junctions. **(C)** When external sample trait data (e.g., clinical or molecular data) are provided by the user, *DJExpress* can identify significant junction-trait associations within a target experimental condition using either correlation analysis, ANOVA test or linear regression models. If correlation is selected by the user (as in the depicted example), the

(Continued)

FIGURE 2 | results are used to construct heatmap or SpliceRadar plots with target splice junctions (e.g., inclusion junctions (red) and exclusion junction (blue) in an exon skipping event). In the case of SpliceRadars, positive correlation coefficients are located within the outer region (green) and negative correlation coefficients are found within the inner region (grey) of the radar chart, allowing the visual inspection of multivariate trait associations to user-selected alternative splicing events.

Group 2: Junctions with differential expression but no differential usage or vice versa, implying the occurrence of generalized changes in expression across the gene, rather than the presence of a differentially spliced region (in this case, either the absolute or relative log-fold change value is not significant).

Group 3: Junctions with divergent levels of differential expression and differential usage, indicating concomitant changes in splicing and total gene expression (in this case, the absolute and relative log-fold change values can substantially vary from each other).

One of the main features of DJE module's approach is the incorporation of an interactive gene-wise junction representation (**Figure 2A**). This approach facilitates straight-forward visual inspection of differential splicing across the gene and exploration of supplementary information about each junction's expression. This includes the above-mentioned classification based on absolute and relative log-fold change patterns, basic statistics on expression levels (e.g., mean and median expression in each experimental condition, number of samples expressing the junction, etc.) as well as the identification of non-annotated and condition-specific junctions. The latter are also called “neojunctions” in the *DJExpress* pipeline, referring to junctions detected in the tested condition but are not found in the control condition.

Junction-Trait Association Module

Further exploration of the potential physiological relevance of alternative splicing is possible through the association of junction expression to external sample traits (e.g., clinical or molecular data). Significant junction-trait linkages are determined by large matrix operations including correlation analysis, ANOVA test or linear regression models [using *cor* and *bicor* from WGCNA (Langfelder and Horvath, 2008) and *Matrix_eQTL_engine* from *MatrixEQTL* (Shabalín, 2012)]. The top significant association can be visualized through heatmap plots or alternatively, using the SpliceRadar plot format (**Figure 2C**), where the coefficient of top-ranked correlations is used to map each junction-trait association within a radar chart. This graphical concept allows the users to simultaneously visualize relevant associations between the expression of selected junctions (e.g., the top most differentially expressed junctions or a subset of junctions within a target gene) and external traits, as well as to elucidate expression-trait patterns shared among junctions of interest with potential biological relevance.

Junction Co-Expression Network Analysis Module

A widely used approach for describing correlation networks in systems biology is the weighted gene co-expression network analysis (WGCNA, Langfelder and Horvath, 2008). WGCNA

is a screening method based on pairwise correlations between features in gene expression data. This approach allows the identification of clusters (or modules) of highly correlated genes, intramodular hub genes and representative module eigengenes (MEs). These can be used in the estimation of module membership values for each gene as well as in association analyses between modules and to external sample traits. This technique has been frequently implemented for the assessment of gene-network signatures and for the identification of functional pathways and candidate molecular biomarkers, integrating gene expression and clinical/molecular data from physiological and disease conditions (Oldham et al., 2008; Presson et al., 2008; Ma et al., 2017; Vieira et al., 2019).

The weighted junction co-expression network analysis module (JCNA) in *DJExpress* provides an implementation of WGCNA algorithms (version 1.70.3, Langfelder and Horvath, 2008) in the context of splice junction expression when sufficient sample size is provided (≥ 15 samples within single experimental conditions as suggested in the WGCNA guidelines) (**Figure 3A**). JCNA initiates with a data pre-processing step where outlier samples (clustered using the average linkage method) and lowly expressed junctions are removed to ensure high confidence network construction. Correlation matrices (e.g., using Pearson, Spearman or the default biweight midcorrelation) (Wilcox, 2012) are then built for all pair-wise junctions. The full network is subsequently specified by a weighted adjacency matrix calculated with an appropriate soft threshold power (Zhang and Horvath, 2005). Summary plots of a network topology analysis are produced by JCNA (following WGCNA guidelines) to aid users in the selection of the soft-thresholding power around which scale-free topology in the junction network is achieved.

Additional parameters such as minimum module size, module detection sensitivity or cut height of the hierarchical clustering dendrogram for module definition can be introduced for junction module identification (**Figure 3B**). Calculation of MEs is also possible, where expression patterns of all junctions in a module are summarized into a single expression profile. This measure is then used in the correlation analysis with sample traits. Notably, ME calculation reduces the computational burden of multiple testing, which otherwise can be exceedingly high since junction quantification datasets usually comprise millions of expression features.

Users can either keep the output of a 1-pass JCNA or can continue into a second round of network construction. During this 2-pass JCNA, the gene expression-specific effect within junction modules is subtracted. This is particularly relevant in the context of junction-trait associations, since a considerable number of co-expressing junctions are expected to cluster into single modules as a result of intrinsic associations at the gene

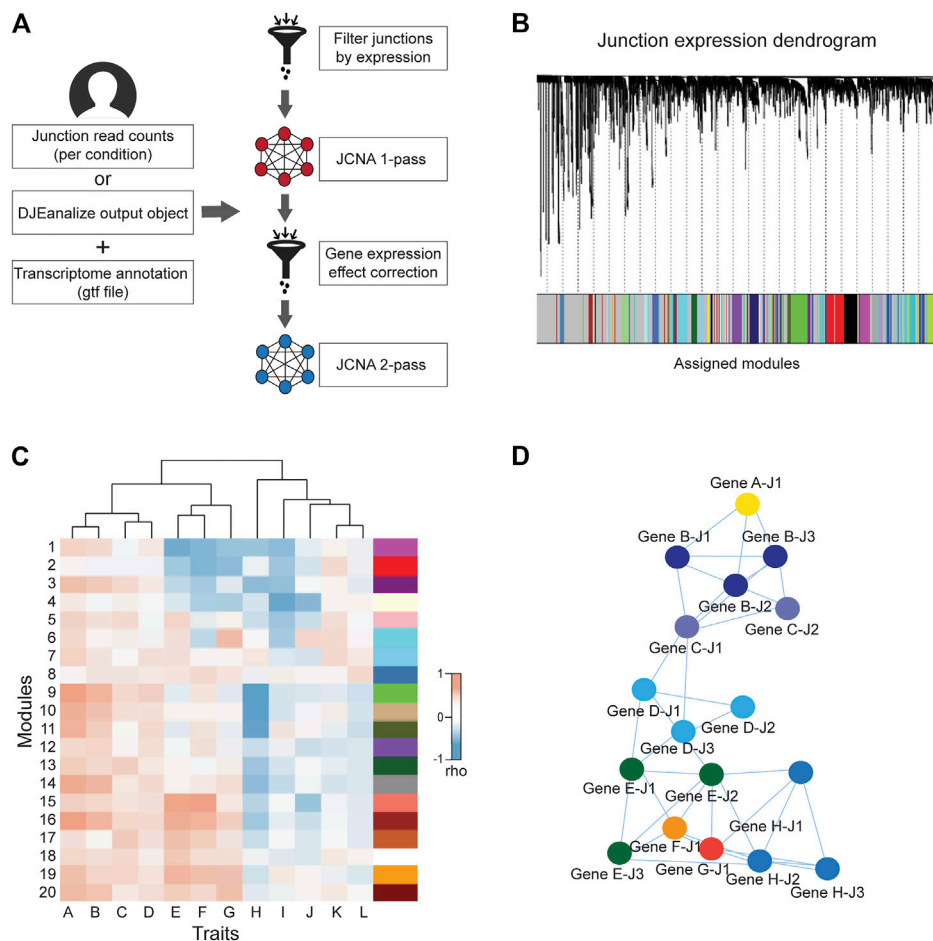


FIGURE 3 | General workflow of JCNA module in *DJExpress*. **(A)** For the *DJExpress* JCNA module, the user needs to provide junction read counts (or the output of the *DJEanalyze* function) and a transcriptome annotation file. After removing outlier samples and lowly expressed junctions, a first round of co-expression analysis is performed where junction modules and module/junction vs trait associations are calculated. The user can continue into a second round of network construction, where co-expression analysis and trait association is produced using gene expression data. This information is used to identify and remove junction-trait correlations from the network that reflect gene expression-based associations. The remaining junction set is used to re-construct junction co-expression modules and module-trait correlations. **(B)** Dendrogram schematic of clustered junctions with assigned modules based on a dissimilarity measure (1-TOM) as described for WGCNA (Langfelder and Horvath, 2008). **(C)** Heatmap schematic of correlations between junction module eigengenes (MEs) and different sample traits. **(D)** Schematic representation of interaction networks of junctions within a co-expression module that can be produced using Cytoscape or VisANT visualization tools. Junctions belonging to the same gene are indicated by the same color.

expression level. Here, 2-pass JCNA improves the identification of true co-splicing signatures, since junctions from the same gene or from highly correlated genes tend to cluster without any specific association to splicing.

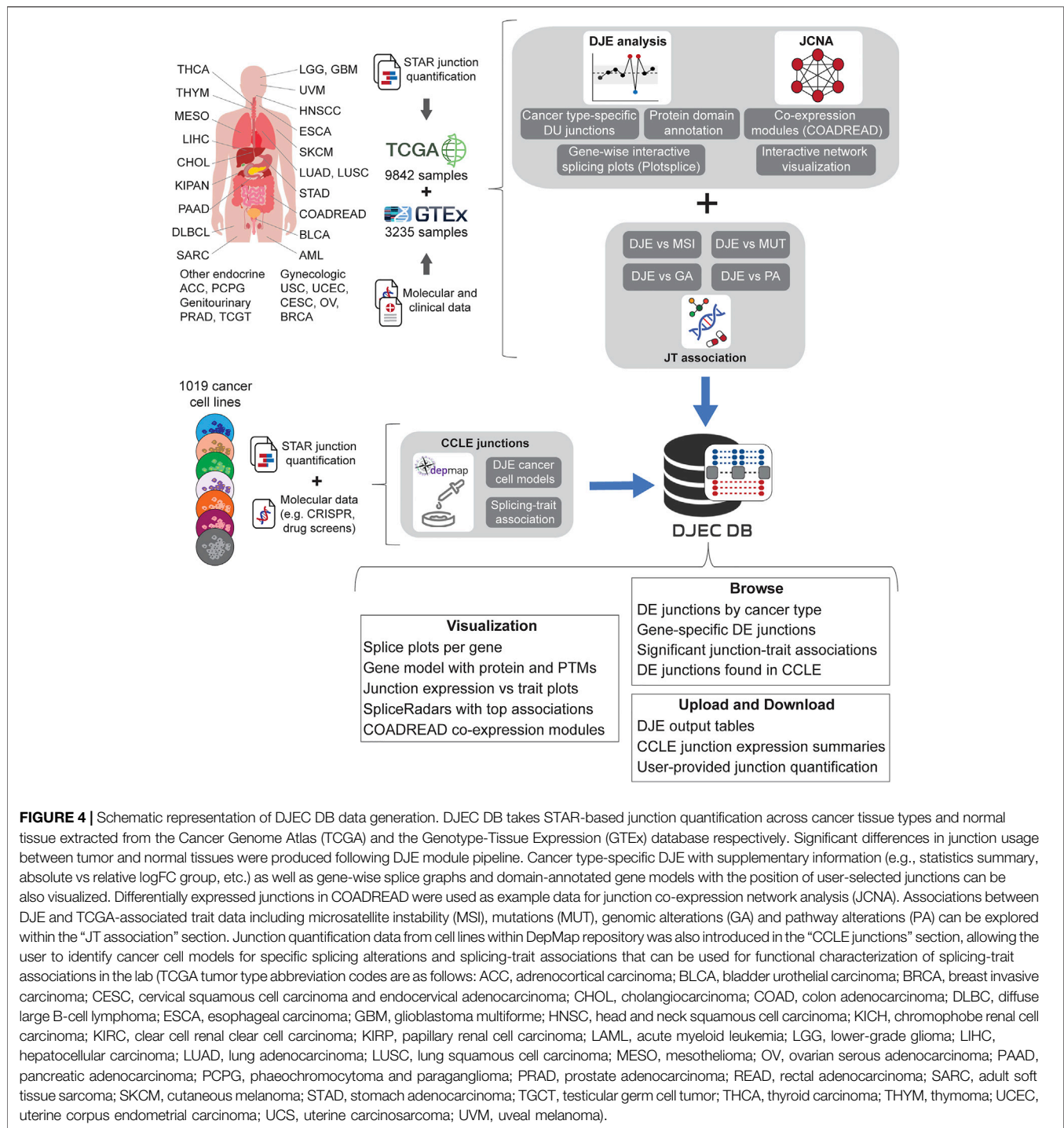
For 2-pass JCNA, gene expression-based networks including correlations with a user-selected sample trait are calculated (Figure 3C). The absolute value of junction significance, which represents the correlation coefficient between a given junction and the selected trait is plotted as a function of the corresponding gene significance. Junctions outside of the distribution by ≥ 2 standard deviations (showing no correlation between junction and gene significance for trait) are kept for network re-construction. Thus, 2-pass JCNA strategy allows the user to further explore associations between molecular/clinical traits and modules of

co-expressed splicing events that can be defined once gene expression-related junction co-expression is identified and removed from the network.

Furthermore, as in the case of WGCNA pipeline, the resulting junction modules from JCNA can be also exported to network graphical tools such as Cytoscape or VisANT for further visual exploration and customization (Figure 3D).

Run Time and Memory Benchmarks

For run time and memory consumption benchmarks of function within the DJE module (*DJEimport*, *DJEannotate*, *DJEprepare* and *DJEanalyze*), we used STAR-derived junction quantification files from the TCGA COADREAD tumor sample cohort. *DJExpress* pipeline was applied 10 times on



two cores of a macOS X 11.6.1 system with 2.3 GHz Quad-Core Intel Core i5 processor and 16 GB of memory, RStudio Desktop 1.4.1106 and R 4.0.5. Each run was performed on datasets with increasing number of samples (e.g., 10, 20, 40, 60, 80, 100, 200, 400, 600, 800, 1000) and 100,000 randomly retrieved splice junctions. For the differential junction expression analysis using *DJEanalyze*, samples were randomly divided into two groups using Bernoulli

distributed values with a 50% probability of success (**Supplementary Figure S1**).

Data Collection for Differential Junction Expression in Cancer Database

Using the pipelines described for the DJE and JCNA modules, we generated DJEC DB, a custom database of cancer-specific splicing

profiles and their association to external traits from tumor samples and cancer cell lines (**Figure 4**). DJEC DB can be accessed through a graphical interface based on the *shiny* package (version 1.6.0) and includes healthy and tumor tissue data for 9,842 human samples across 32 different tumor types from TCGA, 3,235 normal post-mortem tissue samples from GTEx and 1,019 cancer cell lines from the DepMap Project.

Alignment of GTEx and TCGA RNA-seq data sets to the GRCh37 reference genome and subsequent splice junction quantification, as well as removal of low-quality tissue samples was previously done (Kahles et al., 2018) using the STAR aligner tool with the following arguments:

```
STAR --genomeDir GENOME --readFilesIn READ1 READ2
--runThreadN 4 --outFilterMultimapScoreRange 1 --outFilter
MultimapNmax 20 --outFilterMismatchNmax 10 --alignIntron
Max 500000 --alignMatesGapMax 1000000 --sjdbScore 2 --align
SJDBoverhangMin 1 --genomeLoad NoSharedMemory --limit
BAMsortRAM 70000000000 --readFilesCommand cat --outFilter
MatchNminOverLread 0.33 --outFilterScoreMinOverLread 0.33
--sjdbOverhang 100 --outSAMstrandField intronMotif --out
SAMattributes NH HI NM MD AS XS --sjdbGTFfile GEN
CODE_ANNOTATION --limitSjdbInsertNsj 2000000 --out
SAMunmapped None --outSAMtype BAM SortedBy
Coordinate --outSAMheaderHD @HD VN:1.4 --outSAMattrRG
line ID:<ID> --twopassMode Basic --outSAMmultNmax 1
```

We used the raw junction counts from this study as the basis for DJEC DB. For this, differential junction expression analysis was implemented comparing junction abundance between each TCGA cancer type and all GTEx normal tissues. Cancer-specific changes in junction expression can be accessed through the DJE Module section in the DJEC DB web application (**Supplementary Figure S2**). Here, users can select target junctions to visually explore interactive splice plots and differentially expressed junctions in the context of protein domain and post-translational modifications annotated within the Prot2HG database of protein domains mapped to the human genome (Stanek et al., 2020).

In addition to RNA-seq data, the TCGA repository contains an extensive molecular and clinical annotation for tumor samples, including additional omics data (genotyping, DNA methylation, etc.) as well as multiple tumor classifications and clinical records of the patient. This data collection allows comprehensive correlation analyses between junction expression and tumor/patient traits. The junction-trait (JT) module section of DJEC DB (**Supplementary Figure S3**) contains significant linkages found between differentially expressed junctions and microsatellite instability (MSI) or altered oncogenic signaling pathways based on mutations, copy-number changes (CNV), mRNA expression, gene fusions and DNA methylation (Sanchez-Vega et al., 2018). This approach is an adaptation of the Matrix eQTL method (Shabalina, 2012), which uses large matrix operations of linear and ANOVA models containing covariates to account for external factors such as tumor grade or age of the patient.

Moreover, an exemplary co-expression network analysis can be also found within the JCNA section, where users can interactively explore junction expression modules as well as

the results of junction-traits associations in TCGA colorectal (COADREAD) tumors (**Supplementary Figure S4**). This implementation of WGCNA algorithms included the removal of junctions with excessive missing values and sample outliers after sample hierarchical clustering using the *goodSamplesGenes* function (Langfelder and Horvath, 2008). The subsequent soft-thresholding procedure ensures a scale-free network, which emphasizes strong correlations between junctions and penalizes weak correlations. The scale-free network was constructed using the *blockwiseModules* function which converts the correlation matrix into a strengthened adjacency matrix that summarizes the association between all junctions.

Gene-trait correlation matrices were also calculated and used to identify and remove junctions whose correlation to external traits was gene expression-dependent. Junction co-expression modules were identified by dividing the junction expression dendrogram into branches using a dynamic tree cutting algorithm with medium sensitivity for cluster splitting (*deepSplit* = 2). Different colors were then assigned to the modules for subsequent visualization. MEs significance values and correlations between MEs and clinical traits were also calculated. The same was done for individual junction-to-trait correlations.

To implement cancer cell line junction expression data into DJEC DB, we downloaded fastq files from CCLE (available through the Sequence Read Archive (SRA) under accession number PRJNA523380) and carried out alignment and junction quantification with the same strategy that was previously used for TCGA and GTEx data (Kahles et al., 2018). This data was then integrated with DepMap functional genomics data in the CCLE DJE and CCLE SpliceRadar sections of DJEC DB (**Supplementary Figure S5**). CCLE DJE comprises the results of DJE analysis in cancer cell lines within the same tissue of origin versus fibroblasts used as “healthy” control cell lines. Significant correlations between differentially expressed junctions and gene expression, CRISPR gene effect or drug response values (DepMap 21Q3 Public, 2021) are found within CCLE SpliceRadar. Here, users can plot SpliceRadar charts with selected junction-trait associations. These database components aim to facilitate the identification of cancer cell models for specific splicing alterations and junction-trait associations that can be further studied for functional characterization in the lab.

RESULTS

The *DJExpress* toolbox incorporates both an R package (containing DJE and JCNA modules) and a user-friendly Shiny-based web application for a visual exploration of DJEC DB as well as custom DJE analysis for user-provided junction quantification data. Input files can either be STAR aligner-derived “SJ.out.tab” files (containing splice junction counts per sample in tab-delimited format) or any other junction quantification files as long as they contain junction IDs as first columns, following the format chr:start:end:strand (e.g., chr1:123:456:1, where positive

TABLE 2 | Summary of DJE module junction statistics in CCLE.

CCLE tissue	Quantified junctions	DE junctions	DE junctions in Group 1	DE junctions in Group 2	DE junctions in Group 3	Novel junctions	Neojunctions
Brain	120,611	846	74	73	14	3,456	110
Breast	123,349	2,153	499	431	247	3,426	255
Colon	122,639	3,363	663	722	409	3,400	336
Gastric	126,487	2,335	540	486	293	3,806	320
Head-Neck	119,194	2,398	440	391	144	3,573	316
Kidney	117,989	1,231	185	143	119	3,574	164
Leukemia	123,295	3,668	631	1,060	511	3,563	514
Lung	130,297	2,327	386	549	154	3,403	368
Lymphoma	122,911	3,795	689	1,012	524	3,772	354
Myeloma	119,528	3,307	727	678	420	3,734	398
Ovarian	122,251	1,603	295	283	238	3,512	241
Pancreatic	121,817	2,528	448	418	308	3,614	220
Skin	120,200	2,036	186	357	247	3,498	197

or negative strand are coded as 1 and 2, respectively). In the following paragraphs, we describe the use of *DJExpress* and DJEC DB in detail and use case studies to demonstrate how *DJExpress* and DJEC DB can be utilized to identify and computationally explore alternative splice events across cell lines and patient samples.

Differential Junction Expression and Junction-Trait Association Analyses in Cancer Cell Lines

To demonstrate the workflow of *DJExpress*, we analyzed cancer cell lines from the DepMap repository, comprising 13 tissue types that contain ≥ 30 individual cell lines per tissue (brain, breast, colon/colorectal, gastric, head and neck, kidney, leukemia, lung, lymphoma, myeloma, ovarian, pancreatic and skin cancer). **Table 2** summarizes the results of DJE analysis module per tissue, using junction expression in fibroblasts as normal control condition. Users can explore this data in the DJE-CCLE section of DJEC DB.

DJExpress identified on average of 1,918 differentially used junctions ($\text{FDR} < 0.05$ and $|\log\text{FC}| > 1$), including previously described alternative splicing events in cancer, such as the downregulation of *ACTN1* exon 19b (Gardina et al., 2006; Thorsen et al., 2008; Bielli et al., 2018), *VCL* exon 19 (Gardina et al., 2006; Thorsen et al., 2008), the upregulation of *NUMB* exon 12 (Misquitta-Ali et al., 2011; Bechara et al., 2013; Zhang et al., 2014; Zong et al., 2014), *MAP3K7* exon 12 (Munkley et al., 2019; Qiu et al., 2020; Oh et al., 2021), *CTNND1* exon 20 (Yanagisawa et al., 2008; Sebestyen et al., 2015; Wang et al., 2020), and *EXOC1* exon 11 (Ray et al., 2020; Zhang et al., 2020), as well as of exons contained within the variant domain in *CD44* (Shirure et al., 2015; Chen et al., 2018; Wang et al., 2018; Chen et al., 2020) (**Figure 5; Supplementary Figure S6**). Moreover, the gene-wise visualization of differential junction expression allowed the identification of complex alternative splicing patterns and isoform switches in cancer, such as the case of the co-regulated inclusion of exon 11 and exclusion of exon 40 in *MYO18A* in lymphoma and myeloma, the complex local event

involving exons 15–18 in *MARK3* in leukemia, lymphoma, myeloma, breast, colon, gastric, lung and pancreatic cancer, or the isoform switches in *RGS3* in breast, colon, gastric, lung, ovarian and pancreatic cancers, and *INPP5B* in pancreatic cancer cell lines (**Figure 6; Supplementary Figures S7, S8**). These data demonstrate that *DJExpress* can not only reliably identify previously described alternative splicing events but can also facilitate the discovery and visualization of complex splice events within annotated splice regions.

Notably, an average of 3,563 non-annotated splice junctions per tissue and 292 neojunctions (defined as junctions not detected in control fibroblast cell lines) were also discovered by the DJE analysis module (**Table 2**). Here, the visualization of non-annotated junctions within the gene-wise DJE plots allowed us to identify the presence of previously unknown splicing events, including exon skipping, alternative 3' splice sites, alternative 5' splice sites and alternative first and last exons (**Supplementary Figure S9**). Moreover, DJE plots also revealed the presence of novel splice junctions with genomic coordinates that suggest the presence of exons so far not described in the human transcriptome annotation (**Figure 7; Supplementary Figure S10**). These newly identified splicing events are potentially linked to cancer physiology and their functional characterization could be subject of future studies. Nevertheless, to further illustrate the capabilities of *DJExpress* and DJEC DB, we next focused on a well-described alternative splicing switch in *NUMB* mRNA.

Case Study 1: SpliceRadar-Based Identification of *NUMB* Alternative Splicing Regulators

NUMB encodes for a key determinant of cell fate that regulates the trafficking of surface proteins such as Notch, integrins and E-cadherin and can undergo alternative splicing (Nishimura and Kaibuchi, 2007; McGill et al., 2009; Teckchandani et al., 2009; Wang et al., 2009). Inclusion of *NUMB* exon 12 is frequently observed in different types of cancer, leading to a 48 amino acid extension of the proline-rich region (PRR) of the NUMB protein

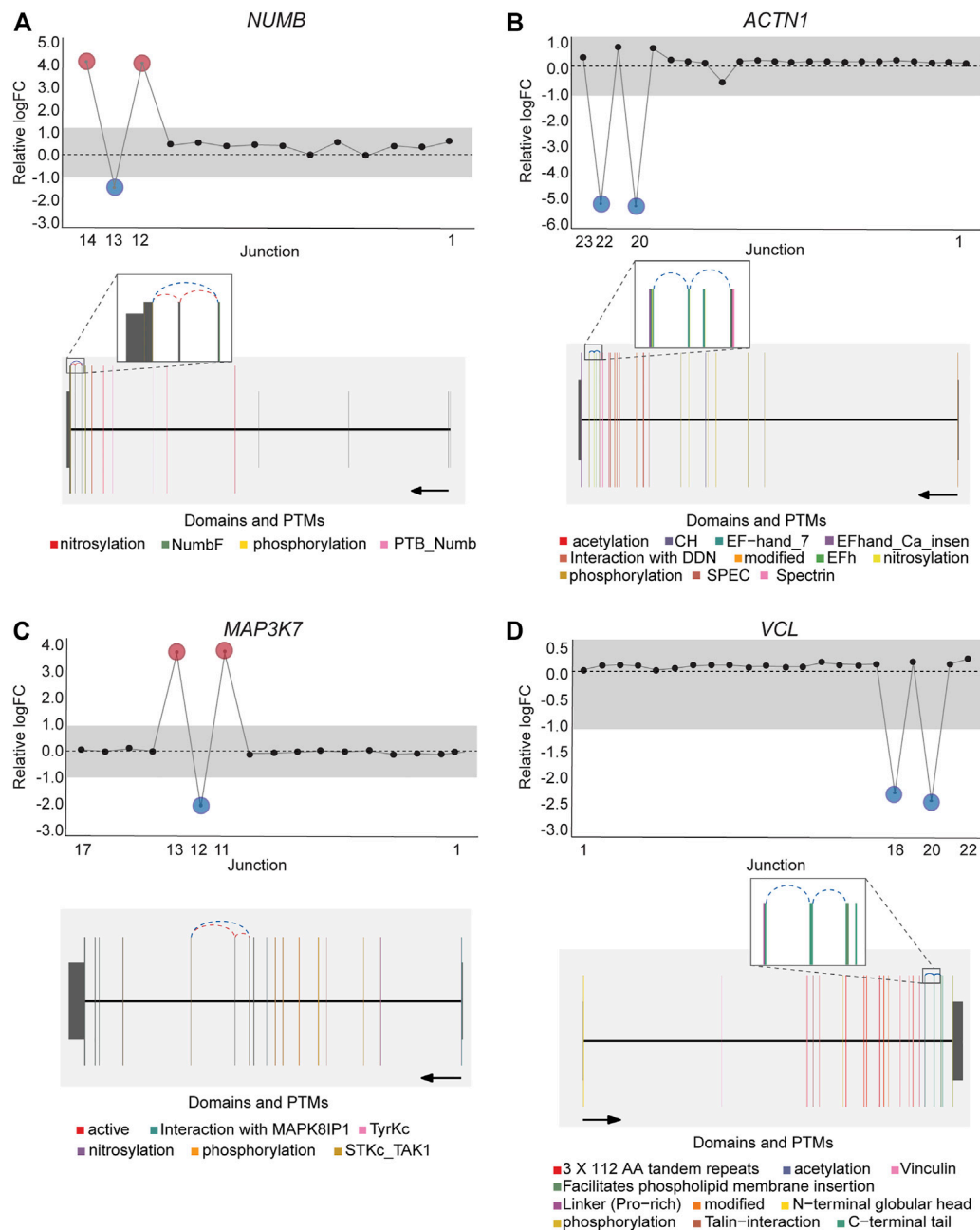


FIGURE 5 | Expression profile and gene context of known alternative splicing events in cancers detected by DJExpress using cancer cell line data. Examples of known cancer-specific splice events are shown as gene-wise splice plots with relative logFC values (upper panels) and gene model plots with exon-to-protein domain annotation (lower panels). **(A,B)** show gene-wise splice plots of exon inclusion events in *NUMB* and *ACTN1* mRNA in breast and lung cancer cell lines, respectively. **(C,D)** show gene-wise splice plots of exon skipping events in *MAP3K7* and *VCL* mRNA in gastric and breast cancer cell lines, respectively (Numbers on the x-axis in the upper panels indicate the first, last and differentially used junctions in the respective gene). Grey area indicate threshold for significance ($|\logFC| > 1.0$). Downregulated and upregulated junctions with $|\logFC|$ above threshold and significant FDR (< 0.05) are shown in blue and red, respectively. These same junctions are indicated within the gene model plots as dashed arcs connecting upstream and downstream exons. Colors within exonic regions indicate the presence of protein domains and/or post translational modifications (PTMs) annotated within the Prot2HG protein domain database. Arrows below gene model plots indicate direction of transcription. Coding and UTR exons are illustrated as long and short exons respectively. Junctions with both absolute and relative logFC above the threshold ($|\logFC| > 1.0$) but no significant FDR (> 0.05) for at least one of them are shown in black).

(Chen et al., 2009; Zhang et al., 2014; Lu et al., 2015; Rajendran et al., 2016). This longer NUMB isoform (Numb-L) was found to promote proliferation, whereas the shorter isoform (Numb-S)

promotes differentiation of cancer cells (Verdi et al., 1999). In lung cancer, the splicing factor *QKI* represses the inclusion of *NUMB* alternative exon through competing with a core splicing

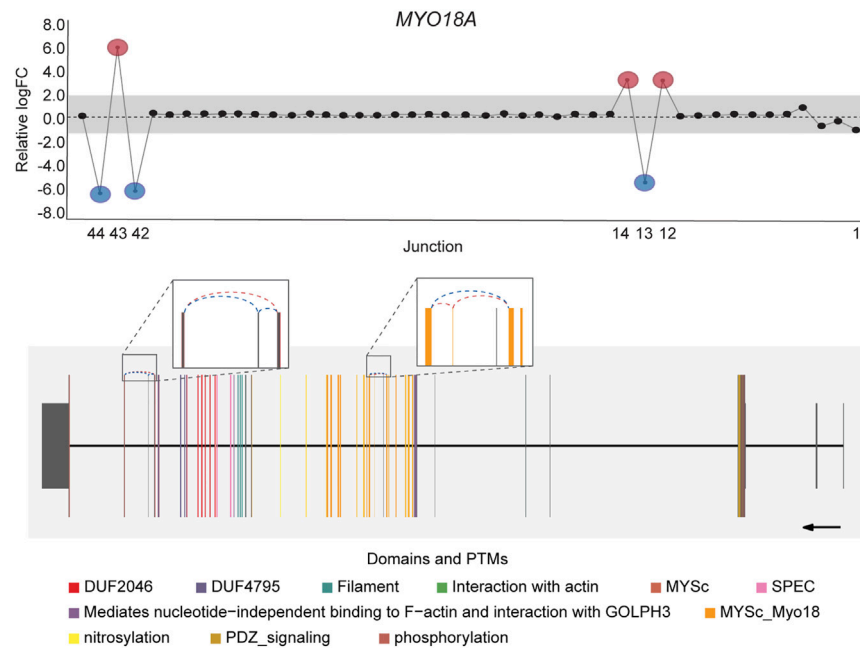


FIGURE 6 | Co-regulated splicing events within *MYO18A* transcript in blood cancer. Differentially used junctions as depicted in the gene-wise splice plot in *MYO18A* indicate the concomitant inclusion of exon 11 and exclusion of exon 40 in Myeloma and Lymphoma cell lines. Gene model plot with Prot2HG-based domain annotation suggest that these co-regulated splicing events involve exonic regions containing known *MYO18A* phosphorylation sites (brown), as well as regions comprising the core myosin-like ATPase motor domain, MYSc_Myo18 (orange). *MYO18A* gene-wise splice plot in lymphoma is used as example (Numbers on the x-axis in the upper panels indicate the first, last and differentially used junctions in the respective gene. Grey area indicate threshold for significance ($|\log FC| > 1.0$). Downregulated and upregulated junctions with $|\log FC|$ above threshold and significant FDR (< 0.05) are shown in blue and red, respectively. These same junctions are indicated within the gene model plots as dashed arcs connecting upstream and downstream exons. Colors within exonic regions indicate the presence of protein domains and/or post translational modifications (PTMs) annotated within the Prot2HG protein domain database. Arrows below gene model plots indicate direction of transcription. Coding and UTR exons are illustrated as long and short exons respectively. Junctions with both absolute and relative $\log FC$ above the threshold ($|\log FC| > 1.0$) but no significant FDR (> 0.05) for at least one of them are shown in black).

factor SF1, thereby inhibiting proliferation and Notch signaling (Zong et al., 2014).

This well-documented *NUMB* isoform switch was also detected with *DJExpress*, which showed a ~16-fold ($\log_2 \sim 4$ -fold) upregulation of *NUMB* exon 12 inclusion junctions in breast cancer cell lines compared to fibroblasts (Figure 5A). A similar *NUMB* splice pattern was observed across other cancer types (data not shown). Furthermore, by using *DJExpress* JT module, we corroborated the positive correlation between *QKI* gene expression and *NUMB* exon 12 exclusion (Figure 8A). Moreover, SpliceRadar-based visualization identified additional positively and negatively correlated splicing regulators, including *SRPK2* and *RBFOX2*, which have both previously been implicated in the regulation of *NUMB* alternative splicing (Lu et al., 2015). Thus, our data suggests that the control of *NUMB* alternative splicing in cancer may involve a more complex regulatory network than previously thought. These data demonstrate that *DJExpress* can not only validate known associations with splice events but can also, through functionality of the SpliceRadar tool, identify additional regulatory networks that may be altered in cancer.

DJEC DB incorporates gene dependencies and drug response data from the DepMap repository. We thus expanded the landscape of phenotypic associations to *NUMB* alternative splicing in lung cancer

cell lines (Figure 8B). Pathway enrichment analysis of significantly associated gene dependencies revealed enrichment of components within the mTOR and insulin signaling pathways. This is consistent with previous studies, which suggested that activated ERK signaling is a common mechanism that regulates *NUMB* isoform expression in breast and lung cancer cells (Rajendran et al., 2016) (Figure 8C). Similarly, SpliceRadar plots using top correlations with drug response values also revealed associations between the expression of exon-inclusion junctions in *NUMB* and cell survival rates after treatment with several compounds targeting PI3K/mTOR and ERK MAPK signaling (Supplementary Figure S11). These data reinforce the notion of a functional connection between *NUMB* exon 12 inclusion and pro-inflammatory signaling cascades.

Taken together, these results illustrate the potential of the *DJExpress* pipeline to identify *bona fide* differentially expressed splice junctions and reveal physiologically relevant associations between junction expression and various external traits. Thus, *DJExpress* can be used to support and generate hypotheses regarding the potential molecular mechanisms involved in the regulation and physiological consequences of alternative splicing.

DJEC DB Data Summary

TCGA project is a large-scale oncology study that has allowed the comprehensive characterization of multiple cancer types using a

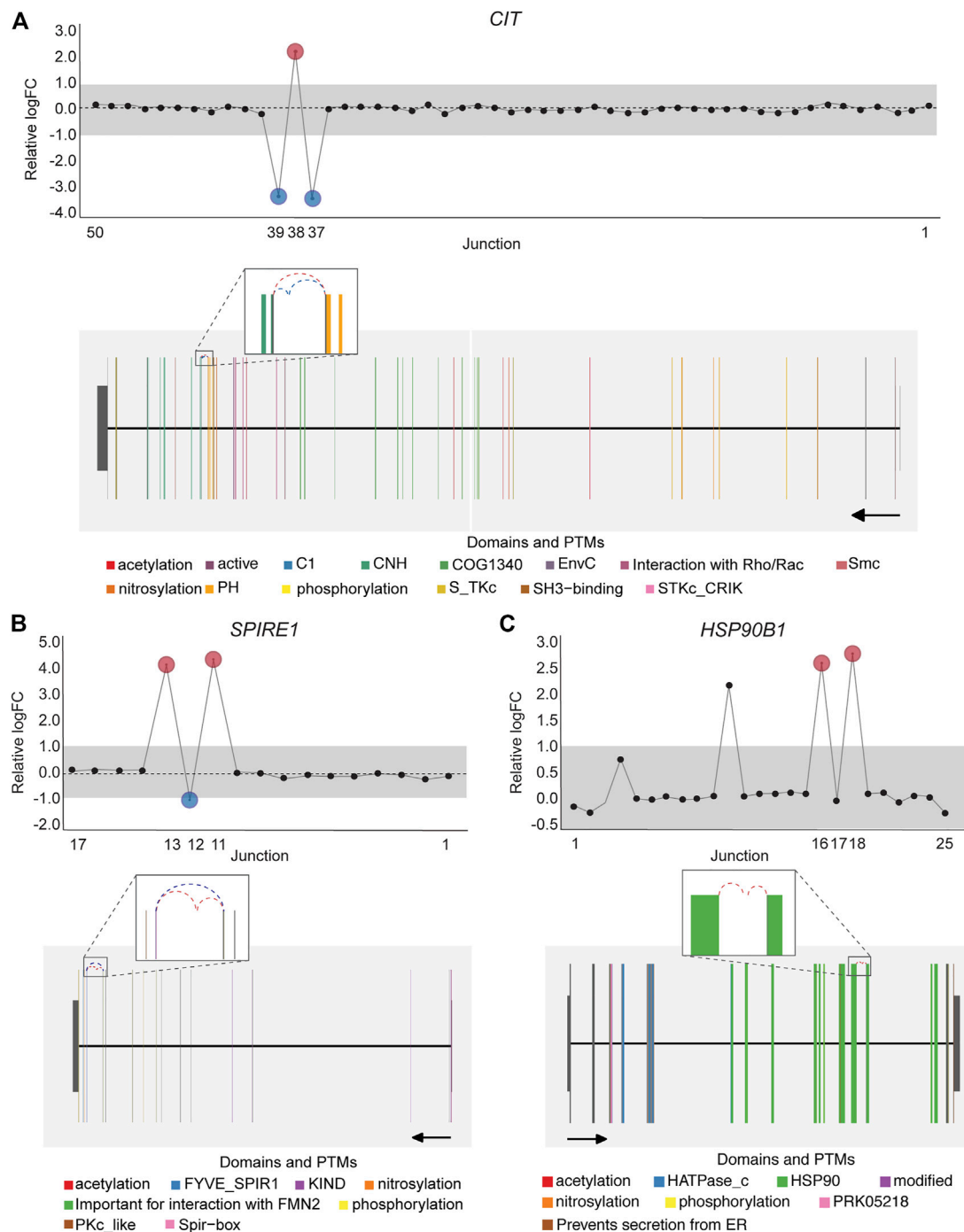


FIGURE 7 | DJE analysis suggests the presence of differentially spliced non-annotated exons in cancer cell lines. Gene-wise splicing as well as gene model plots show non-annotated splice junctions whose gene location indicates the presence of exons not described in the human transcriptome annotation. **(A)** Differentially expressed non-annotated junctions between exon 37 and 38 located in the vicinity of the CNH (dark green) and PH (orange) domains in *CIT*. **(B)** Differentially expressed non-annotated junctions between exon 12 and 13 in *SPIRE1*, which contain the Spir-box domain (pink) involved in the interaction between SPIRE1 and formin (FMN)-type actin nucleators, as well as protein phosphorylation sites (yellow). **(C)** Differentially expressed non-annotated junctions between exon 13 and 14 in *HSP90B1* occurring within the HSP90 chaperone domain (green). For *CIT* and *SPIRE1* gene-wise splice plots, breast cancer is used as example. For *HSP90B1*, lung cancer is used as example. (Numbers on the x-axis in the upper panels indicate the first, last and differentially used junctions in the respective gene. Grey area indicate threshold for significance ($|\logFC| > 1.0$). Downregulated and upregulated junctions with $|\logFC|$ above threshold and significant FDR (< 0.05) are shown in blue and red, respectively. These same junctions are indicated within the gene model plots as dashed arcs connecting upstream and downstream exons. Colors within exonic regions indicate the presence of protein domains and/or post translational modifications (PTMs) annotated within the Prot2HG protein domain database. Arrows below gene model plots indicate direction of transcription. Coding and UTR exons are illustrated as long and short exons respectively. Junctions with both absolute and relative \logFC above the threshold ($|\logFC| > 1.0$) but no significant FDR (> 0.05) for at least one of them are shown in black).

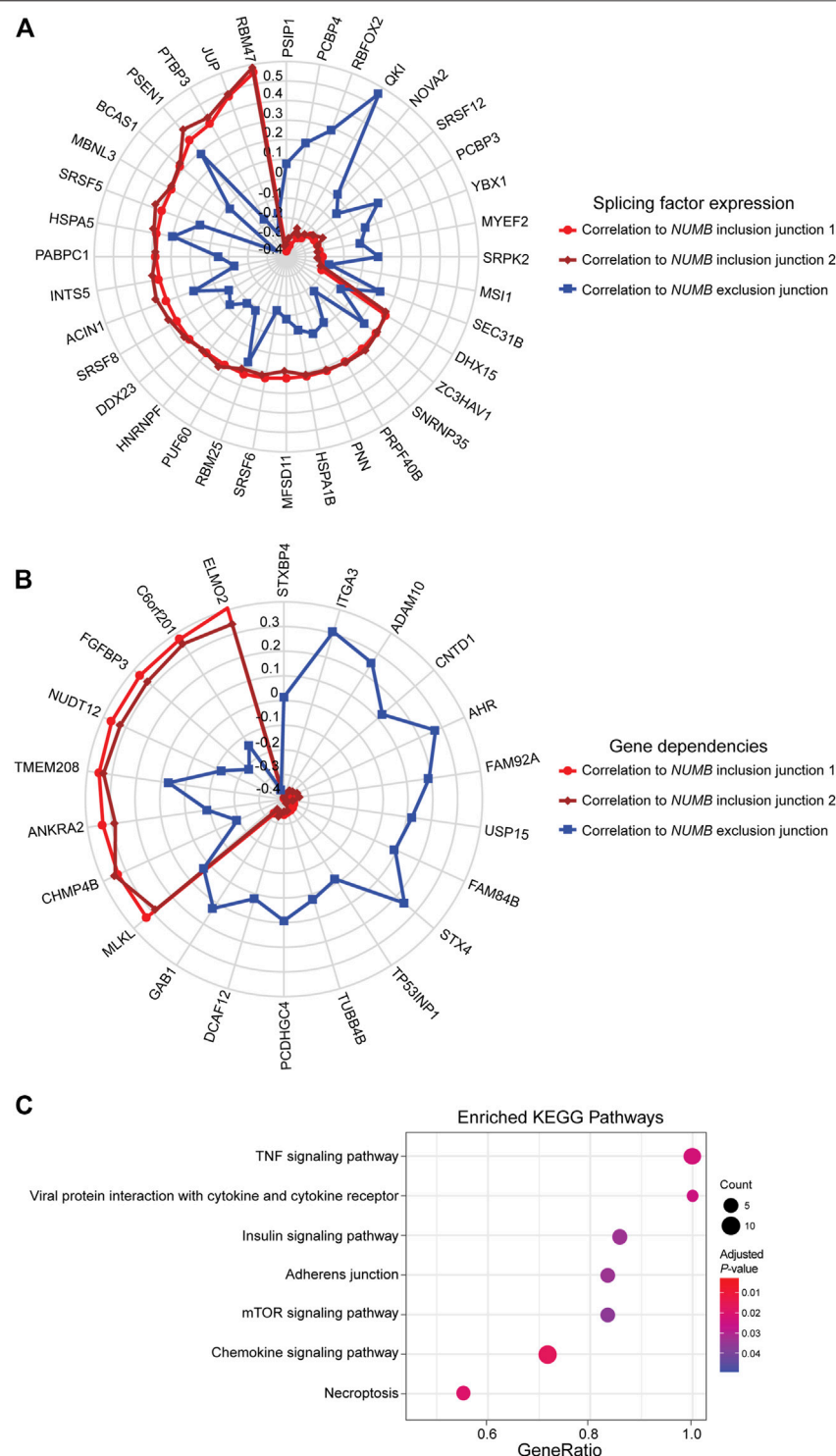


FIGURE 8 | SpliceRadar plots of top trait associations to *NUMB* alternative splicing in lung cancer. **(A)** Expression of splice junctions supporting exon 12 inclusion in *NUMB* mRNA was correlated to the expression of a panel of manually curated splicing regulators in lung cancer cell lines. The top-ranked correlation coefficients ($FDR < 0.05$ and $|\rho| > 0.2$) were used to construct the SpliceRadar chart with splicing factors depicted along the spokes, revealing a general trend of anti-correlation patterns to splicing factor expression between inclusion (red and dark red) and exclusion (blue) junctions. Previously known associations to *NUMB* splicing were corroborated (e.g., *QKI*, *RBFOX2* and *SRPK2*), and novel associations with similar correlation levels were identified, suggesting a more complex regulatory network of *NUMB* alternative splicing than previously described. **(B)** SpliceRadar plot showing top-ranked correlations ($FDR < 0.05$ and $|\rho| > 0.2$) between exon inclusion junction expression in *NUMB* and gene dependencies (defined as gene loss effect on cell survival) using DepMap CRISPR screen data. Anti-correlation patterns of dependency values and expression of inclusion and exclusion junctions are also observed as in the case of panel **(A)**. **(C)** KEGG pathway enrichment analysis using gene names of significantly associated dependencies ranked by correlation coefficient. The enrichment plot shows top over-represented pathways within *NUMB* splicing-correlated gene dependencies (Dot size represents the number of genes in each KEGG pathway, color gradient indicates significance level of adjusted *p*-values).

TABLE 3 | Summary of DJE and JT junction statistics in DJEC DB.

TCGA tissue cohort	Sample size	Quantified junctions	DE junctions	Associations to genomic alterations	Associations to mutations	Associations to pathway alterations
ACC	79	13,827,029	2,335	1	2	—
BLCA	408	14,369,479	2,935	215	274	—
BRCA	1,083	15,445,200	3,740	334	306	15
CESC	304	14,260,819	4,808	14	20	—
CHOL	36	13,786,637	8,446	10	10	—
COADREAD	372	14,315,224	5,534	49	44	—
DLBC	48	13,822,896	6,150	9	5	—
GBM	165	13,995,214	12,781	2	4	—
HNSC	500	14,592,967	5,745	49	117	2
KIPAN	738	14,965,143	2,836	92	93	1
LGG	526	14,536,867	6,771	6,708	6,061	404
LIHC	372	855,905	4,996	97	99	—
LUAD	516	14,681,817	3,931	153	149	—
LUSC	500	14,804,638	4,721	107	114	10
MESO	82	13,866,293	4,078	—	—	—
OV	199	16,204,728	8,509	9	10	—
PAAD	178	13,981,645	4,942	26	26	—
PCPG	183	14,428,362	8,973	228	228	—
PRAD	497	1,166,561	4,097	85	94	—
SARC	257	14,106,882	1,810	12	50	—
SKCM	471	14,106,882	3,436	16	11	—
STES	535	18,214,111	7,155	418	330	—
TGCT	156	14,050,087	9,684	14	14	—
THCA	500	14,437,693	4,885	699	714	37
THYM	118	13,939,486	3,860	30	31	—
UCEC	179	14,038,958	9,241	114	99	—
UCS	56	13,829,412	9,091	6	5	—
UVM	80	13,809,902	9,285	—	—	—

catalogue of clinical and molecular data, including RNA sequencing from thousands of patients across multiple tumor types. This resource harbors an excellent opportunity for cancer researchers and clinicians to explore and define tumor-specific transcriptomic signatures, and to integrate them with additional external traits such as mutations, copy number variations (CNV) or microsatellite instability (MSI). These features of TCGA can facilitate identification of novel therapeutic or diagnostic biomarkers. However, TCGA alternative splicing analyses, particularly the association of splice events with clinical and molecular traits, is currently not available in an accessible way.

To fill this gap, we generated DJEC DB, a platform that provides an integration of differential junction expression analysis with TCGA molecular and clinical data. For this, we used splice junction quantification from a recently published study (Kahles et al., 2018) where TCGA and GTEx RNA-seq samples were re-analyzed using 2-pass STAR alignment, thereby allowing identification of annotated and *de novo* splice events. Additionally, we quantified junction expression in cancer cell lines from CCLE fastq files and integrated this data with functional genomics data sets from the DepMap repository.

DJEC DB comprises four main sections: 1) Differential Junction Expression (DJE) in TCGA vs GTEx tissue, 2) Junction-Trait (JT) associations using external clinical and molecular sample data, 3) Junction Co-expression Network Analysis (JCNA) using junction expression in colorectal (COADREAD) tissue samples as example dataset, and 4)

Differential Junction Expression in cancer cell lines and association with DepMap functional genomics data (DJE-CCLE).

The DJE section comprises summary statistics and visualization options for an average of 6,345 differentially expressed junctions across the 32 tumor tissue types analyzed (FDR <0.05 and |logFC| > 2, **Table 3**). In the JT section, an average of 674 statistically significant associations are shown between differentially expressed junctions and altered oncogenic signaling pathways determined by the presence of mutations, CNVs, altered gene expression, gene fusions, DNA methylation and MSI (in the case of COADREAD tumors).

To exemplify the use of the JCNA approach, we selected the 372 samples from the TCGA COADREAD tumor cohort to construct a junction co-expression network (see methods for details). For this, we used a minimum module size of 20 junctions and an unsigned network type, meaning that the weight of connection between nodes (junctions) is calculated irrespectively of the direction of the association, so modules can contain both, positively and negatively correlated junctions (**Supplementary Figure S4**).

From a total of 7,404 junctions filtered by their gene expression-independent association to sample traits, 36 expression modules were found for this tumor type, with an average of 206 junctions per module. Module-trait associations were also determined throughout the correlation between ME expression values and tumor stage, MSI, mutations in TP53,

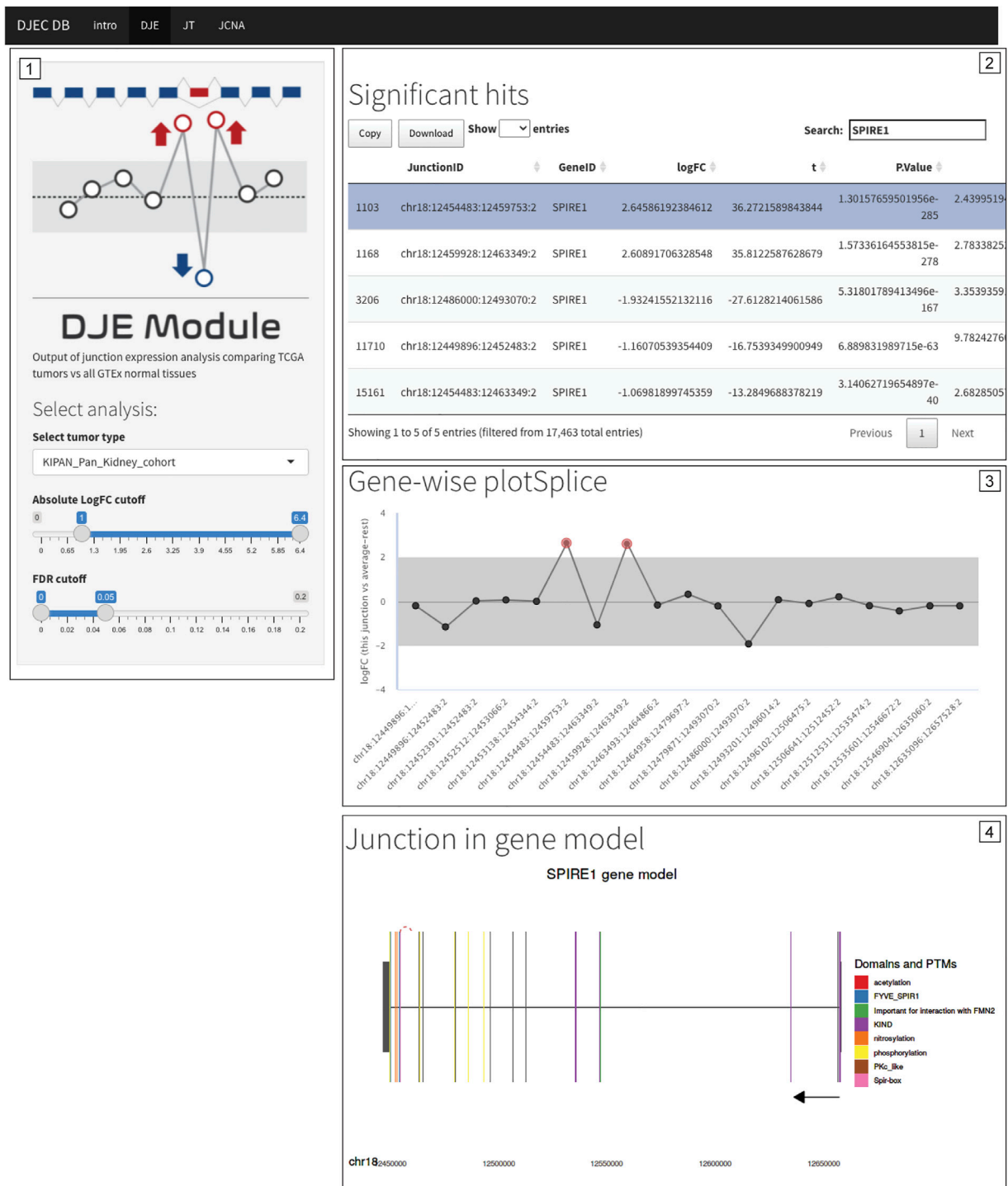


FIGURE 9 | Differentially expressed non-annotated junctions in *SPIRE1* are also found in the context of primary tumor tissue. Differential expression of junctions suggesting the presence of a non-annotated exon in *SPIRE1* mRNA were not only identified in cancer cell lines (see **Figure 7B**) but are also found in BRCA, LUAD, KIPAN, PRAD, and THCA TCGA cohorts. Caption of DJEC DB DJE analysis in KIPAN is shown as example. The exon inclusion event can be found by filtering for differentially expressed junctions following cutoff criteria of <0.05 for FDR and $|\log FC| > 1.0$ (Panel 1) and then selecting any of the two inclusion junctions based on their genomic coordinates (Panel 2). DJEC DB displays gene-wise splice plots (Panel 3) as well as domain-annotated gene model plots (Panel 4).

EGFR, KRAS and BRAF genes, as well as expression across six splicing factor gene modules previously calculated from gene expression data.

Finally, the DJE-CCLE section contains the results of the differential junction expression analysis of normal fibroblast cells vs cancer cell lines clustered by tissue of origin, as

described above. Significant correlations between junction expression and functional genomics data obtained from the DepMap repository are displayed in a summary table and selected association patterns can be visualized using SpliceRadar plots.

Search and Browse DJEC DB

Within the DJE section, users can first define the target tumor tissue type as well as the logFC and FDR cutoffs for the significance in differential expression (**Supplementary Figure S2**). A table with the summary statistics is displayed and specific target genes or junctions can be selected by the users in order to display gene-wise splice plots as well as a zoomable gene model plots with exon-to-protein domain annotation. In addition, junction-trait associations in TCGA can be explored within the JT section following user-defined tumor tissue type and external molecular trait options (**Supplementary Figure S3**).

For the JCNA section using the TCGA COADREAD sample cohort, a junction dendrogram with expression module assignment, as well as a module-trait association heatmap are displayed (**Supplementary Figure S4**). For intramodular analysis, users can select specific modules and traits to visualize module-to-trait significance plots, as well as module networks in interactive format. Both are helpful in identifying centrally located intramodular hub junctions with high module membership as well as high significance for selected traits. This allows the user to generate testable hypotheses about junction module expression, regulation and association to cancer phenotypes that can be implemented in validation experiments.

Similar interactive visualization can be also found within the DJE-CCLE section. Here, users can select the tissue of origin, the significance cutoff for differential expression, as well as target genes/junctions and junction-trait associations to be displayed in gene-wise splice and SpliceRadar plots (**Supplementary Figure S5**).

Case Study 2: Cancer Cell Line DJE Signature Is Recapitulated by Tumor Tissue Analysis in DJEC DB

One of the central features of DJEC DB is the possibility to interrogate the presence of alternative splicing patterns observed in cancer cell lines in the context of tumor tissues. *NUMB*, *VCL*, *MAP3K7* and *EXOC1* exon skipping events are examples of known splicing events that can be also observed in tumor tissue (**Supplementary Figures S12–S15**). Notably, the presence of a differentially expressed non-annotated exon between exon 12 and 13 in *SPIRE1*, which we detected in cancer cell lines (**Figure 7B**), was also identified in BRCA, LUAD, KIPAN, PRAD, and THCA cohorts by DJEC DB data using gene-wise splicing visualization (**Figure 9**). This suggests that the alternative inclusion of this previously unknown region in *SPIRE1* transcript may be a common feature across different cancer types *in vitro* and *in vivo*. These data demonstrate the applicability of DJEC DB in identifying and cross-validating potentially oncogenic alternative splicing patterns both in cancer cell lines and tumor tissue.

The JT module in DJEC DB provides a workflow to associate junction expression with user-provided molecular or clinical traits. In the case of *CTNND1* splicing event, we found significant associations between the expression of exon 20 inclusion junctions and *TP53* mutation status in BRCA, as well as with amplification of *CCND1* gene and epigenetic silencing of *CDKN2A* in STES (**Supplementary Figure S16**). This is consistent with previous studies indicating that *CCND1* isoforms expression regulates cell proliferation and cell cycle progression by controlling the levels of cyclin proteins in cancer cells (Chartier et al., 2007; Jiang et al., 2012; Liu et al., 2014).

Taken together, these data corroborate DJEC DB as a valuable bioinformatics resource for the exploration and visualization of differential junction expression, as well as for the interrogation of physiologically relevant junction-trait associations in the context of global splicing analysis in cancer cell lines and tumor tissue.

DISCUSSION

With the increasing availability of NGS data sets, the possibility to perform transcriptome-wide alternative splicing analysis has become a commonality rather than an exception in disease research. Nevertheless, computational analysis pipelines that allow the broad research community to effortlessly interrogate alternative splicing phenotypes are largely missing.

Our custom pipeline, *DJExpress*, aims to address this issue. With *DJExpress*, we have incorporated multiple existing algorithms in a novel computational approach for differential splicing analysis, which is suitable for analysis of small-scale as well as large-scale splice junction datasets. Moreover, *DJExpress* allows the analysis of millions of exon-exon boundaries per sample, using *limma*'s statistical framework. *Limma*'s algorithm has been shown to be highly accurate for gene expression analysis (Law et al., 2014; Corchete et al., 2020; Gerard, 2020), although a comprehensive analysis of accuracy for splicing is beyond the scope of this work and remains as a future direction. Nevertheless, the implication of *limma* methodology proved to be highly flexible. This is not only the case in terms of model specification (any contrast in a linear model including the use of continuous as well as categorical predictors can be related to differential junction expression) but also for the various parameters introduced into the fit model, including posterior variance estimators, observation weights and variance modelling. These features, together with *limma*'s additional data pre-processing methods such as variance stabilization, all help to improve inference of differential junction expression.

Importantly and similar to gene expression studies (Peixoto et al., 2015), removing or accounting for both known and unknown confounding factors (e.g., technical biases such as batch effects, or population structure such as molecular or clinical subtypes) is crucial when analyzing alternative splicing phenotypes in RNA-Seq data sets (Slaff et al., 2021). Confounding factors can greatly increase the numbers of false positives and negatives, which ultimately will affect interpretation of potential

biological relationships. Thus users should test for potential known confounder effects in their data, for example by using PCA or UMAP plots, and use dedicated tools to correct for confounders such as limma, ComBat, RUV, SVA and MOCCASIN (Leek, 2014; Risso et al., 2014; Zhang et al., 2020; Slaff et al., 2021).

Apart from these statistical aspects, *DJExpress* provides a comprehensive framework to graphically summarize differential splicing. The adapted *limma*-based visualization approach allows inspection of alternative splicing not only at the level of individual junction loci, but also in the presence of more complex splicing patterns. These can involve simultaneous changes in the expression of multiple junctions across the entire gene. This is particularly advantageous, considering that existing splicing analysis tools are either focused on the definition of local alternative splicing events which can be both simple (exon skipping, alternative 3' or 5' splice sites, etc.) or complex (simultaneous occurrence of multiple splice events in a given mRNA), or only allow detection of known transcript isoforms. Thus, most previous tools disregard the simultaneous visual representation of the full spectrum of up- and down-regulated splicing patterns in a gene that is retrieved through junction quantification. Broadly used exceptions are LeafCutter (Li et al., 2018) and MAJIQ (Vaquero-Garcia et al., 2016), which can both also represent complex splicing changes across the entire mRNA.

Notably, the differential junction usage analysis by *DJExpress* does not allow a direct assessment of intron retention events, which require intron and intron-exon junction read counts for their quantification. Nevertheless, dedicated tools such as MAJIQ (Vaquero-Garcia et al., 2016), IRFinder (Middleton et al., 2017), iREAD (Li et al., 2020) or S-IRFinder (Broseus and Ritchie, 2020) are specifically designed for quantification of intron retention events and are thus well-suited for this specific type of analysis.

Recently, RNA-seq data from TCGA and GTEx was integrated within a large transcriptomic profiling workflow, including splicing quantification of more than 20,000 human normal and tumor tissue samples (Kahles et al., 2018). Although this study provided unified splicing data across healthy and tumor tissue, the analysis is based on the construction of complex splicing graphs across thousands of samples and genes which are difficult to access and interpret. Furthermore, approaches to explore the data in a graphically visualized format were not the scope of this previous study. This limited the availability and accessibility of this data for the general research community as well as the feasibility of splicing-trait association analyses using genomic, epigenetic, and clinical records available within the TCGA repository. These points are addressed by *DJExpress* and DJEC DB which facilitate easy access, analysis and visualization of cancer splicing data. Moreover, by providing a simple analysis workflow for custom data sets, our pipeline is not restricted to cancer researchers but can be used to pursue a broad variety of alternative splicing-related scientific questions.

In conjunction with the usability of the *DJExpress* for differential splicing analysis and visualization using custom RNA-Seq data, the multidimensional integration of cancer data within DJEC DB represents a comprehensive resource of cancer-specific splicing signatures and junction-trait associations. We demonstrated that our pipeline has the potential to unveil novel splicing-related molecular signatures, which may contribute to improved patient stratification and more effective cancer treatment strategies. Moreover, the integration of DepMap data allows association of junction expression with molecular features such as gene dependencies and drug response profiles. This will help researchers to identify cancer cell models for specific splicing alterations that can then be used for functional characterization in the lab.

Another recently established cancer splicing repository, RJunBase (Li et al., 2021), follows a similar splicing analysis strategy as DJEC DB. While focusing on back-splice and fusion junctions, RJunBase provides splicing patterns at junction level and median junction expression information in GTEx and TCGA samples. However, it lacks differential junction expression analyses between cancer and healthy tissue and does not include association of splice events with molecular or clinical data. Thus, compared to RJunBase, DJEC DB not only includes differential junction expression analyses but also provides functional associations of splicing changes with phenotypic traits. These features make DJEC DB a comprehensive data base that can facilitate the discovery of novel cancer-related aberrant splicing patterns with potential phenotypic consequences.

Taken together, *DJExpress* provides researchers with a comprehensive toolbox for exploration of alternative splicing phenotypes in health and disease, and, with DJEC DB, includes multi-level data of alternative splicing signatures in healthy tissue, tumors and cancer cell lines.

DATA AVAILABILITY STATEMENT

GTEx and TCGA raw junction counts were provided by Dr. Andre Kahles (Biomedical Informatics Group, Department of Computer Science, ETH Zürich). All TCGA molecular and clinical data sets used in this study are publicly available and can be found here: <https://portal.gdc.cancer.gov/>. All cell line functional genomics data used in this study is publicly available and can be found here: <https://depmap.org/portal/download/>. All raw RNA-Seq data files of cell lines from CCLE are available through the Sequence Read Archive under accession number PRJNA523380. All additional data and code are available from the authors upon reasonable request. *DJExpress* R package is available at <https://github.com/MauerLab/DJExpress>. DJEC DB database is available at <https://gitlab.com/mauerlab/djecdb>.

AUTHOR CONTRIBUTIONS

JM conceived the study; LMG-P wrote the code and ran the *in-silico* analyses; LMG-P and JM wrote the manuscript.

FUNDING

This work was supported by Merck KGaA, Darmstadt, Germany (CrossRef Funder ID: 10.13039/100009945).

ACKNOWLEDGMENTS

We thank all members of the Mauer laboratory for support. We thank Arne Knudsen for testing the *DJExpress* package and for critical feedback. We also would like to thank Edith Ross, Juliane Braun and Christina Esdar (Merck KGaA) for constructive feedback and helpful discussion. **Figure 4** was created using images from iStock (<https://www.istockphoto.com>) under standard license.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fbinf.2022.786898/full#supplementary-material>

Supplementary Figure 1 | Performance evaluation of DJE module. Median (A) and log2 median (B) process time following 10 repetitions of data import (*DJEimport*), junction annotation (*DJEannotate*), expression filtering (*DJEprepare*), normalization and differential junction expression analysis (*DJEanalyze*) within the DJE module of *DJExpress*. (C) Median memory consumption (in bytes) of the entire DJE module. Error bars represent standard deviations. Default settings with increasing sample size and random relative group sizes are used in the analysis.

Supplementary Figure 2 | Interactive DJE visualization in tumors using DJEC DB. (A) Start interface of the DJE section in DJEC DB. Panel 1 highlights the selection option section. Users can define the TCGA tumor type, and the significance cutoff for differential junction usage based on minimal $|\log FC|$ and FDR values. Panel 2 shows the downloadable summary statistics table for junctions passing the selected cutoff. Here, users can filter junctions by browsing specific gene IDs, junction IDs or genomic coordinates. After selecting a target junction by clicking over it on the table, gene-wise splice plots as well as junction in domain-annotated gene model context (Panels 3 and 4 respectively) can be interactively visualized. Hovering over each junction in the gene-wise splice plot displays a box with summarized DJE information, including relative and absolute $\log FC$ values, FDR values and expression group of the selected junction. Colors within exonic regions in the gene model plot indicate the presence of protein domains and/or post-translational modifications (PTMs). The position of the selected junction within the gene model plot is indicated by a dashed arc whose color correspond to the type of differential expression (blue for downregulation and red for upregulation). Specific regions within the gene model plot (e.g., position of the selected junction) can be further explored by cursor selection, which displays a zoomed image version of the selected gene region. (B) KIF13A exon inclusion event in BRCA TCGA cohort is used as an example. Significance cutoff was set to $|\log FC| > 2.0$ and minimal FDR cutoff of 0.05. The two exon inclusion junctions are shown in red within the gene-wise splice plot, and the gene model plot indicate the position of the selected junction, which happens close to an annotated phosphorylation site of the protein.

Supplementary Figure 3 | Visualization of JT section within DJEC DB. This section contains the results of the junction-trait association analyses using ANOVA and linear models from *Matrix eQTL* methods (Shabalin, 2012). Differentially expressed junctions within each TCGA tumor type were associated to microsatellite instability (MSI) or altered oncogenic signaling pathways based on mutations, copy-number changes (CNV), mRNA expression, gene fusions and DNA methylation (Sanchez-Vega et al., 2018). Users can select the tissue of interest, as well as the trait to which junction expression is associated (Panel 1). A downloadable summary statistics table is displayed (Panel 2), where specific genes, junctions, genomic coordinates or traits can be browsed. When a specific association is selected from the table, interactive junction-trait association boxplots are displayed (Panel 3) and hovering over them shows

summarized statistics of the analysis. The image contains the example of the association between a differentially expressed junction in the transcript of S100 Calcium Binding Protein A14 (*S100A14*) and MSI, with high levels of MSI (MSI-H) in tumors (violet) being associated to significantly more inclusion levels of the junction than low levels of MSI (MSI-L) (red) and microsatellite stable (MSS) (blue) colorectal tumors.

Supplementary Figure 4 | Junction Co-expression Network Analysis (JCNA) of TCGA COADREAD in DJEC DB. (A) JCNA section comprises the results of the junction co-expression analysis across the 372 samples from the TCGA COADREAD tumor type. 7,404 junctions were clustered into 36 expression modules. The dendrogram of clustered junctions is displayed (panel 2), where each branch in the figure represents one junction, and every color below represents one co-expression module. The heatmap of module-trait associations (panel 3) based on correlation coefficients between junction modules and traits is also shown (blue and red indicate positive and negative correlations respectively). Traits are in the x-axis and junction modules with their respective assigned letter and color are in the y-axis. Traits analyzed include Microsatellite instability (MSI), BRAF, KRAS EGFR and TP53 mutation status, tumor stage and 6 co-expression modules of splicing factors calculated for COADREAD samples (SFG1-6). (B) Interactive scatter diagram of module membership vs. junction significance is shown when users select specific traits and modules within the selection options section (panel 1). (C) For the selected module, an interactive junction network is also displayed. Each node in the network represents a single junction. Junctions are colored based on gene ID. Users can select target genes within the network to highlight their respective junctions (e.g., EDEM2 junctions in the zoomed image).

Supplementary Figure 5 | Visualization of junction-trait associations using DepMap gene dependencies within JT-CCLE section in DJEC DB. This section contains the results of the junction-trait correlation analyses using junction expression and genome-wide gene dependency screens in cancer cell lines. Users can select the tissue of interest, as well as the absolute correlation coefficient cutoff to be used for SpliceRadar visualization (panel 1). A downloadable correlation matrix is displayed (panel 2), where specific genes, junctions, genomic coordinates or traits can be browsed. When specific junctions are selected (maximum 3) from the table, interactive SplicePlots with top 50 junction-dependencies correlations are displayed (panel 3). An example of significant associations between *MYO18A* exon 40 expression and gene dependencies in lymphoma cell lines is shown.

Supplementary Figure 6 | Illustration of known alternative splicing in cancer using DJEC DB. (A) Cancer-specific inclusion of exon 11 in *EXOC1* involving differentially used junctions 11, 12 and 13. The alternative splicing events occurs within the C-terminus Sec3_C domain (pink) and adjacent to several phosphorylation sites (brown) as depicted by the domain-annotated gene model plot. (B) Exon 20 inclusion event in *CTNND1*, involving junctions 20 and 23. This exon localizes at the C-terminal domain of *CTNND1* and in the vicinity of several phosphorylation sites as indicated in the gene model plot. (C) Differentially used junctions are depicted within the gene-wise splice plot in *CD44* (downregulated junction indicating the exclusion of the variable region and upregulated junctions indicating the inclusion of exons 7–14 within the variable region). Gene model plot with Prot2HG-based domain annotation indicate that the variable region in *CD44* correspond to the proteolytically cleavable extracellular Stem domain (dark gold) as previously described. For differential junction expression in *EXOC1*, *CTNND1* and *CD44*, colon, pancreatic and breast cancer cell line are shown as examples, respectively. (Numbers on the x-axis in the upper panels indicate the first, last and differentially used junctions in the respective gene. Grey area indicate threshold for significance ($|\log FC| > 1.0$). Downregulated and upregulated junctions with $|\log FC|$ above threshold and significant FDR (< 0.05) are shown in blue and red, respectively. These same junctions are indicated within the gene model plots as dashed arcs connecting upstream and downstream exons. Colors within exonic regions indicate the presence of protein domains and/or post translational modifications (PTMs) annotated within the Prot2HG protein domain database. Arrows below gene model plots indicate direction of transcription. Coding and UTR exons are illustrated as long and short exons respectively. Junctions with both absolute and relative $\log FC$ above the threshold ($|\log FC| > 1.0$) but no significant FDR (> 0.05) for at least one of them are shown in black. Junctions with either relative or absolute $\log FC$ below the indicated threshold are shown in grey).

Supplementary Figure 7 | Example local complex event in *MARK3* transcript in several cancer types. (A) Differentially used junctions as depicted in the gene-wise splice plot and gene model plot in *MARK3* indicate the presence of a splicing event involving several co-regulated junctions between exons 15–18 (the event accounts for a double exon skipping event, where several exon-exon junctions, including an

alternative 3' splice site event are downregulated). CCLE Breast cancer vs fibroblast analysis cell lines is used as example. (Numbers on the x-axis in the upper panels indicate the first, last and differentially used junctions in the respective gene. Grey area indicate threshold for significance ($|\log FC| > 1.0$). Downregulated and upregulated junctions with $|\log FC|$ above threshold and significant FDR (<0.05) are shown in blue and red, respectively. These same junctions are indicated within the gene model plots as dashed arcs connecting upstream and downstream exons. Colors within exonic regions indicate the presence of protein domains and/or post translational modifications (PTMs) annotated within the Prot2HG protein domain database. Arrows below gene model plots indicate direction of transcription. Coding and UTR exons are illustrated as long and short exons respectively. Junctions with both absolute and relative $\log FC$ above the threshold ($|\log FC| > 1.0$) but no significant FDR (>0.05) for at least one of them are shown in black). **(B)** *DJEplotSplice* function in *DJExpress* allows the alternative interactive visualization of all found junctions for a target gene within the original junction quantification data, including those removed after coverage filtering. The full gene-wise plot of *MARK3* reveals the presence of 1084 junctions detected across all analyzed samples. Junctions filtered out for differential analysis based on user-defined expression cutoffs are shown in clear grey. *DJEplotSplice* output offers an additional read coverage information across the gene using the loess fit of median junction read count (blue line) as readout. Numbers in the x-axis of the read coverage plot indicate genomic coordinates of *MARK3* gene structure.

Supplementary Figure 8 | Examples of isoform switches detected by *DJExpress* in cancer cell lines. Visualization of differentially used junctions within gene-wise splice plots and gene model plots reveals cases of upregulation and downregulation of specific transcript isoforms. **(A)** *INPP5B* gene-wise splice plot in pancreatic cancer cell lines indicates the presence of one upregulated junction and a series of consecutive downregulated junctions at the 5' region of the gene. When compared to the transcript isoform annotation for *INPP5B*, this pattern is indicative of downregulation of the long *INPP5B* isoform (bottom right) containing five additional exons at the 5' region which corresponds to the Type II inositol 1,4,5-trisphosphate 5-phosphatase PH protein domain (INPP5B_PH) (green), while the short isoform (top right) containing an alternative first exon downstream of the INPP5B_PH domain appears upregulated. **(B)** *RGS3* isoform switch is also observed in breast, colon, gastric, lung, ovarian and pancreatic cancers. The series of upregulated junctions belongs to a long isoform version of *RGS3*, while downregulated junctions correspond to a shorter transcript variant with an alternative downstream promoter. This short isoform shares its second and third exon with the long isoform but differs in four downstream exons containing the Regulator of G protein Signaling (RGS_RGS3) (brown) protein domain. *RGS3* gene-wise splice plot in gastric cell lines is shown as example (Numbers on the x-axis in the upper panels indicate the first, last and differentially used junctions in the respective gene. Grey area indicate threshold for significance ($|\log FC| > 1.0$). Downregulated and upregulated junctions with $|\log FC|$ above threshold and significant FDR (<0.05) are shown in blue and red, respectively. These same junctions are indicated within the gene model plots as dashed arcs connecting upstream and downstream exons. Colors within exonic regions indicate the presence of protein domains and/or post translational modifications (PTMs) annotated within the Prot2HG protein domain database. Arrows below gene model plots indicate direction of transcription. Coding and UTR exons are illustrated as long and short exons respectively. Junctions with both absolute and relative $\log FC$ above the threshold ($|\log FC| > 1.0$) but no significant FDR (>0.05) for at least one of them are shown in black. Junctions with either relative or absolute $\log FC$ below the indicated threshold are shown in grey).

Supplementary Figure 9 | Example of alternative splicing event types identified by *DJExpress*. Differentially used non-annotated junctions are representative of different types of alternative splicing events. **(A)** *XRCC6* gene-wise splice plot in breast cancer cell lines indicates the presence of an alternative 3' splice site (A3'SS) in exon 6. This event occurs within the Von Willebrand factor type A protein domain (VWA_ku) (pink) known to be involved in protein-protein interactions. **(B)** An alternative first exon (AFE) event is detected in *BIN1* in lymphoma cell lines. The downregulated first exon is known to contain a region required for interaction with *BIN2* (orange). **(C)** Detection of an alternative 5' splice site (A5'SS) involving the first exon of *LDLRAP1* in myeloma. **(D)** The upregulated junction in *C11orf58* in brain cancer cell lines indicates the presence of both, an alternative 5' splice site (A5'SS) and an alternative 3' splice site (A3'SS) in exon 2 and 3, respectively, which occurs inside the region corresponding to the Small acidic protein family (SAMP) domain (pink) (Numbers on the x-axis in the upper panels indicate the first, last and differentially used junctions in the respective gene. Grey area indicate threshold for significance ($|\log FC| > 1.0$). Downregulated and upregulated junctions with $|\log FC|$ above threshold and significant FDR (<0.05) are shown in blue and red, respectively. These same junctions are indicated within the gene model plots as dashed arcs connecting upstream and downstream exons. Colors within exonic regions indicate the presence of protein domains and/or post translational modifications (PTMs) annotated within the Prot2HG protein

domain database. Arrows below gene model plots indicate direction of transcription. Coding and UTR exons are illustrated as long and short exons respectively. Junctions with both absolute and relative $\log FC$ above the threshold ($|\log FC| > 1.0$) but no significant FDR (>0.05) for at least one of them are shown in black).

Supplementary Figure 10 | Example of a differentially spliced non-annotated exon in cancer cell lines. Differentially expressed non-annotated junctions indicate the presence of an exon inclusion event (junctions 18–20) between exon 17 and 18 involving the actin-binding module (LWEQ) (violet) in *TLN1* as observed in the domain-annotated gene model plot. *TLN1* plots in breast cancer cell lines are used as example (Numbers on the x-axis in the upper panels indicate the first, last and differentially used junctions in the respective gene. Grey area indicate threshold for significance ($|\log FC| > 1.0$). Downregulated and upregulated junctions with $|\log FC|$ above threshold and significant FDR (<0.05) are shown in blue and red, respectively. These same junctions are indicated within the gene model plots as dashed arcs connecting upstream and downstream exons. Colors within exonic regions indicate the presence of protein domains and/or post translational modifications (PTMs) annotated within the Prot2HG protein domain database. Arrows below gene model plots indicate direction of transcription. Coding and UTR exons are illustrated as long and short exons respectively. Junctions with both absolute and relative $\log FC$ above the threshold ($|\log FC| > 1.0$) but no significant FDR (>0.05) for at least one of them are shown in black).

Supplementary Figure 11 | SpliceRadar plot of top associations between *NUMB* alternative splicing and drug treatment response in lung cancer. Expression of splice junctions involved in the exon inclusion event of *NUMB* was correlated to cell survival rates after drug treatment using DepMap drug screens data in lung cancer cell lines. The top-ranked correlation coefficients (FDR < 0.05 and $|\rho| > 0.2$) were used to construct the SpliceRadar plot. A general trend of anti-correlation patterns with inclusion (red and dark red) and exclusion (blue) junctions are observed. Boxes indicate drugs targeting PI3K/mTOR and ERK MAPK signaling.

Supplementary Figure 12 | DJE section of DJEC DB showing summary statistics table, gene-wise splice plots and gene model plots of *NUMB* in TCGA BRCA. The two upregulated junctions indicating the inclusion of exon 12 in *NUMB* are shown in red within the gene-wise splice plot and the selected junction in the summary statistics table is also highlighted within the gene model plot (Panel 1 highlights the selection option section. Panel 2 contains the summary statistics table. Panel 3 and 4 show the gene-wise splice plot and the domain-annotated gene model plot, respectively).

Supplementary Figure 13 | Downregulation of exon 19 in *VCL* illustrated by DJE section in DJEC DB. Exon inclusion junctions are shown in blue within the gene-wise splice plot and the selected downregulated junction in the summary statistics table is also shown within the gene model plot. CESC TCGA results are shown as example (Panel 1 highlights the selection option section. Panel 2 contains the summary statistics table. Panel 3 and 4 show the gene-wise splice plot and the domain-annotated gene model plot, respectively).

Supplementary Figure 14 | Cancer-specific upregulation of exon 12 in *MAP3K7* as shown in DJEC DB. Exon inclusion and exclusion junctions are highlighted in red and blue respectively within the gene-wise splice plot. The selected upregulated junction in the summary statistics is illustrated within the gene model plot. COADREAD TCGA results are shown as example (Panel 1 highlights the selection option section. Panel 2 contains the summary statistics table. Panel 3 and 4 show the gene-wise splice plot and the domain-annotated gene model plot, respectively).

Supplementary Figure 15 | Cancer-specific alternative splicing in *EXOC1* as shown in DJEC DB. Junctions indicating the upregulation of exon 11 in *EXOC1* are shown in red within the gene-wise splice plot. The selected upregulated junction in the summary statistics is illustrated within the gene model plot. LUAD TCGA results are shown as example (Panel 1 highlights the selection option section. Panel 2 contains the summary statistics table. Panel 3 and 4 show the gene-wise splice plot and the domain-annotated gene model plot, respectively) (Panel 1 highlights the selection option section. Panel 2 contains the summary statistics table. Panel 3 and 4 show the gene-wise splice plot and the domain-annotated gene model plot, respectively).

Supplementary Figure 16 | Significant associations using *Matrix eQTL* methods between *CTNND1* exon 20 inclusion event and genomic alterations in TCGA are shown within the JT section of DJEC DB. Selecting "Associations with Genomic Alterations" and "BRCA" tumor type within the selection panel (Panel 1), followed by "CTNND1" gene ID browsing within the summary statistics table (Panel 2) displays the significant association to *TP53* mutation. Box plots show decreased exon junction expression in the presence of *TP53* mutation (MUT), compared to wild-type (WT) tumor samples (Panel 3). amplification of *CCND1* gene and epigenetic silencing of *CDKN2A* are also significantly associated to *CTNND1* alternative splicing event in TCGA STES (Panel 4).

REFERENCES

- Alamancos, G. P., Pagès, A., Trincado, J. L., Bellora, N., and Eyra, E. (2015). Leveraging Transcript Quantification for Fast Computation of Alternative Splicing Profiles. *RNA* 21, 1521–1531. doi:10.1261/rna.051557.115
- Barbosa-Morais, N. L., Irimia, M., Pan, Q., Xiong, H. Y., Gueroussov, S., Lee, L. J., et al. (2012). The Evolutionary Landscape of Alternative Splicing in Vertebrate Species. *Science* 338, 1587–1593. doi:10.1126/science.1230612
- Bechara, E. G., Sebestyén, E., Bernardis, I., Eyra, E., and Valcárcel, J. (2013). RBM5, 6, and 10 Differentially Regulate NUMB Alternative Splicing to Control Cancer Cell Proliferation. *Mol. Cell* 52, 720–733. doi:10.1016/j.molcel.2013.11.010
- Bielli, P., Panzeri, V., Lattanzio, R., Mutascio, S., Pieracciolli, M., Volpe, E., et al. (2018). The Splicing Factor PTBP1 Promotes Expression of Oncogenic Splice Variants and Predicts Poor Prognosis in Patients with Non-muscle-invasive Bladder Cancer. *Clin. Cancer Res.* 24, 5422–5432. doi:10.1158/1078-0432.CCR-17-3850
- Bray, N. L., Pimentel, H., Melsted, P., and Pachter, L. (2016). Erratum: Near-Optimal Probabilistic RNA-Seq Quantification. *Nat. Biotechnol.* 34, 888–897. doi:10.1038/nbt0816-888d
- Broseus, L., and Ritchie, W. (2020). S-IRFinder: Stable and Accurate Measurement of Intron Retention. *bioRxiv* 0625, 164699. doi:10.1101/2020.06.25.164699
- Chartier, N. T., Oddou, C. I., Lainé, M. G., Ducarouge, B., Marie, C. A., Block, M. R., et al. (2007). Cyclin-dependent Kinase 2/cyclin E Complex Is Involved in P120 Catenin (P120ctn)-dependent Cell Growth Control: A New Role for P120ctn in Cancer. *Cancer Res.* 67, 9781–9790. doi:10.1158/0008-5472.CAN-07-0233
- Chen, C., Zhao, S., Karnad, A., and Freeman, J. W. (2018). The Biology and Role of CD44 in Cancer Progression: Therapeutic Implications. *J. Hematol. Oncol.* 11, 64–73. doi:10.1186/s13045-018-0605-5
- Chen, H., Chen, X., Ye, F., Lu, W., and Xie, X. (2009). Symmetric Division and Expression of its Regulatory Gene Numb in Human Cervical Squamous Carcinoma Cells. *Pathobiology* 76, 149–154. doi:10.1159/000209393
- Chen, K. L., Li, D., Lu, T. X., and Chang, S. W. (2020). Structural Characterization of the CD44 Stem Region for Standard and Cancer-Associated Isoforms. *Int. J. Mol. Sci.* 21. doi:10.3390/ijms21010336
- Corchete, L. A., Rojas, E. A., Alonso-López, D., De Las Rivas, J., Gutiérrez, N. C., and Burguillo, F. J. (2020). Systematic Comparison and Assessment of RNA-Seq Procedures for Gene Expression Quantitative Analysis. *Sci. Rep.* 10, 19737. doi:10.1038/s41598-020-76881-X
- DepMap 21Q3 Public (2021). DepMap 21Q3 Public. Available at: https://figshare.com/articles/dataset/DepMap_21Q3_Public/15160110/2 (Accessed August 18, 2021).
- Dobin, A., Davis, C. A., Schlesinger, F., Drenkow, J., Zaleski, C., Jha, S., et al. (2013). STAR: Ultrafast Universal RNA-Seq Aligner. *Bioinformatics* 29, 15–21. doi:10.1093/bioinformatics/bts635
- Emig, D., Salomonis, N., Baumbach, J., Lengauer, T., Conklin, B. R., and Albrecht, M. (2010). AltAnalyze and DomainGraph: Analyzing and Visualizing Exon Expression Data. *Nucleic Acids Res.* 38, W755–W762. doi:10.1093/nar/gkq405
- Gallego-Paez, L. M., Bordone, M. C., Leote, A. C., Saraiva-Agostinho, N., Ascensão-Ferreira, M., and Barbosa-Morais, N. L. (2017). Alternative Splicing: the Pledge, the Turn, and the Prestige: The Key Role of Alternative Splicing in Human Biological Systems. *Hum. Genet.* 136, 1015–1042. doi:10.1007/s00439-017-1790-y
- Gardina, P. J., Clark, T. A., Shimada, B., Staples, M. K., Yang, Q., Veitch, J., et al. (2006). Alternative Splicing and Differential Gene Expression in Colon Cancer Detected by a Whole Genome Exon Array. *BMC Genomics* 7, 325. doi:10.1186/1471-2164-7-325
- Gerard, D. (2020). Data-based RNA-Seq Simulations by Binomial Thinning. *BMC Bioinformatics* 21, 206. doi:10.1186/S12859-020-3450-9
- Hu, Z., Mellor, J., Wu, J., and DeLisi, C. (2004). VisANT: An Online Visualization and Analysis Tool for Biological Interaction Data. *BMC Bioinformatics* 5, 17–18. doi:10.1186/1471-2105-5-17
- Irimia, M., Weatheritt, R. J., Ellis, J. D., Parikshak, N. N., Gonatopoulos-Pournatzis, T., Babor, M., et al. (2014). A Highly Conserved Program of Neuronal Microexons Is Misregulated in Autistic Brains. *Cell* 159, 1511–1523. doi:10.1016/j.cell.2014.11.035
- Jiang, G., Wang, Y., Dai, S., Liu, Y., Stoecker, M., Wang, E., et al. (2012). P120-catenin Isoforms 1 and 3 Regulate Proliferation and Cell Cycle of Lung Cancer Cells via β -catenin and Kaiso Respectively. *PLoS One* 7, e30303. doi:10.1371/journal.pone.0030303
- Jiang, W., and Chen, L. (2021). Alternative Splicing: Human Disease and Quantitative Analysis from High-Throughput Sequencing. *Comput. Struct. Biotechnol. J.* 19, 183–195. doi:10.1016/j.csbj.2020.12.009
- Kahles, A., Lehmann, K. V., Toussaint, N. C., Hüser, M., Stark, S. G., Sachsenberg, T., et al. (2018). Comprehensive Analysis of Alternative Splicing across Tumors from 8,705 Patients. *Cancer Cell* 34, 211–e6. doi:10.1016/j.ccell.2018.07.001
- Kahles, A., Ong, C. S., Zhong, Y., and Ratsch, G. (2016). SplAdder: Identification, Quantification and Testing of Alternative Splicing Events from RNA-Seq Data. *Bioinformatics* 32, 1840–1847. doi:10.1093/bioinformatics/btw076
- Katz, Y., Wang, E. T., Airoldi, E. M., and Burge, C. B. (2010). Analysis and Design of RNA Sequencing Experiments for Identifying Isoform Regulation. *Nat. Methods* 7, 1009–1015. doi:10.1038/nmeth.1528
- Langfelder, P., and Horvath, S. (2008). WGCNA: An R Package for Weighted Correlation Network Analysis. *BMC Bioinformatics* 9, 559. doi:10.1186/1471-2105-9-559
- Law, C. W., Chen, Y., Shi, W., and Smyth, G. K. (2014). Voom: Precision Weights Unlock Linear Model Analysis Tools for RNA-Seq Read Counts. *Genome Biol.* 15, R29–R17. doi:10.1186/gb-2014-15-2-r29
- Leek, J. T. (2014). SvaSeq: Removing Batch Effects and Other Unwanted Noise from Sequencing Data. *Nucleic Acids Res.* 42, e161. doi:10.1093/NAR/GKU864
- Li, B., and Dewey, C. N. (2011). RSEM: Accurate Transcript Quantification from RNA-Seq Data with or without a Reference Genome. *BMC Bioinformatics* 12, 323. doi:10.1186/1471-2105-12-323
- Li, H. D., Funk, C. C., and Price, N. D. (2020). IREAD: A Tool for Intron Retention Detection from RNA-Seq Data. *BMC Genomics* 21, 128. doi:10.1186/s12864-020-6541-0
- Li, Q., Lai, H., Li, Y., Chen, B., Chen, S., Li, Y., et al. (2021). RJunBase: A Database of RNA Splice Junctions in Human normal and Cancerous Tissues. *Nucleic Acids Res.* 49, D201–D211. doi:10.1093/nar/gkaa1056
- Li, Y. I., Knowles, D. A., Humphrey, J., Barbeira, A. N., Dickinson, S. P., Im, H. K., et al. (2018). Annotation-free Quantification of RNA Splicing Using LeafCutter. *Nat. Genet.* 50, 151–158. doi:10.1038/s41588-017-0004-9
- Liao, Y., Smyth, G. K., and Shi, W. (2019). The R Package Rsubread Is Easier, Faster, Cheaper and Better for Alignment and Quantification of RNA Sequencing Reads. *Nucleic Acids Res.* 47, e47. doi:10.1093/nar/gkz114
- Liu, X., Caffrey, T. C., Steele, M. M., Mohr, A., Singh, P. K., Radhakrishnan, P., et al. (2014). MUC1 Regulates Cyclin D1 Gene Expression through P120 Catenin and β -catenin. *Oncogenesis* 3 (3), e107. doi:10.1038/oncsis.2014.19
- Lonsdale, J., Thomas, J., Salvatore, M., Phillips, R., Lo, E., Shad, S., et al. (2013). The Genotype-Tissue Expression (GTEx) Project. *Nat. Genet.* 45, 580–585. doi:10.1038/ng.2653
- Lu, Y., Xu, W., Ji, J., Feng, D., Sourbier, C., Yang, Y., et al. (2015). Alternative Splicing of the Cell Fate Determinant Numb in Hepatocellular Carcinoma. *Hepatology* 62, 1122–1131. doi:10.1002/hep.27923
- Ma, C., Lv, Q., Teng, S., Yu, Y., Niu, K., and Yi, C. (2017). Identifying Key Genes in Rheumatoid Arthritis by Weighted Gene Co-expression Network Analysis. *Int. J. Rheum. Dis.* 20, 971–979. doi:10.1111/1756-185X.13063
- McGill, M. A., Dho, S. E., Weinmaster, G., and McGlade, C. J. (2009). Numb Regulates post-endocytic Trafficking and Degradation of Notch1. *J. Biol. Chem.* 284, 26427–26438. doi:10.1074/jbc.M109.014845
- Middleton, R., Gao, D., Thomas, A., Singh, B., Au, A., Wong, J. J., et al. (2017). IRFinder: Assessing the Impact of Intron Retention on Mammalian Gene Expression. *Genome Biol.* 18, 51–11. doi:10.1186/S13059-017-1184-4/FIGURES/5
- Misquitta-Ali, C. M., Cheng, E., O'Hanlon, D., Liu, N., McGlade, C. J., Tsao, M. S., et al. (2011). Global Profiling and Molecular Characterization of Alternative Splicing Events Misregulated in Lung Cancer. *Mol. Cell Biol.* 31, 138–150. doi:10.1128/mcb.00709-10
- Munkley, J., Li, L., Krishnan, S. R. G., Hysenaj, G., Scott, E., Dalglish, C., et al. (2019). Androgen-regulated Transcription of ESRP2 Drives Alternative Splicing Patterns in Prostate Cancer. *Elife* 8. doi:10.7554/eLife.47678.001

- Nishimura, T., and Kaibuchi, K. (2007). Numb Controls Integrin Endocytosis for Directional Cell Migration with aPKC and PAR-3. *Dev. Cell* 13, 15–28. doi:10.1016/j.devcel.2007.05.003
- Oh, J., Pradella, D., Kim, Y., Shao, C., Li, H., Choi, N., et al. (2021). Global Alternative Splicing Defects in Human Breast Cancer Cells. *Cancers (Basel)* 13, 3071. doi:10.3390/cancers13123071
- Oldham, M. C., Konopka, G., Iwamoto, K., Langfelder, P., Kato, T., Horvath, S., et al. (2008). Functional Organization of the Transcriptome in Human Brain. *Nat. Neurosci.* 11, 1271–1282. doi:10.1038/nn.2207
- Oltean, S., and Bates, D. O. (2014). Hallmarks of Alternative Splicing in Cancer. *Oncogene* 33, 5311–5318. doi:10.1038/ncr.2013.533
- Paronetto, M. P., Passacantilli, I., and Sette, C. (2016). Alternative Splicing and Cell Survival: From Tissue Homeostasis to Disease. *Cell Death Differ* 23, 1919–1929. doi:10.1038/cdd.2016.91
- Patro, R., Duggal, G., Love, M. I., Irizarry, R. A., and Kingsford, C. (2017). Salmon Provides Fast and Bias-Aware Quantification of Transcript Expression. *Nat. Methods* 14, 417–419. doi:10.1038/nmeth.4197
- Patro, R., Mount, S. M., and Kingsford, C. (2014). Sailfish Enables Alignment-free Isoform Quantification from RNA-Seq Reads Using Lightweight Algorithms. *Nat. Biotechnol.* 32, 462–464. doi:10.1038/nbt.2862
- Peixoto, L., Risso, D., Poplawski, S. G., Wimmer, M. E., Speed, T. P., Wood, M. A., et al. (2015). How Data Analysis Affects Power, Reproducibility and Biological Insight of RNA-seq Studies in Complex Datasets. *Nucleic Acids Res.* 43, 7664–7674. doi:10.1093/NAR/GKV736
- Presson, A. P., Sobel, E. M., Papp, J. C., Suarez, C. J., Whistler, T., Rajeevan, M. S., et al. (2008). Integrated Weighted Gene Co-expression Network Analysis with an Application to Chronic Fatigue Syndrome. *BMC Syst. Biol.* 2, 95–21. doi:10.1186/1752-0509-2-95
- Qiu, Y., Lyu, J., Dunlap, M., Harvey, S. E., and Cheng, C. (2020). A Combinatorially Regulated RNA Splicing Signature Predicts Breast Cancer EMT States and Patient Survival. *RNA* 26, 1257–1267. doi:10.1261/RNA.074187.119
- Rajendran, D., Zhang, Y., Berry, D. M., and McGlade, C. J. (2016). Regulation of Numb Isoform Expression by Activated ERK Signaling. *Oncogene* 35, 5202–5213. doi:10.1038/ncr.2016.69
- Ray, D., Yun, Y. C., Idris, M., Cheng, S., Boot, A., Iain, T. B. H., et al. (2020). A Tumor-Associated Splice-Isoform of MAP2K7 Drives Dedifferentiation in MBNL1-Low Cancers via JNK Activation. *Proc. Natl. Acad. Sci. U. S. A.* 117, 16391–16400. doi:10.1073/pnas.2002499117
- Risso, D., Ngai, J., Speed, T. P., and Dudoit, S. (2014). Normalization of RNA-Seq Data Using Factor Analysis of Control Genes or Samples. *Nat. Biotechnol.* 32, 896–902. doi:10.1038/NBT.2931
- Ritchie, M. E., Phipson, B., Wu, D., Hu, Y., Law, C. W., Shi, W., et al. (2015). Limma powers Differential Expression Analyses for RNA-Sequencing and Microarray Studies. *Nucleic Acids Res.* 43, e47. doi:10.1093/nar/gkv007
- Ryan, M. C., Cleland, J., Kim, R., Wong, W. C., and Weinstein, J. N. (2012). SpliceSeq: A Resource for Analysis and Visualization of RNA-Seq Data on Alternative Splicing and its Functional Impacts. *Bioinformatics* 28, 2385–2387. doi:10.1093/bioinformatics/bts452
- Sanchez-Vega, F., Mina, M., Armenia, J., Chatila, W. K., Luna, A., La, K. C., et al. (2018). Oncogenic Signaling Pathways in the Cancer Genome Atlas. *Cell* 173, 321–e10. doi:10.1016/j.cell.2018.03.035
- Saraiva-Agostinho, N., and Barbosa-Morais, N. L. (2019). Psychomics: Graphical Application for Alternative Splicing Quantification and Analysis. *Nucleic Acids Res.* 47, e7. doi:10.1093/nar/gky888
- Scotti, M. M., and Swanson, M. S. (2016). RNA Mis-Splicing in Disease. *Nat. Rev. Genet.* 17, 19–32. doi:10.1038/nrg.2015.3
- Sebestyen, E., Zawisza, M., and Eyra, E. (2015). Detection of Recurrent Alternative Splicing Switches in Tumor Samples Reveals Novel Signatures of Cancer. *Nucleic Acids Res.*
- Shabalín, A. A. (2012). Matrix eQTL: Ultra Fast eQTL Analysis via Large Matrix Operations. *Bioinformatics* 28, 1353–1358. doi:10.1093/bioinformatics/bts163
- Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., et al. (2003). Cytoscape: A Software Environment for Integrated Models of Biomolecular Interaction Networks. *Genome Res.* 13, 2498–2504. doi:10.1101/gr.1239303
- Shen, S., Park, J. W., Lu, Z. X., Lin, L., Henry, M. D., Wu, Y. N., et al. (2014). rMATS: Robust and Flexible Detection of Differential Alternative Splicing from Replicate RNA-Seq Data. *Proc. Natl. Acad. Sci. U. S. A.* 111, E5593–E5601. doi:10.1073/pnas.1419161111
- Shirure, V. S., Liu, T., Delgadillo, L. F., Cuckler, C. M., Tees, D. F., Benencia, F., et al. (2015). CD44 Variant Isoforms Expressed by Breast Cancer Cells Are Functional E-Selectin Ligands under Flow Conditions. *Am. J. Physiol. Cell Physiol* 308, C68–C78. doi:10.1152/ajpcell.00094.2014
- Slaff, B., Radens, C. M., Jewell, P., Jha, A., Lahens, N. F., Grant, G. R., et al. (2021). MOCCASIN: a Method for Correcting for Known and Unknown Confounders in RNA Splicing Analysis. *Nat. Commun.* 12, 1–9. doi:10.1038/s41467-021-23608-9
- Stanek, D., Bis-Brewer, D. M., Saghira, C., Danzi, M. C., Seeman, P., Lassuthova, P., et al. (2020). Prot2HG: A Database of Protein Domains Mapped to the Human Genome. *Database (Oxford)* 2020, 161. doi:10.1093/database/baz161
- Sterne-Weiler, T., Weatheritt, R. J., Best, A. J., Ha, K. C. H., and Blencowe, B. J. (2018). Efficient and Accurate Quantitative Profiling of Alternative Splicing Patterns of Any Complexity on a Laptop. *Mol. Cell* 72, 187–e6. doi:10.1016/j.molcel.2018.08.018
- Teekchandani, A., Toida, N., Goodchild, J., Henderson, C., Watts, J., Wollscheid, B., et al. (2009). Quantitative Proteomics Identifies a Dab2/integrin Module Regulating Cell Migration. *J. Cell Biol.* 186, 99–111. doi:10.1083/jcb.200812160
- Thorsen, K., Sørensen, K. D., Brems-Eskildsen, A. S., Modin, C., Gaustadnes, M., Hein, A. M., et al. (2008). Alternative Splicing in colon, Bladder, and Prostate Cancer Identified by Exon Array Analysis. *Mol. Cell. Proteomics* 7, 1214–1224. doi:10.1074/mcp.M700590-MCP200
- Tomczak, K., Czerwińska, P., and Wiznerowicz, M. (2015). The Cancer Genome Atlas (TCGA): An Immeasurable Source of Knowledge. *Contemp. Oncol. (Pozn)* 19, A68–A77. doi:10.5114/wo.2014.47136
- Trapnell, C., Pachter, L., and Salzberg, S. L. (2009). TopHat: Discovering Splice Junctions with RNA-Seq. *Bioinformatics* 25, 1105–1111. doi:10.1093/bioinformatics/btp120
- Trapnell, C., Williams, B. A., Pertea, G., Mortazavi, A., Kwan, G., Van Baren, M. J., et al. (2010). Transcript Assembly and Quantification by RNA-Seq Reveals Unannotated Transcripts and Isoform Switching during Cell Differentiation. *Nat. Biotechnol.* 28, 511–515. doi:10.1038/nbt.1621
- Vaquero-Garcia, J., Barrera, A., Gazzara, M. R., González-Vallinas, J., Lahens, N. F., Hogenesch, J. B., et al. (2016). A New View of Transcriptome Complexity and Regulation through the Lens of Local Splicing Variations. *Elife* 5, e11752. doi:10.7554/eLife.11752
- Verdi, J. M., Bashirullah, A., Goldhawk, D. E., Kubu, C. J., Jamali, M., Meakin, S. O., et al. (1999). Distinct Human NUMB Isoforms Regulate Differentiation vs. Proliferation in the Neuronal Lineage. *Proc. Natl. Acad. Sci. U. S. A.* 96, 10472–10476. doi:10.1073/pnas.96.18.10472
- Vieira, S. E., Bando, S. Y., De Paulis, M., Oliveira, D. B. L., Thomazelli, L. M., Durigon, E. L., et al. (2019). Distinct Transcriptional Modules in the Peripheral Blood Mononuclear Cells Response to Human Respiratory Syncytial Virus or to Human Rhinovirus in Hospitalized Infants with Bronchiolitis. *PLoS One* 14, e0213501. doi:10.1371/journal.pone.0213501
- Wang, E. T., Sandberg, R., Luo, S., Khrebtkova, I., Zhang, L., Mayr, C., et al. (2008). Alternative Isoform Regulation in Human Tissue Transcriptomes. *Nature* 456, 470–476. doi:10.1038/nature07509
- Wang, K., Singh, D., Zeng, Z., Coleman, S. J., Huang, Y., Savich, G. L., et al. (2010). MapSplice: Accurate Mapping of RNA-Seq Reads for Splice Junction Discovery. *Nucleic Acids Res.* 38, e178. doi:10.1093/nar/gkq622
- Wang, Y., Chen, S. X., Rao, X., and Liu, Y. (2020). Modulator-Dependent RBPs Changes Alternative Splicing Outcomes in Kidney Cancer. *Front. Genet.* 11, 265. doi:10.3389/fgene.2020.00265
- Wang, Z., Sandiford, S., Wu, C., and Li, S. S. (2009). Numb Regulates Cell-Cell Adhesion and Polarity in Response to Tyrosine Kinase Signalling. *EMBO J.* 28, 2360–2373. doi:10.1038/emboj.2009.190
- Wang, Z., Zhao, K., Hackert, T., and Zöller, M. (2018). CD44/CD44v6 a Reliable Companion in Cancer-Initiating Cell Maintenance and Tumor Progression. *Front. Cell Dev. Biol.* 6, 97. doi:10.3389/fcell.2018.00097
- Wilcox, R. R. (2012). *Introduction to Robust Estimation and Hypothesis Testing*. doi:10.1016/C2010-0-67044-1
- Yanagisawa, M., Huvelde, D., Kreinest, P., Lohse, C. M., Cheville, J. C., Parker, A. S., et al. (2008). A P120 Catenin Isoform Switch Affects Rho Activity, Induces Tumor Cell Invasion, and Predicts Metastatic Disease. *J. Biol. Chem.* 283, 18344–18354. doi:10.1074/jbc.M801192200

- Zhang, B., and Horvath, S. (2005). A General Framework for Weighted Gene Co-expression Network Analysis. *Stat. Appl. Genet. Mol. Biol.* 4, Article17. doi:10.2202/1544-6115.1128
- Zhang, S., Bao, Y., Shen, X., Pan, Y., Sun, Y., Xiao, M., et al. (2020a). RNA Binding Motif Protein 10 Suppresses Lung Cancer Progression by Controlling Alternative Splicing of Eukaryotic Translation Initiation Factor 4H. *EBioMedicine* 61, 103067. doi:10.1016/j.ebiom.2020.103067
- Zhang, S., Liu, Y., Liu, Z., Zhang, C., Cao, H., Ye, Y., et al. (2014). Transcriptome Profiling of a Multiple Recurrent Muscle-Invasive Urothelial Carcinoma of the Bladder by Deep Sequencing. *PLoS One* 9, e91466. doi:10.1371/journal.pone.0091466
- Zhang, Y., Parmigiani, G., and Johnson, W. E. (2020b). ComBat-seq: Batch Effect Adjustment for RNA-Seq Count Data. *NAR Genom Bioinform* 2, lqaa078. doi:10.1093/NARGAB/LQAA078
- Zong, F. Y., Fu, X., Wei, W. J., Luo, Y. G., Heiner, M., Cao, L. J., et al. (2014). The RNA-Binding Protein QKI Suppresses Cancer-Associated Aberrant Splicing. *Plos Genet.* 10, e1004289. doi:10.1371/journal.pgen.1004289

Conflict of Interest: LG-P and JM are employees of BioMed X Institute (GmbH), Heidelberg, Germany. Merck KGaA had no part in the study design and collection, analysis, and interpretation of the results but provided feedback regarding the general research strategy.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Gallego-Paez and Mauer. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Strategies for the Production of Molecular Animations

Erik Werner*

RNS Berlin, Berlin, Germany

OPEN ACCESS

Edited by:

Sean O'Donoghue,
Garvan Institute of Medical Research,
Australia

Reviewed by:

Jean-Karim Hériché,
European Molecular Biology
Laboratory Heidelberg, Germany
Christopher Hammang,
The University of Sydney, Australia

*Correspondence:

Erik Werner
erik.werner@rns.berlin

Specialty section:

This article was submitted to
Data Visualization,
a section of the journal
Frontiers in Bioinformatics

Received: 12 October 2021

Accepted: 14 April 2022

Published: 16 May 2022

Citation:

Werner E (2022) Strategies for the
Production of Molecular Animations.
Front. Bioinform. 2:793914.
doi: 10.3389/fbinf.2022.793914

Molecular animations play an increasing role in scientific visualisation and science communication. They engage viewers through non-fictional, documentary type storytelling and aim at advancing the audience. Every scene of a molecular animation is to be designed to secure clarity. To achieve this, knowledge on design principles from various design fields is essential. The relevant principles help to draw attention, guide the eye, establish relationships, convey dynamics and/or trigger a reaction. The tools of general graphic design are used to compose a signature frame, those of cinematic storytelling and user interface design to choreograph the relative movement of characters and cameras. Clarity in a scientific visualisation is reached by simplification and abstraction where the choice of the adequate representation is of great importance. A large set of illustration styles is available to choose the appropriate detail level but they are constrained by the availability of experimental data. For a high-quality molecular animation, data from different sources can be integrated, even filling the structural gaps to show a complete picture of the native biological situation. For maintaining scientific authenticity it is good practice to mark use of artistic licence which ensures transparency and accountability. The design of motion requires knowledge from molecule kinetics and kinematics. With biological macromolecules, four types of motion are most relevant: thermal motion, small and large conformational changes and Brownian motion. The principles of dynamic realism should be respected as well as the circumstances given in the crowded cellular environment. Ultimately, consistent complexity is proposed as overarching principle for the production of molecular animations and should be achieved between communication objective and abstraction/simplification, audience expertise and scientific complexity, experiment and representation, characters and environment as well as structure and motion representation.

Keywords: molecular animation, scientific visualisation, consistent complexity, design, advance the audience, cinematic storytelling, molecule motion, dynamic realism

1 INTRODUCTION

Modern technology makes video an easily accessible and therefore omnipresent medium in our lives and therefore also in the fields of knowledge and communication where it unfolds its full potential in the form of molecular animations. Molecular animation can be described as motion design for biological macromolecules. Since these nano-scale characters are not directly visible to the human eye, we need to draw on an array of visualisation methods to communicate them. Combining scientific illustration with motion gives us the opportunity to visualise the dynamics of the molecular system.

Molecular animations can be used in science communication, education and research (Iwasa, 2010 and 2015). They create interest, increase memory and lead to better comprehension of complex subjects. In research, they provide researchers an insight into processes by summarising and contextualising a mechanism and can also support grant applications, marketing, social and environmental campaigns and many more.

In 2008 Gael McGill announced “Molecular Movies ... Coming to a Lecture near You” (McGill, 2008), describing the upcoming trend to use professional 3D software known from movies. In 2010 a group of experts met at the Workshop on Molecular Animation in San Francisco (Bromberg, Chiu and Ferrin, 2010) to discuss needs and requirements. Since then, several major publications appeared focused on scientific visualisation (O'Donoghue et al., 2010a; O'Donoghue et al., 2010b; Johnson and Hertig, 2014; Kozlíková et al., 2017; Goodsell and Jenkinson, 2018; Olson, 2018) or visualisation software (Goddard and Ferrin, 2007; Martinez et al., 2019). In addition, the potential of animations has been highlighted by the exemplary work of Iwasa (Iwasa, 2010 and 2015), Berry (TED, 2012) and others, and software development, for example of ePMV (Johnson et al., 2011) and Molecular Maya (McGill, 2010), enable the use of structural data in professional 3D software.

This perspective article formulates a number of guidelines with relevance for molecular animations based on knowledge and literature from the main fields design, scientific visualisation, molecular kinetics/kinematics and cinematography/storytelling. It may contribute to a theoretical basis for the field of scientific and especially molecular animation.

Molecular animation is understood here as the visualisation of the structure and dynamic of macromolecular biomolecules and their substrates within the context of the living cell, at the molecular nano-scale. Molecular animation therefore can be seen as a subspace of data visualisation (with structural and related dynamic data being a subset of all scientific data) and medical illustration/animation, that includes the biological mesoscale (larger than molecular complexes, smaller than a cell; see Johnson (VIZBI, 2012b), Le Muzic et al. (2014) and Goodsell et al. (2020) for details) and macroscale dimensions (cells, organelles, organs, organisms). Consequently, this article concentrates on aspects and principles most relevant to molecular animations and may omit some others. Please see the Supplement for a detailed description of the methodology used in deriving these guidelines.

2 ADVANCE THE AUDIENCE AND ENGAGE IT THROUGH STORYTELLING

2.1 Build the Basics and Advance the Audience

Every design object should serve a purpose that benefits the user. A molecular animation is an audiovisual design object, usually aimed at a specific target audience whose expertise level may vary; see McGill (VIZBI, 2012a) and Johnson and Hertig (2014). For an audience to benefit from an animation, it is essential to adjust the complexity to the actual expertise level. So it is deemed a good

idea to introduce a topic with basic knowledge and allow everybody to connect, irrespective of their expertise. The higher the audience's expertise level and audiovisual literacy the shorter the introduction can be. An animation may quickly go into the latest results and very complex detail when it addresses advanced experts. However, it may still be necessary to explain the basic principles, the relevant visual conventions and also to refresh the memory of a viewer.

The audience generally engages with the animation to learn something new and interesting. It should therefore be the goal of every animation to advance the audience - to introduce something new, more complex, more challenging - and allow them to extend their knowledge (Johnson and Hertig, 2014). This means that the complexity level of the animation can go at least one step further than the one that is indicated by the expertise limit of the typical viewer.

2.2 Adjust the Video Output to Reflect the Consumption Scenario

A user consumes a molecular animation through a digital screen, either in a guided presentation or in a stand-alone format, for example on a video platform, which directly influences the time of engagement (Frankel and DePace, 2012). In a presentation format, a speaker usually guides the audience through the animation within a larger context adjusted for the specific expertise level of the actual audience. The animation is usually shown only once and therefore needs to put special emphasis on clarity and simplicity. A stand-alone animation can be paused and repeated and therefore allows more complexity. The content of an animation can be adjusted to those different scenarios through an output strategy that makes use of a modular toolbox and animation helpers such as labels, sound, voiceover, subtitles or annotations. A full parent version includes all available scenes and covers all relevant communication objectives. For a specific scenario, a selection of scenes serves as a derivative.

2.3 Use Filmmaking Production Techniques

The output strategy for an animation should be planned at the very beginning of the production, a process that follows the three production stages similar to a movie; please see Sharpe et al. (2008), Jantzen et al. (2015) and Lepito (2018) for details. The pre-production stage includes the agreement on the communication objectives, decisions on a look and feel (style, colour, typography, narration, etc.), discovery of the story, scripting, storyboarding, creation of animatics and the output strategy. The production phase includes the creation of models and their dynamic animation as well as the implementation of lighting, cameras, materials, textures and shaders to create a render of each frame. In the post-production stage, individual sequences are combined into a composite, combined into a final edit with labels and sound and finally rendered out in a delivery format. While a molecular movie requires special knowledge mainly in the first two stages, post-production does not fundamentally

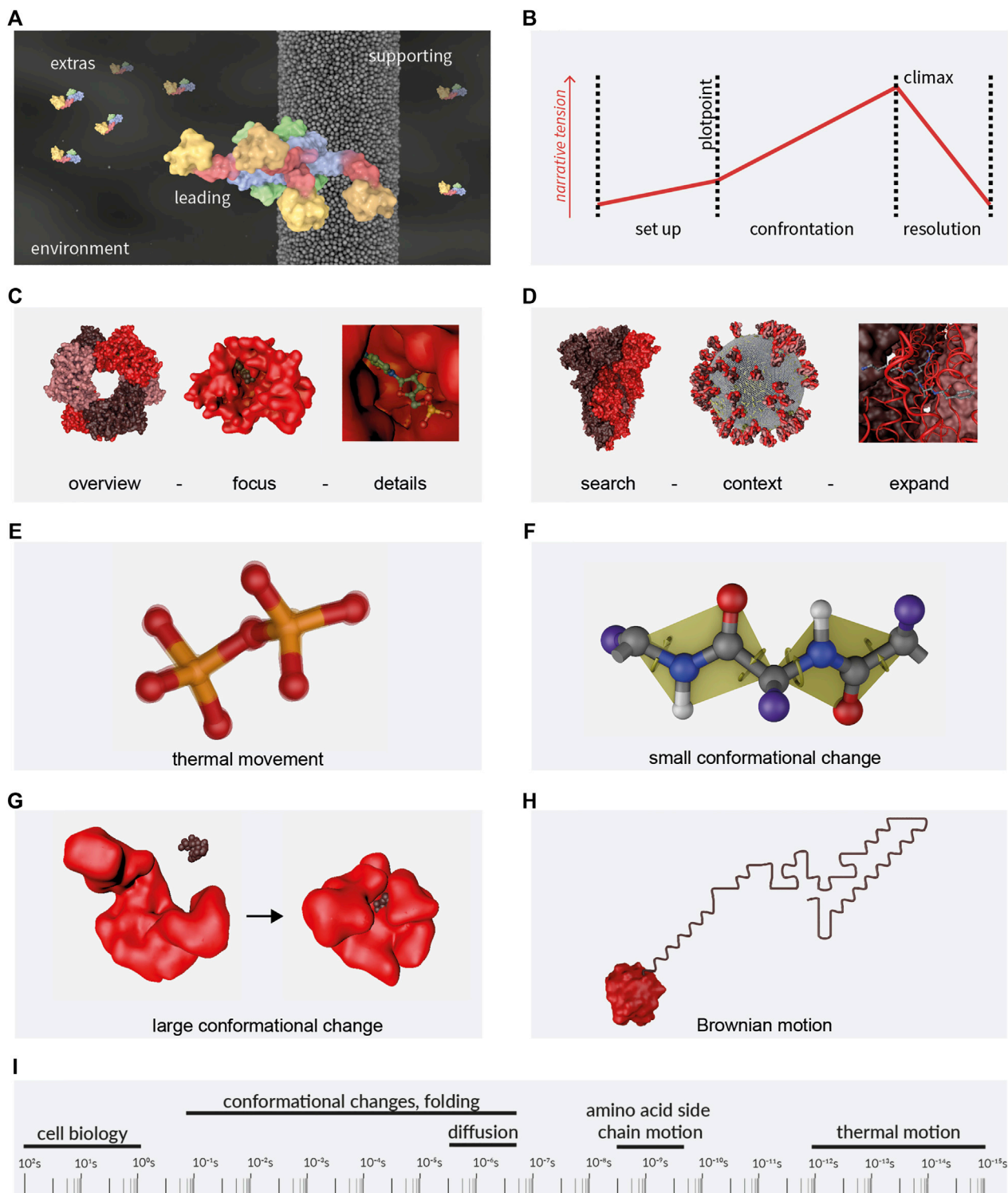


FIGURE 1 | Storytelling and motion. **(A)** Molecular Actors. **(B)** Molecular Storytelling. The three stages of a classical story. Presentation Order, **(C)** overview first (hexamer in the example), zoom (monomer with ligand) and filter (ligand details); and **(D)** search (spike protein), show context (whole virus), expand (detail of the spike protein hinge region). Visualisation of the movement in/of macromolecules: **(E)** thermal movement **(F)** small and **(G)** large conformational changes and **(H)** Brownian motion. **(I)** Timescales of biological processes. Molecules depicted are: **(A)** Dynamin1, PDB code 3SNH, (Faelber et al., 2011); **(C)** and **(H)** NMNAT1, PDB code 1GZU (Werner et al., 2002); **(D)** SARS-CoV-2 proteins S [PDB code 1KDI, (Gobeil et al., 2021)], E [PDB code 7K3G (Mandala et al., 2020)], M by Mahtarin et al. (2020) and Swiss-Model entries for P0DTC4; **(G)** Adenylat Kinase, PDB codes 4AKE (Müller et al., 1996) and 2ECK (Berry et al., 2006). PDB: Protein Database (Berman et al., 2000); Swiss-Model Repository (Bienert et al., 2017).

differ from most video or movie projects. Subsequent chapters therefore concentrate on the preparation and production of molecular movies.

2.4 Engage the Audience Through Storytelling

A typical research project is structured as a chronological sequence of concept/hypothesis development, planning, experiment, data analysis and data interpretation. However, many research reports already use narrative elements in the IMRAD format to present the knowledge: introduction (exposition), methods (rising action), results (climax), analysis (falling action) and discussion (resolution) (ElShafie, 2018). In an animation, we can make use of a narrative. Storytelling can attract the audience's attention, make them care and leave a lasting impression by including stakes and allowing the audience to relate to the story (Ma et al., 2012; Lepito, 2018). This makes the science more meaningful to them without compromising on scientific accuracy, objectivity and therefore credibility (ElShafie, 2018). Due to the persuasive nature of narratives, science animators need to include ethical considerations related to the underlying communication objective (persuasion or comprehension), the level of accuracy (external realism, representativeness) or the use of narrative at all, especially when addressing non-expert audiences (Dahlstrom, 2014).

The cinematic genre of molecular animations is best described as non-fictional documentary. Documentary storytellers must not invent and cannot make compromises when it comes to the facts. Instead, they need to be guided by and find the story in the material itself (Bernard, 2010). The story itself may be one of exploration, where the researchers are portrayed on their journey to discovery (Berlin, 2016). This is comparable to the well known narrative of the hero's journey (Vogler, 2007). However, the molecular story may as well stand on its own. The molecular characters in an animation can be seen as playing roles similar to actors in a movie, even though they do not make conscious decisions but rather follow the laws of physics. Main characters carry the story, side characters support it and extras create the background, see **Figure 1A** for an example. All of them act in an environment that can have a strong influence on the story by setting the location and external conditions.

Contextualised in the wider field of data visualisation, a molecular animation belongs to one of seven genres of narrative visualisation: film/video/animation. The narrative structure tactics is strongly author-driven. As such, an animation is characterised by linear ordering of scenes, heavy use of labels, headlines and annotations (messaging) and a lack of interactivity (Segel and Heer, 2010). The author is in full control of the animation which constitutes passive storytelling (Wohlfart and Hauser, 2007).

2.5 Chose the Story Structure

Any story can be characterised by the three act structure that goes back to Aristotle's Poetics (see the english translation, (Aristotle, 1996)) and **Figure 1B**) and includes a setting (establishment of environment and characters; act 1), a plot with rising narrative

tension (act 2, the protagonist on a pursuit) and a resolution (act 3), see also ElShafie (2018). The structure of the story however does not have to be linear. It is determined by the tools of cinematography, editing and compositing. A viewer can get a certain understanding of the topic through an overview that shows all involved elements at the same time. It helps creating a reliable and recognisable framework to come back to when needed. While zooming into detail, the content is filtered and unnecessary element and details are left out. This follows Ben Shneidermann's visual information-seeking mantra "overview first, zoom and filter, then details on demand" (Shneiderman, 1996) (**Figure 1C**). For datasets with high complexity, an alternative is: "search, show context, expand", where we begin with a starting point, reflect on the contextual aspects and expand further context and detail when needed (**Figure 1D**; Munzner, 2014). Other story structures include comparative visualisation (side-by-side comparison) and iterative visualisation (a repetitive pattern when focusing on several features in the same context), see Wohlfart and Hauser (2007) and Ma et al. (2012).

3 DESIGN EVERY SCENE TO SECURE CLARITY

Many molecular animation concepts include a storyboard with illustrated signature frames, the narration and possibly animatics for the timing. The more complex a topic and an animation are, the more important a storyboard becomes. Every signature frame, the transitions between them and the relative movement within a scene need to be designed to achieve the specific communication objective of that scene. Technically, this can be achieved by keyframe interpolation, particle or molecular dynamics with defined starting points and dynamic field parameters.

3.1 Follow Design Principles

The signature frames are individual images that represent important situations of the story. The molecular animator should be able to create a clear design for them and therefore have a good knowledge on design principles from various fields, most importantly graphic design, motion design, user interface design and cinematography/film. The design principles can be categorised into five (partially overlapping) areas: draw attention, guide the eye, establish relationships, convey dynamics and create emotion /reaction. The methodology for the selection of principles relevant for molecular animations is described in the Supplement, including a visualisation of the principles in **Supplementary Figures 9S–13S**; those from general graphic design are mainly based on monographs "Graphik und Gestaltung" (Wäger, 2014) and "Perception of Design" (Ware, 2012).

3.2 Use the Tools of Cinematography

An animation is created by moving from signature frame to signature frame and includes the relative movement of characters and camera view. Characters may enter, stay in or leave the frame. Or they can move with the camera relative to other characters or the environment. Also, camera

movement can be combined with character movement. The techniques and principles of non-dialogue cinematic storytelling (Sijl, 2005; Mercado, 2010; Raschke, 2013) help to reach the communication objectives and include setting the look and feel through composition and lighting and finally positioning and moving the camera through a scene while maintaining that look and feel. The attention of the audience, its emotions and interest are led by changing these parameters and also the sound design. Viewing axes and depth of field establish the relation between characters and both character and camera motion establish an order of events. The choice of focal length, editing, transitions and time alterations all play important roles and are chosen dependent of the communication objective. Overreaching principles from cinematographic storytelling include “story is king”, were all elements visible on the screen support the story and “show, not just tell” where the eye is guided by visual highlighting (Lepito, 2018).

3.3 Learn From User Interface Design

The motion lessons of interface design (material.io/Google, 2021) support the choreography of character movements. They deal with the speed of incoming and outgoing elements, their duration in the frame and the grouping of movements based on the complexity. This way, we can define the movement path or the fade-in/fade-out properties of the characters.

3.4 Ignore the “Disney Animation Principles”

It needs to be mentioned, that the well known set of “Disney animation principles” (Thomas and Johnston, 1981) does not apply to the nano-world of molecules. They cover *timing and spacing, easing, mass and weight, squash and stretch, follow through and overlapping, secondary action, arcs, solid drawing, anticipation, exaggeration, staging and appeal*. With the help of those principles, natural movements are recreated based on material properties, following the laws of classical physics and building on every-day-life experience in the macro world. An experience that does not exist in the nano-world of atoms and molecules. Here, movement is determined by random collisions, diffusion gradients and thermal motion. So, with the exception of the more general principles *staging* and *appeal*, those for the design of motion need to be set aside for the animations of molecules.

4 CHOSE ADEQUATE REPRESENTATION TO ILLUSTRATE CURRENT KNOWLEDGE

4.1 Simplify and Abstract

Clarity is the overall goal of any design process and it is therefore also important for scientific visualisation in general. It is often reached by simplification (displaying fewer items) and abstraction (using simpler forms of an item), or both in combination. Clarity avoids clutter in the frame while unburdening the perceptual system of the viewer. However, it needs to be carefully balanced with the addition of more complex detail in order to advance the audience. In fact, in the nano-world

of macromolecules, there is always a certain level of abstraction involved in scientific visualisations which automatically leaves room for interpretation (Sharpe et al., 2008). A taxonomy of types of abstraction includes symbolic representations, schematic diagrams, graphs, cartoons and realistic representations (Offerdahl, et al., 2018; Goodsell and Jenkinson, 2018). They all can play a role in molecular animations and need to be chosen dependent on the audience expertise level and the specific communication objective.

4.2 Chose a Representation of Biological Macromolecules

The visualisation of a biological macromolecule (molecular graphics) can range from very simple to very complex. Illustrative and abstract representations include 1D formats (letter codes), 2D formats (letter-code with crosslinks, schematic) and 3D formats (arbitrary organic shapes, backbone and ribbon/cartoon representations). 3D-surface abstractions include beads representations (one bead per subunit, e.g., amino acids) and coarse approximation. 3D atomistic surface models show more detail and include convolution surface models (e.g., Gaussian), molecular skin surfaces, ligand excluded surfaces, solvent excluded surfaces and solvent accessible surfaces. 3D atomistic space filling models are characterised by each atom being represented by a sphere with a radius based on the Van der Waals radii. 3D atomistic bond-centric models include hyperballs, licorice and ball-and-stick representations or even quantum mechanical models. Please see Kozlíková et al. (2017), Goodsell and Jenkinson (2018) and Olson (2018) for detailed descriptions and Johnson and Hertig (2014), Biocinematics (2016) and O'Donoghue et al. (2010b) for overviews. **Figure 2B** includes the representations of a short beta-strand polypeptide with increasing complexity from bottom to top.

4.3 Respect Constraints

Choosing the adequate representation can be challenging and depends on several factors like the actual, specific communication objective, the experimental data available, the topical context, the target audience expertise level, and others. To avoid misconceptions, it is recommended to avoid causing superficial understanding due to over-simplification and abstraction. A depiction may be taken literally and not interpreted according to underlying scientific knowledge. Whole biological concepts can be basically understood like this, but actual insight may not go beyond the simplicity of the representation explaining it (Goodsell and Jenkinson, 2018). Mixed representations are popular to highlight details and help to establish the character relationships.

4.4 Ensure Scientific Authenticity and Transparency

At the same time it should also be avoided to imply more knowledge than the data actually provides. The quality and resolution of the available experimental data restricts the level

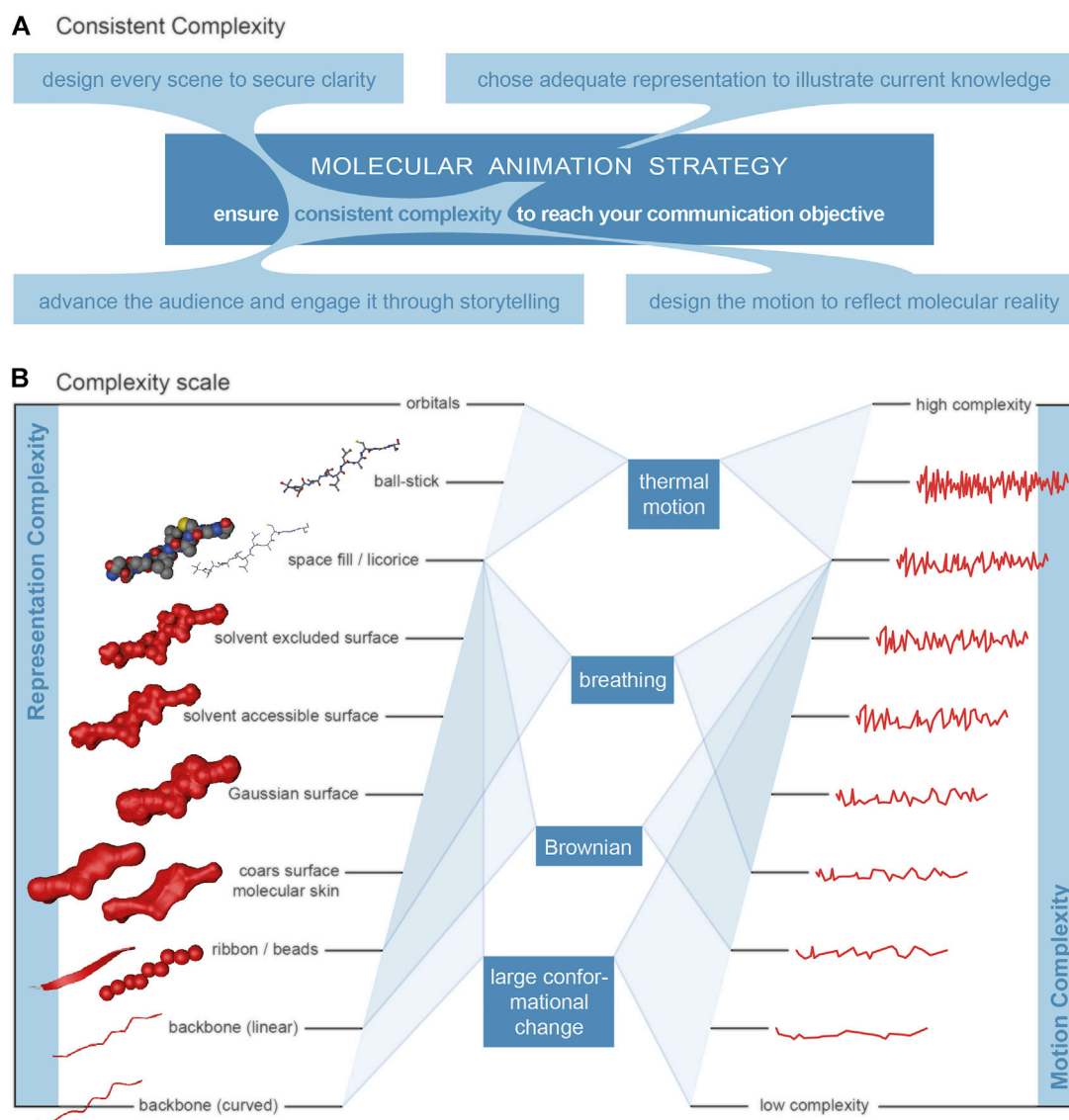


FIGURE 2 | Complexity. (A) Visualisation of the overarching principle for the production of molecular animations, connecting the four main elements audience, design, representation and motion. (B) Relationship between the level of detail of a macromolecule representation and the motion complexity for four relevant types of motion. The complexity of motion (on the right) is visualised by a waveform-like line where a higher frequency represents more frequent changes and therefore higher complexity. Representations, level of detail (left). Selected representations of a beta-sheet poly-peptide on a scale of increasing level of detail/complexity from the bottom to the top. Illustrative abstract representations: curved backbone, linear backbone and ribbon/cartoon (Richardson, 1985; Carson, 1987). Surface abstractions: beads and coarse approximation (Blinn, 1982). Atomistic surface representation: molecular skin surface (MSS), Gaussian surface, solvent accessible surface (SAS) (Sanner, et al., 1996) and solvent excluded surface (SES) (Connolly, 1983). Atomistic space-filling representation: space-fill (Corey and Pauling, 1953; Koltun, 1965a and; Koltun, 1965b). Atomistic bond-centric representations: licorice, ball-and-stick (Fieser, 1963).

of structural detail shown in a scientific visualisation or animation. However, that does not keep us from integrating data from all kinds of resources or even models, predictions or hypotheses (O'Donoghue et al., 2010b; Ward et al., 2013). Even with gaps in the data, it may still be useful to display the complete macromolecule and to model the structure gaps based on the best knowledge available. We always should consider full length, native proteins for a more realistic visualisation because it reflects the natural cellular environment. This use of artistic licence plays an important

role in scientific visualisation; see Goodsell and Jenkinson (2018) and Goodsell and Johnson (2007) for details on this topic. In the spirit of scientific authenticity - which is fundamental for the credibility of molecular animations - it is good practice to mark the use of artistic licence by differentiated representation, render style, colour and/or by annotation. This ensures transparency and therefore increases the accountability of an animation (Jantzen et al., 2015). The viewer should be able to judge, which aspects are data derived and which are more hypothetical.

5 DESIGN THE MOTION TO REFLECT MOLECULAR REALITY

The design of motion requires biophysical knowledge from molecule kinetics and kinematics. For biological macromolecules four types of motion are most relevant: thermal motion, small and large conformational changes (all intrinsic) and Brownian motion (with a molecule as one unit), see visualisations in **Figures 1E–H**. More specific cases are the two-dimensional movement of a protein in a membrane and directed motor activities like the kinesin or myosin transport activities. Please see Johnson and Hertig (2014) and O'Donoghue et al. (2010b) for detail and Phillips et al. (2012), Nelson (2014) and Kuriyan et al. (2012) for the biophysical principles. It is a massive challenge to reflect various molecular motion time scales that span 17 orders of magnitude (McGill, 2008), see **Figure 1I**. However, the amplitude of a movement is often proportional to its frequency, so very rapid movements can be left out when the complete protein is shown. They need to be considered though, when the intrinsic dynamic plays a role for the function (Eisenmesser et al., 2005).

5.1 Thermal Motion

Thermal motion is observed for any atom in a molecule and has a very high frequency, dependent on the temperature. The higher the temperature, the stronger the positional dislocation. It can often be neglected in an animation, especially when other, lower frequency motions are shown. It remains relevant in very complex and highly detailed animations, where visualising the movement of individual atoms or even quantum mechanical detail (orbitals) increases the accuracy of the representation.

5.2 Small Conformational Changes

Small conformational changes are based on the rotational freedom of a bond between two atoms, e.g., the rotation between peptide bonds (see Dong, 2021). Accumulated along a network of macromolecule residues, they can add up to far-reaching and larger movements. The rotational freedom can be restricted through non-covalent bonds like hydrogen-bonds, salt-bridges or hydrophobic interactions. Small conformational changes should be included in animations with high complexity and especially when they are central for the visualisation of the molecular mechanism and therefore the function of the macromolecule. Together with thermal motion, small conformational changes are responsible for the “breathing” of a protein (Makowski et al., 2008) which can be represented by a fluctuating surface.

5.3 Large Conformational Changes

Biological macromolecules and especially proteins often have domains, subdomains or other structural motifs (like alpha-helices and beta-sheets) that show a certain rigidity within themselves while flexible loops between them are responsible for movements of those substructures relative to each other. Flexible loops themselves also may undergo large conformational changes to fulfil a function. Many biological processes depend on this type of large conformational

changes. They are often visualised by the intrinsic movement of surface representations, but also ribbon-type cartoons.

5.4 Brownian Motion

When macromolecules in a cellular environment move as one unit, they usually do so by random collision with other molecules, caused by their thermal motion and often described as random walk. Collisions create an external force that is not directional, so the movement of the macromolecule is random and not caused by long-range attracting forces between two reaction partners. This provides a challenge for a molecular animator, who needs to find a balance between the visualisation of the non-directional nature of the random motion and the actual approach of reaction partners within the timeframe of the scene. Le Muzic et al. (2014) describe a way to blend random walk with linear interpolation in a particle based metabolic network model to simulate this motion.

5.5 Ensure Dynamic Realism

The motion-equivalent to structural detail in a representation is called dynamic realism by Jantzen (Biocinematics, 2016). Both, structural and dynamic information need to be considered in a molecular animation where the representation becomes unrealistic when it is inconsistent between structure and motion. Dynamic realism means that abstract, less detailed structure representations go along with simple dynamics and more detailed structure representations also require more realistic dynamic representations (Biocinematics, 2016). The complexity of motion is interpreted here as the changes of direction and acceleration, rather than those of the actual speed.

5.6 Reflect a Crowded Cellular Environment

In a crowded cellular environment, the set of principles described by Jantzen et al. (2017) should be respected. In short and partially merged: I. permanent Brownian motion causes collisions and therefore movement, there are no long-range forces; II. biological macromolecules underlay internal flexibility but they have defined boundaries; III. in the cell, there are many instances of a molecule and not all react; IV. the cell is a crowded environment that does not show aqueous effects. The representation of individual elements in a crowded environment however does not require the full detail of all elements. The further away an element from the main focus, the fewer atoms can be displayed without losing any major information (Le Muzic et al., 2014, 2015).

6 DISCUSSION

In systems of high dynamics such as the nano-world of biological macromolecules, the medium of video can play out its strength. Compared to explanatory text, an animation can often be more efficient and intuitive. Compared to a diagram or illustration it can be more accurate and detailed. Consequently, an animation used in research and education should reflect the comprehensive knowledge of a system. Only then the complexity of the system can be communicated realistically and used for the development and evaluation of hypotheses. Failure to reflect the complexity leads to the misconception that a complex system is indeed

simple and also to flawed future experiment design. As a consequence, it is recommended to use 2D sketches and cartoon style shading when little is known about a system and a 3D animation for established mechanisms (Iwasa, 2010).

6.1 Consistent Complexity

The main determinant for the complexity of a molecular animation is the complexity of the actual communication objective, the point that needs to come across, the focus of attention. The communication objective of a particular scene can be very specific and is usually related to one of these two categories (or a combination of both): a) the properties of the components - structure, chemical and physical properties, relation towards each other, etc.; and b) the dynamic of the system, the changes over time. The complexity of a communication objective is then directly associated to the complexity of the main element. The principle of consistent complexity (see **Figure 2A** for a visualisation) is proposed as overarching principle for the production of molecular animations. The communication objectives can be reached with clarity when consistent complexity is achieved for the relevant aspects.

6.1.1 Communication Objective and Abstraction/Simplification

A simple point is often made best with a simple rather than a complex visualisation, as the latter can distract or overwhelm the viewer. A complex point however usually requires a more complex visualisation because the detail is just not there in a simpler representation and the viewer is usually not able to interpret it on his/her own.

6.1.2 Audience Expertise and Scientific Complexity

An animation should aim towards advancing the audience but not overwhelm it. Hence, the scientific complexity of the visual story needs to be in balance with the audience's level of expertise and visual literacy. The theoretical framework for visual storytelling developed by Botsis et al. (2020) is helpful to evaluate the individual characteristics of a visual story. It goes back to Cairo's Visualization Wheel (Cairo, 2012) and comprises six contrasting pairs of characteristics: conceptualisation - figuration, functionality - decoration, density - lightness, multidimensionality - unidimensionality, originality - familiarity and novelty - redundancy. A consistent story represents the set of characteristics that are mentioned first in a pair (high complexity) or second (low complexity). Practically, a modular approach for the combination of scenes can help to target a specific audience.

6.1.3 Experiment and Representation

The representation detail should match the quality and resolution of the existing experimental basis. This helps to avoid the impression of more knowledge than there actually is. For scientific authenticity, the use of artistic licence should be transparently annotated but not avoided if it helps the representation of the realistic conditions.

6.1.4 Characters and Environment

Large differences in the representation complexity of neighbouring character levels (main, side, extras, environment) should be avoided.

This is an issue for mixed representations. A more gradual change of the level of detail helps to avoid visual breaks. The environment should have less complexity than the characters, but may well become the focus of attention for another communication objective and have its complexity increased for another scene.

6.1.5 Structure and Motion Representation

The complexity of the structure representation needs to be matched with the one of the motion representation (dynamic realism). While the complexity of a structure representation is easily understood as the level of detail, the complexity of a motion is less well intuitive. **Figure 2B** suggests a complexity scale for properties and their associated movements. The level of detail of a biomolecule representation does not necessarily correlate with the speed and amplitudes of the different types of motion.

We need to look at the different motion types in order to relate the complexity of a representation with the complexity of a motion. **Figure 2B** sets them into relation and gives an indication which type of motion should be shown in association with a certain representation detail. Thermal motion should be included in an animation when the communication objective focusses on the atomistic reaction detail. It can be neglected at protein (surface) level, where protein breathing should be included and thermal motion adds next to nothing to the accumulative motion. Large conformational changes are often at the heart of an animation and the centre of the communication objective. The inclusion in the animation is therefore a matter of course and relevant on the domain/subdomain level. Brownian Motion is relevant for the overall motion of molecules as a unit and enables reactions between molecules in the first place. It should therefore be included on that protein level. However, it needs to be mentioned that the relations described are a first indication only and that a specific communication objective may well require different combinations of representation and motion complexity.

AUTHOR CONTRIBUTIONS

EW is the sole contributor to this article. It is based on EW's Bachelor Thesis entitled "Science vs. Design - Principle strategies for the production of molecular animations", submitted to the design academie berlin (now SRH Berlin School of Design and Communication) (Werner, 2019) and the poster "Strategies for the Production of Molecular Animations" presented at VIZBI (Werner, 2021).

FUNDING

Only internal resources of RNS Berlin.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fbinf.2022.793914/full#supplementary-material>

REFERENCES

- Aristotle (1996). *Poetics*. London: Penguin Classics.
- Berlin, H. A. (2016). Communicating Science: Lessons from Film. *Trends Immunol.* 37 (4), 256–260. doi:10.1016/j.it.2016.02.006
- Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., et al. (2000). The Protein Data Bank. *Nucleic Acids Res.* 28 (1), 235–242. doi:10.1093/nar/28.1.235
- Bernard, S. C. (2010). *Documentary Storytelling: Creative Nonfiction on Screen*. Waltham: Focal Press.
- Berry, M. B., Bae, E., Bilderback, T. R., Glaser, M., and Phillips, G. N. (2006). Crystal Structure of Adp/amp Complex of Escherichia Coli Adenylate Kinase. *Proteins* 62 (2), 555–556. doi:10.1002/prot.20699
- Bienert, S., Waterhouse, A., de Beer, T. A., Tauriello, G., Studer, G., Bordoli, L., et al. (2017). The Swiss-Model Repository-New Features and Functionality. *Nucleic Acids Res.* 45 (D1), D313–D319. doi:10.1093/nar/gkw1132
- Biocinematics (2016). Molecular Visualization: Principles and Practice. Available at <https://www.youtube.com/watch?v=G5FxDpBMUHE> (Accessed May 31st, 2019).
- Blinn, J. F. (1982). A Generalization of Algebraic Surface Drawing. *ACM Trans. Graph.* 1 (3), 235–256. doi:10.1145/357306.357310
- Botsis, T., Fairman, J. E., Moran, M. B., and Anagnostou, V. (2020). Visual Storytelling Enhances Knowledge Dissemination in Biomedical Science. *J. Biomed. Inform.* 107, 103458. doi:10.1016/j.jbi.2020.103458
- Bromberg, S., Chiu, W., and Ferrin, T. E. (2010). Workshop on Molecular Animation. *Structure* 18 (10), 1261–1265. doi:10.1016/j.str.2010.09.001
- Cairo, A. (2012). *The Functional Art: An Introduction to Information Graphics and Visualization*. Berkeley: New Riders.
- Carson, M. (1987). Ribbon Models of Macromolecules. *J. Mol. Graphics* 5 (2), 103–106. doi:10.1016/0263-7855(87)80010-3
- Connolly, M. L. (1983). Solvent-Accessible Surfaces of Proteins and Nucleic Acids. *Science* 221 (4612), 709–713. doi:10.1126/science.6879170
- Corey, R. B., and Pauling, L. (1953). Molecular Models of Amino Acids, Peptides, and Proteins. *Rev. Scientific Instr.* 24 (8), 621–627. doi:10.1063/1.1770803
- Dahlstrom, M. F. (2014). Using Narratives and Storytelling to Communicate Science with Nonexpert Audiences. *Proc. Natl. Acad. Sci. U S A*. 111 Suppl 4 (Suppl. 4), 13614–13620. doi:10.1073/pnas.1320645111
- Dong, M. (2021). A Minireview on Temperature Dependent Protein Conformational Sampling. *Protein J.* 40 (4), 545–553. doi:10.1007/s10930-021-10012-x
- Eisenmesser, E. Z., Millet, O., Labeikovsky, W., Korzhnev, D. M., Wolf-Watz, M., Bosco, D. A., et al. (2005). Intrinsic Dynamics of an Enzyme Underlies Catalysis. *Nature* 438 (7064), 117–121. doi:10.1038/nature04105
- ElShafie, S. J. (2018). Making Science Meaningful for Broad Audiences through Stories. *Integr. Comp. Biol.* 58 (6), 1213–1223. doi:10.1093/icb/icy103
- Faelber, K., Posor, Y., Gao, S., Held, M., Roske, Y., Schulze, D., et al. (2011). Crystal Structure of Nucleotide-free Dynamin. *Nature* 477 (7366), 556–560. doi:10.1038/nature10369
- Fieser, L. F. (1963). Plastic Dreiding Models. *J. Chem. Educ.* 40 (9), 457–459. doi:10.1021/ed040p457
- Frankel, F. C., and DePace, A. H. (2012). *Visual Strategies - a Practical Guide to Graphics for Scientists and Engineers*. New Haven: Yale University Press.
- Gobeil, S. M., Janowska, K., McDowell, S., Mansouri, K., Parks, R., Manne, K., et al. (2021). D614g Mutation Alters Sars-Cov-2 Spike Conformation and Enhances Protease Cleavage at the S1/S2 Junction. *Cell Rep* 34 (2), 108630. doi:10.1016/j.celrep.2020.108630
- Goddard, T. D., and Ferrin, T. E. (2007). Visualization Software for Molecular Assemblies. *Curr. Opin. Struct. Biol.* 17 (5), 587–595. doi:10.1016/j.sbi.2007.06.008
- Goodsell, D. S., and Jenkinson, J. (2018). Molecular Illustration in Research and Education: Past, Present, and Future. *J. Mol. Biol.* 430 (21), 3969–3981. doi:10.1016/j.jmb.2018.04.043
- Goodsell, D. S., and Johnson, G. T. (2007). Filling in the Gaps: Artistic License in Education and Outreach. *Plos Biol.* 5 (12), e308. doi:10.1371/journal.pbio.0050308
- Goodsell, D. S., Olson, A. J., and Forli, S. (2020). Art and Science of the Cellular Mesoscale. *Trends Biochem. Sci.* 45 (6), 472–483. doi:10.1016/j.tibs.2020.02.010
- Iwasa, J. H. (2010). Animating the Model Figure. *Trends Cel Biol* 20 (12), 699–704. doi:10.1016/j.tcb.2010.08.005
- Iwasa, J. H. (2015). Bringing Macromolecular Machinery to Life Using 3d Animation. *Curr. Opin. Struct. Biol.* 31, 84–88. doi:10.1016/j.sbi.2015.03.015
- Jantzen, S. G., Jenkinson, J., and McGill, G. (2015). Transparency in Film: Increasing Credibility of Scientific Animation Using Citation. *Nat. Methods* 12 (4), 293–297. doi:10.1038/nmeth.3334
- Jantzen, S., McGill, G., and Jenkinson, J. (2017). Molecular Visualization Principles. Available at <https://bmcresearch.utm.utoronto.ca/sciencevislab/index.php/portfolio/molecular-visualization-principles/> (Accessed Oct 10th, 2021).
- Johnson, G. T., Autin, L., Goodsell, D. S., Sanner, M. F., and Olson, A. J. (2011). Epmv Embeds Molecular Modeling into Professional Animation Software Environments. *Structure* 19 (3), 293–303. doi:10.1016/j.str.2010.12.023
- Johnson, G. T., and Hertig, S. (2014). A Guide to the Visual Analysis and Communication of Biomolecular Structural Data. *Nat. Rev. Mol. Cel Biol.* 15 (10), 690–698. doi:10.1038/nrm3874
- Koltun, W. L. (1965b). Precision Space-Filling Atomic Models. *Biopolymers* 3 (6), 665–679. doi:10.1002/bip.360030606
- Koltun, W. L. (1965a). *Space Filling Atomic Units and Connectors for Molecular Models*. U.S. Patent No US-3170246-A. Washington, DC: U.S. Patent and Trademark Office.
- Kozliková, B., Krone, M., Falk, M., Lindow, N., Baaden, M., Baum, D., et al. (2017). Visualization of Biomolecular Structures: State of the Art Revisited. *Comput. Graphics Forum* 36 (8), 178–204. doi:10.1111/cgf.13072
- Kuriyan, J., Konforti, B., and Wemmer, D. (2012). *The Molecules of Life: Physical and Chemical Principles*. New York: Garland Science Taylor & Francis Group.
- Le Muzic, M., Autin, L., Parulek, J., and Viola, I. (20152015). Cellview: A Tool for Illustrative and Multi-Scale Rendering of Large Biomolecular Datasets. *Eurographics Workshop Vis. Comput. Biomed.* 2015, 61–70. doi:10.2312/vcbm.20151209
- Le Muzic, M., Parulek, J., Stavrum, A. K., and Viola, I. (2014). Illustrative Visualization of Molecular Reactions Using Omniscient Intelligence and Passive Agents. *Comput. Graphics Forum* 33 (3), 141–150. doi:10.1111/cgf.12370
- Lepito, A. (2018). Where Animation and Science Meet. *Integr. Comp. Biol.* 58 (6), 1279–1282. doi:10.1093/icb/icy074
- Ma, K. L., Liao, I., Frazier, J., Hauser, H., and Kostis, H. N. (2012). Scientific Storytelling Using Visualization. *IEEE Comput. Graph. Appl.* 32 (1), 12–19. doi:10.1109/MCG.2012.24
- Mahtarin, R., Islam, S., Islam, M. J., Ullah, M. O., Ali, M. A., and Halim, M. A. (2020). Structure and Dynamics of Membrane Protein in Sars-Cov-2. *J. Biomol. Struct. Dyn.*, 1–14. (online ahead of print). doi:10.1080/07391102.2020.1861983
- Makowski, L., Rodi, D. J., Mandava, S., Minh, D. D., Gore, D. B., and Fischetti, R. F. (2008). Molecular Crowding Inhibits Intramolecular Breathing Motions in Proteins. *J. Mol. Biol.* 375 (2), 529–546. doi:10.1016/j.jmb.2007.07.075
- Mandala, V. S., McKay, M. J., Shcherbakov, A. A., Dregni, A. J., Kolocouris, A., and Hong, M. (2020). Structure and Drug Binding of the Sars-Cov-2 Envelope Protein Transmembrane Domain in Lipid Bilayers. *Nat. Struct. Mol. Biol.* 27 (12), 1202–1208. doi:10.1038/s41594-020-00536-8
- Martinez, X., Krone, M., Alharbi, N., Rose, A. S., Laramée, R. S., O'Donoghue, S., et al. (2019). Molecular Graphics: Bridging Structural Biologists and Computer Scientists. *Structure* 27 (11), 1617–1623. doi:10.1016/j.str.2019.09.001
- material.io/Google (2021). Understanding Motion. Available at <https://material.io/design/motion/> (Accessed Oct 10th, 2021).
- McGill, G. (2008). Molecular movies. Coming to a Lecture Near You. *Cell* 133 (7), 1127–1132. doi:10.1016/j.cell.2008.06.013
- McGill, G. (2010). Molecular Maya: Adapting Hollywood's Tools for Biovisualization. Available at <http://plato.cgl.ucsf.edu/Workshops/AnimationWorkshop2010/Videos/1-GaelMcGill.mov> (Accessed May 31st, 2019).
- Mercado, G. (2010). *The Filmmaker's Eye: Learning (And Breaking) the Rules of Cinematic Composition*. London: Taylor & Francis.
- Müller, C. W., Schlauderer, G. J., Reinstein, J., and Schulz, G. E. (1996). Adenylate Kinase Motions during Catalysis: An Energetic Counterweight Balancing Substrate Binding. *Structure* 4 (2), 147–156. doi:10.1016/s0969-2126(96)00018-4

- Munzner, T. (2014). *Visualization Analysis and Design: Principles, Techniques, and Practice*. London: Taylor & Francis.
- Nelson, P. (2014). *Biological Physics. Energy, Information, Life*. New York: W. H. Freeman.
- O'Donoghue, S. I., Gavin, A. C., Gehlenborg, N., Goodsell, D. S., Hériché, J. K., Nielsen, C. B., et al. (2010a). Visualizing Biological Data-Now and in the Future. *Nat. Methods* 7 (3Suppl. 1), S2–S4. doi:10.1038/nmeth.f.301
- O'Donoghue, S. I., Goodsell, D. S., Frangakis, A. S., Jossinet, F., Laskowski, R. A., Nilges, M., et al. (2010b). Visualization of Macromolecular Structures. *Nat. Methods* 7 (3Suppl. 1), S42–S55. doi:10.1038/nmeth.1427
- Offerdahl, E. G., Arneson, J. B., and Byrne, N. (2018). Lighten the Load: Scaffolding Visual Literacy in Biochemistry and Molecular Biology. *CBE Life Sci. Educ.* 16 (19), 1–11. doi:10.1187/cbe.16-06-0193
- Olson, A. J. (2018). Perspectives on Structural Molecular Biology Visualization: From Past to Present. *J. Mol. Biol.* 430 (21), 3997–4012. doi:10.1016/j.jmb.2018.07.009
- Phillips, R., Kondev, J., Theriot, J., and Garcia, H. (2012). *Physical Biology of the Cell*. New York: Garland Science Taylor & Francis Group.
- Raschke, H. (2013). *Szenische Auflösung. Wie Man Sich Eine Filmszene Erarbeitet (Praxis Film)*. Konstanz: UVK Verlagsgesellschaft.
- Richardson, J. S. (1985). Schematic Drawings of Protein Structures. *Methods Enzymol.* 115, 359–380. doi:10.1016/0076-6879(85)15026-3
- Sanner, M. F., Olson, A. J., and Spehner, J. C. (1996). Reduced Surface: An Efficient Way to Compute Molecular Surfaces. *Biopolymers* 38 (3), 305–320. doi:10.1002/(SICI)1097-0282(199603)38:3%3C305:AID-BIP4%3E3.0.CO;2-Y
- Segel, E., and Heer, J. (2010). Narrative Visualization: Telling Stories with Data. *IEEE Trans. Vis. Comput. Graph.* 16 (6), 1139–1148. doi:10.1109/TVCG.2010.179
- Sharpe, J., Lumsden, C. J., and Woolridge, N. (2008). *Silico : 3d Animation and Simulation of Cell Biology with Maya and Mel*. Amsterdam: Elsevier.
- Shneiderman, B. (1996). “The Eyes Have it: A Task by Data Type Taxonomy for Information Visualizations,” in Proc. 1996 IEEE Symp on Visual Languages, Boulder, CO, USA, 336–343.
- Sijll, J. V. (2005). *Cinematic Storytelling: The 100 Most Powerful Film Conventions Every Filmmaker Must Know*. Burbank: Michael Wiese Productions.
- TED (2012). Drew Berry: Animations of Unseeable Biology. Available at <https://www.youtube.com/watch?v=WFCvkkDSfIU> (Accessed Feb 24th, 2022).
- Thomas, F., and Johnston, O. (1981). *Disney Animation: The Illusion of Life*. Glendale: Disney Editions.
- VIZBI (2012a). Gaël McGill: Visualizing Protein Dynamics. Available at <https://vimeo.com/26199717> (Accessed Jun 2nd, 2019).
- VIZBI (2012b). Graham Johnson: Mesoscale Visualization. Available at <https://vimeo.com/43094587> (Accessed Jun 26th, 2019).
- Vogler, C. (2007). *The Writer's Journey: Mythic Structure for Writers*. Burbank: Michael Wiese Productions.
- Wäger, M. (2014). *Grafik Und Gestaltung: Das Umfassende Handbuch*. Bonn: Rheinwerk Design.
- Ward, A. B., Sali, A., and Wilson, I. A. (2013). Biochemistry. Integrative Structural Biology. *Science* 339 (6122), 913–915. doi:10.1126/science.1228565
- Ware, C. (2012). *Information Visualization: Perception for Design (Interactive Technologies)*. Burlington: Morgan Kaufmann.
- Werner, E., Ziegler, M., Lerner, F., Schweiger, M., and Heinemann, U. (2002). Crystal Structure of Human Nicotinamide Mononucleotide Adenylyltransferase in Complex with Nmn. *FEBS Lett.* 516 (1–3), 239–244. doi:10.1016/s0014-5793(02)02556-5
- Werner, E. (2019). Science vs. Design, Principle Strategies for the Production of Molecular Animations. B. A. thesis. Berlin: design akademie berlin, SHR Hochschule für Kommunikation und Design.
- Werner, E. (2021). Strategies for the Production of Molecular Animations. Available at <https://vizbi.org/Posters/2021/vB17> (Accessed Mar 25th, 2021).
- Wohlfart, M., and Hauser, H. (2007). “Story Telling for Presentation in Volume Visualization,” in Eurographics/IEEE-VGTC Symposium on Visualization, Norrköping, Sweden, 91–98.

Conflict of Interest: The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Werner. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



SingleCAnalyzer: Interactive Analysis of Single Cell RNA-Seq Data on the Cloud

Carlos Prieto*, David Barrios and Angela Villaverde

Bioinformatics Service, Nucleus, University of Salamanca, Salamanca, Spain

Single-cell RNA sequencing (scRNA-Seq) enables researchers to quantify the transcriptomes of individual cells. The capacity of researchers to perform this type of analysis has allowed researchers to undertake new scientific goals. The usefulness of scRNA-Seq has depended on the development of new computational biology methods, which have been designed to meeting challenges associated with scRNA-Seq analysis. However, the proper application of these computational methods requires extensive bioinformatics expertise. Otherwise, it is often difficult to obtain reliable and reproducible results. We have developed SingleCAnalyzer, a cloud platform that provides a means to perform full scRNA-Seq analysis from FASTQ within an easy-to-use and self-exploratory web interface. Its analysis pipeline includes the demultiplexing and alignment of FASTQ files, read trimming, sample quality control, feature selection, empty droplets detection, dimensional reduction, cellular type prediction, unsupervised clustering of cells, pseudotime/trajectory analysis, expression comparisons between groups, functional enrichment of differentially expressed genes and gene set expression analysis. Results are presented with interactive graphs, which provide exploratory and analytical features. SingleCAnalyzer is freely available at <https://singleCAnalyzer.eu>.

Keywords: ScRNA-seq, data visualization, single cell, web server, data analysis

OPEN ACCESS

Edited by:

Yann Ponty,
École Polytechnique, France

Reviewed by:

Lionel Spinelli,
Aix-Marseille Université, France
Sebastian Will,
École Polytechnique, France

*Correspondence:

Carlos Prieto
cprietos@usal.es

Specialty section:

This article was submitted to
Data Visualization,
a section of the journal
Frontiers in Bioinformatics

Received: 11 October 2021

Accepted: 09 May 2022

Published: 23 May 2022

Citation:

Prieto C, Barrios D and Villaverde A
(2022) SingleCAnalyzer: Interactive
Analysis of Single Cell RNA-Seq Data
on the Cloud.
Front. Bioinform. 2:793309.
doi: 10.3389/fbinf.2022.793309

INTRODUCTION

Single-cell RNA sequencing (scRNA-seq) has allowed for the quantification of RNA transcripts within individual cells. These assays allow researchers to explore cell-to-cell variability and meet new scientific goals. In the last few years, scRNA-seq has been applied, for example, to differentiate tumor cells from healthy ones, deconvolute immune cells, describe states of cell differentiation and development, and to identify rare populations of cells that cause disease (Haque et al., 2017). Although experimental scRNA-seq assays are becoming increasingly user-friendly, the analysis of sequencing data is complex. Data analysis requires the application of complex computational pipelines and data analysis methods that require bioinformatics expertise (Hwang et al., 2018). The interpretation of scRNA-seq results is strongly influenced by its analysis pipeline, and the incorrect application of methods could lead to conclusions that are incorrect. Since data analysis is complex and very important for correctly interpreting results, the development of analysis tools that produce reliable results and minimize the possibility of error is essential for enhancing the usefulness of scRNA-seq data.

Throughout the last 5 years, some software development projects have aimed to address the absence of software available for the analysis of scRNA-seq data (Guo et al., 2015; Gardeux et al.,

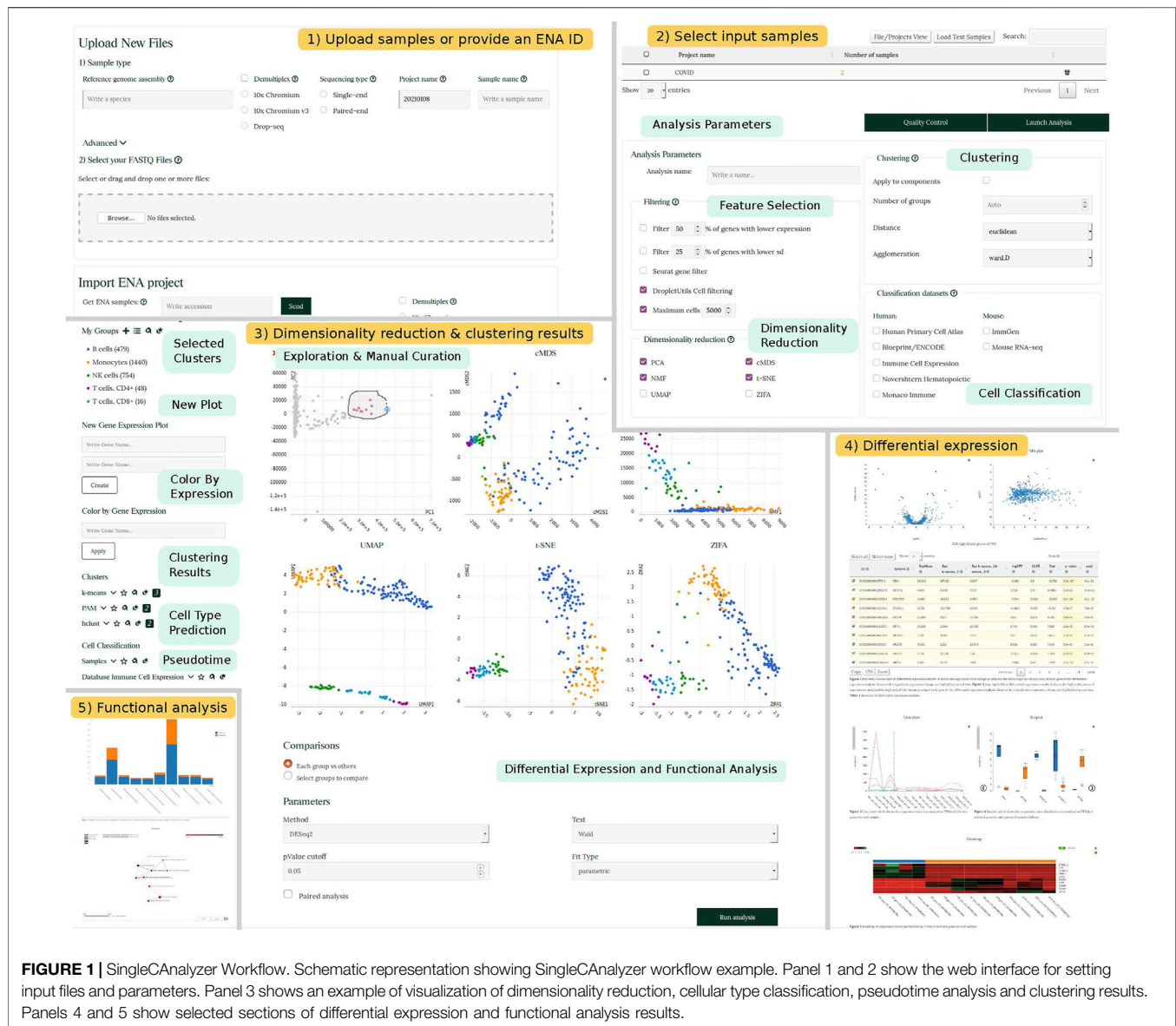


FIGURE 1 | SingleCAnalyzer Workflow. Schematic representation showing SingleCAnalyzer workflow example. Panel 1 and 2 show the web interface for setting input files and parameters. Panel 3 shows an example of visualization of dimensionality reduction, cellular type classification, pseudotime analysis and clustering results. Panels 4 and 5 show selected sections of differential expression and functional analysis results.

2017; Kiselev et al., 2017; Lin et al., 2017; Perraudeau et al., 2017; Zhu et al., 2017; Scholz et al., 2018; Wagner and Yanai, 2018; Chen et al., 2019; Monier et al., 2019; Stuart et al., 2019). Designers of the projects have developed analysis pipelines that can be executed with R or Python function calls or with websites. Although the platforms have tremendous utility, they do possess some usability and functionality limitations that should be solved. For example, none of the applications are capable of analysing raw sequencing files (FASTQ), they do not allow for the interactive selection of groups and a few provide an integrated functional analysis of results (see **Supplementary Table S1**).

We have developed SingleCAnalyzer to provide a Web application server that performs a fully interactive and comprehensive analysis of scRNA-Seq data with two simple steps. It provides an integrated and interactive platform which is able to process sequencing files (FASTQ) and perform full

scRNA-seq analyses and the functional analysis of results. It was implemented as a cloud analysis platform that can be executed without installing any software. SingleCAnalyzer facilitates the analysis of scRNA-seq data to non-experienced users and provides quick exploratory analyses to computational biologists.

RESULTS

The SingleCAnalyzer Website

The front-end of SingleCAnalyzer has been designed to provide a means to fully analyse scRNA-Seq data using the following two steps: 1) Setting input files and analysis parameters and 2) cluster determination and the execution of comparative analysis. In the first step, FASTQ/HDF5 files are uploaded or an ENA project identifier is provided by the user. Basic information regarding the

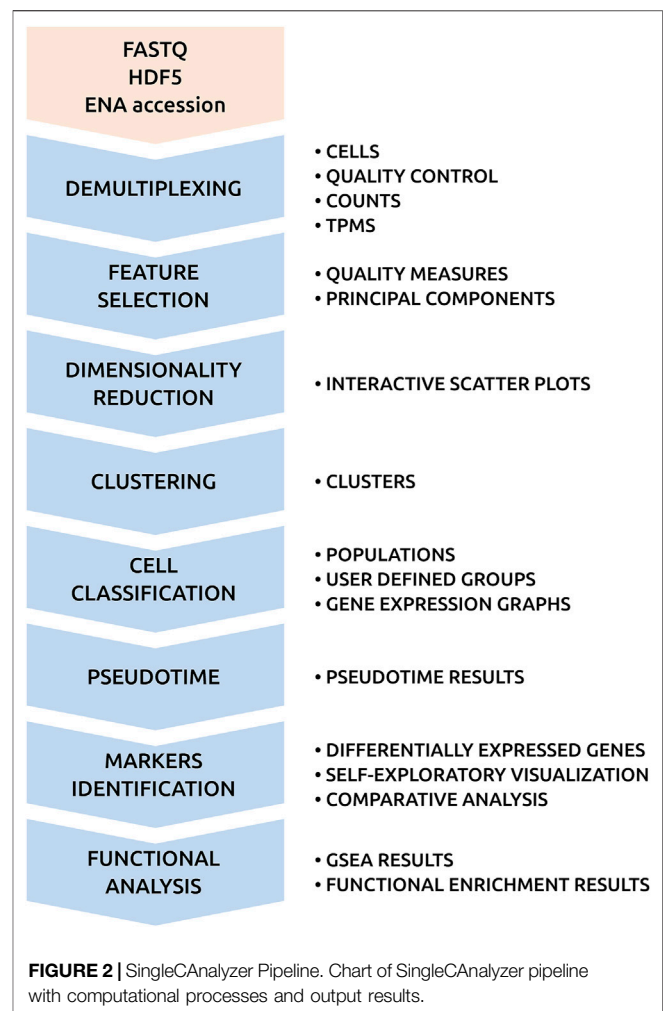
species studied and type of sequencing performed, as well as optional parameters for the alignments of sequences can also be specified on the web. Once the files are uploaded, demultiplexed and aligned, users may perform further analysis including feature selection, empty droplet deletion, dimensional reduction, prediction of cellular type, analysis of trajectories/pseudotime and unsupervised clustering. These analyses can be performed and adjusted by selecting parameters in the ‘analysis parameters’ section (**Figure 1**).

Cluster determination and the execution of comparative analysis is accomplished through the website, which provides an interactive interface that allows the user to visualise cellular type prediction, pseudotime predictions or clustering results via six interconnected scatterplots generated using each dimensionality reduction technique. Point colour and type can be changed according to each analysis results. Users can also generate new representations of gene pairs and colour the points based on gene expression values. This interface specifies the most adequate aggrupation, cellular classification or time frame and is guided by the user’s knowledge regarding the samples studied. On the interface, the user can also launch a comparative analysis of all groups, or manually determine which groups should be compared. The comparative analysis includes an analysis of differentially expressed groups of genes, and the functional analysis of gene ontology categories and pathways.

Results are displayed in tabular form, which reveal the execution status of each computational process and provide a link to final results. These are provided as static reports and interactive web pages. Results regarding the quantification of gene expression values are provided with a table of quantification statistics and downloadable files that contain information for aligned reads regarding the number of reads generated per transcript and the number of transcripts per million (TPM). The quality control page descriptively reveals the distribution patterns of expression using box plots, reveals estimated numbers of expressed genes using a bar plot and represents the first two components of a PCA analysis. The clustering results page integrates dimensionality reduction, clustering, pseudotime and cellular classification results within self-explanatory interface which can also generate static reports that incorporate user modifications and launch comparisons between groups. Reports containing results are generated for each comparison, which include differential expression, functional enrichment and GSEA analysis. Differential expression results are summarised in a table which is linked to the following means to visualise data: MA plot, volcano plot, box plot, line chart and heatmap. Functional analyses are also summarised in tables and interactive visual means to represent data such as bar plots, networks and symmetric heatmaps are provided.

Supplementary Table S1 shows a comparison with 12 scRNA-Seq analysis platforms. The main features of the SingleCAnalyzer website are:

- scRNA-Seq analysis from raw FASTQ, HDF5 files or ENA project identifications



- Fully functional cloud platform that does not require the installation of software
- Semiautomated analysis which avoids the need for configuration using complex parameters
- User guided classification of cells within groups that is guided by interconnected graphs that integrate dimensional reduction, cellular type prediction, trajectory analysis and unsupervised clustering results
- Performs FASTQ processing, gene filtering, empty droplets detection, gene quantification, dimensionality reduction, unsupervised clustering, differential expression, functional overrepresentation and gene set expression analyses
- Straightforward presentation of results using interactive visual representations of data and provides a means to generate reports that are publication ready

Analysis Pipeline

Figure 2 shows the analysis pipeline of SingleCAnalyzer. It integrates generally accepted tools used for the analysis of RNA-Seq data, which also perform well as computational resources. **Supplementary Table S2** shows the computational time required to analyse nine scRNA-Seq public data sets. The

complete analysis of 154 demultiplexed samples takes an average of 36 min, which allows for the real time analysis of low cell number scRNA-Seq experiments. The most time-consuming processes in the pipeline involves the upload, demultiplexing and alignment of samples, which are tasks that are performed in parallel. This parallelisation reduces the global analysis time by 57%, which makes the time requirement of our cloud infrastructure equal to virtual machine or local pipelines solutions. Moreover, SingleCAnalyzer does not store raw sequences or aligned files in order to avoid user disk space limitations, and the number of analysed samples of non-commercial cloud platforms.

The next steps of the pipeline include feature selection, empty droplets detection, dimensionality reduction, cellular type prediction, trajectory/pseudotime analysis and unsupervised clustering. SingleCAnalyzer applies gene filtering, which is based on user input parameters to avoid non-informative output, noise or drop out events. Afterward, six dimensionality reduction methods are applied to the data and samples are visualised using interactive scatter plots. Simultaneously, four unsupervised clustering algorithms are applied to produce nine possible clustering divisions for each method, a cellular type prediction method is executed for each training dataset, and a pseudotime analysis is performed (see methods). These cluster types can be mapped on interactive plots at the request of the user.

Based on the unsupervised or manually curated clusters produced, users can identify gene characteristics and the functions of each group by launching comparison analysis. This feature incorporates the differential expression analysis of groups and the functional analysis of gene ontologies and pathways. The analysis pipeline also processes quality control, clustering, differential expression and functional analysis results, and integrates them in an interactive and self-explanatory web interface.

SingleCAnalyzer was conceived as an agile project, and new scRNA-Seq analysis methods can be integrated within its analysis pipeline. Only generally accepted methods that have been demonstrated to generate reliable and reproducible results that require reasonable quantities of computational resources will be considered for addition to our cloud platform. The increasing development of computational methods will inspire the adaptation of the platform to meet the needs of researchers as scientific trends regarding scRNA-SEQ data analysis emerge.

Interactive Visualization

Visualization is a key aspect on the interpretation of scRNA-Seq results (Cakir et al., 2020). Analysis pipelines performs scatter plots for the representation of dimensional reduction results where point colors represent clusters, cell types, gene expression or trajectory features of each cell (Kiselev et al., 2017; Lin et al., 2017; Stuart et al., 2019). These plots are adequate for publishing results, but not for explorative analyses. At present, new technologies based on JavaScript enable the generation of interactive graphs in a Web User Interface. They allow the connection between graphs and the use of HTML5 components which could control visualization aspects. SingleCAnalyzer adopts this technology to visualize

information, interconnect graphs, show meta-information, calculate descriptive statistics, generate new graphs under user request and change the representation features interactively. SingleCAnalyzer includes six different graphical representations such as scatter plot, bar chart, heatmap, network, boxplot and density plot. These graphs are interactive, and the user can modify them by clicking on tables, html controls or other graphs.

The central result page is the representation of dimensional reduction and clustering of cells. It is composed by scatter plots where points represent cells, and the user can select the color and shape of points manually or by using clustering, cell population, pseudotime or gene expression results. The user can also explore group frequencies and define resulting groups based on meta-information or cell disposition on the graph. All graphs are interconnected, changes on graphical attributes or cell selections are synchronized on all displays. Cells can be located in all the graphs with a selection over one graph or by means of the locate samples menu. The application also allows the generation of new scatter plots which represent the expression of two genes in each cell.

Once the user defines the groups, he can launch a comparative expression analysis which results in two types of interactive reports. One is the differential expression report which are composed of interactive scatterplots, a boxplot, a line plot and a heatmap. All these graphs show information on mouse action and are connected with the table which summarizes the statistical analysis. They enable the comprehensive exploration of results and the query of information about expression changes of genes. The other report is the functional analysis which includes self-explanatory graphs such as bar plots, networks of terms and triangular heatmaps. Networks and heatmaps represent relations between gene sets which helps in the identification of related gene functions or pathways, while the bar plot shows the number of observed versus expected genes in each category.

Visualization features of SingleCAnalyzer enable the exploration and interpretation of results in an integrated platform which covers the main steps of scRNA-Seq analysis. The platform was presented and discussed at the VIZBI21 conference, where some improvements were suggested by attendants (VIZBI, 2022). Suggestions were focused on improving the usability of the platform and the adaptation of the analysis pipeline for their objectives. For example, an attendant required an adaptation for the analysis of RNA-Seq data which was developed and can be executed disabling the multiplexing process. SingleCAnalyzer is also distributed as a Docker machine and our graphical functions will be made public as R packages for its open use in analysis pipelines. All the representations performed with SingleCAnalyzer can be downloaded as graphical files ready for its inclusion in publications and analysis reports.

MATERIAL AND METHODS

Implementation

The SingleCAnalyzer website runs using LAMP architecture (Linux, Apache, MySQL and PHP). The front-end of the

website was developed using PHP, HTML5, JavaScript, D3, JQuery, AJAX and CSS3. Its implementation was based on the RaNA-Seq project, which contains similar alignment, differential expression and functional analysis tools (Prieto and Barrios, 2019). The analysis pipeline can be executed by a task manager that runs the analysis processes using R, Python or Linux Bash Shell. It also balances the computational load on our high-performance computing cluster. The analysis pipeline integrates cutting-edge tools which rapidly and reliably analyse scRNA-Seq data. **Figure 2** shows a flowchart of the pipeline used. We have optimised the analysis processes in our pipeline by harnessing computational clustering. Most of the tasks of analysis can be executed in real time. This optimisation has facilitated the development of an open and free cloud-based system.

FASTQ Processing

Raw sequence files in FASTQ format can be demultiplexed with Alevin software (version 1.3.0) (Srivastava et al., 2019) or pre-processed using the Fastp tool (version 0.19.4) (Chen et al., 2018). Gene expression quantification of genes in the selected reference genome is performed using Alevin or Salmon software (Patro et al., 2017). The platform can be used to assess data generated from any organism. At present, we have downloaded the most popular genomes from Ensembl (164 genomes) and have incorporated their transcriptome indexes within our server (Cunningham et al., 2019). Quality control of samples is performed based on the alignment summary, descriptive statistics and the Alevin report of demultiplexed samples non-supervised clustering performed using AlevinQC package (version 1.4.0).

Gene Filtering

Gene filters based on the quantification of gene expression, which reduce the noise and computational costs are available on SingleCAnalyzer. The current version can filter genes with the lowest levels of expression or standard deviations. We have also integrated the function 'FindVariableFeatures' within the Seurat package (version 3.2.2), which can identify variably genes by considering the strong relationship between variability and expression level (Stuart et al., 2019). Moreover, the user can also perform further dimensionality reduction and clustering processes by analysing the principal components obtained via principal component analysis (PCA). The optimum number of components used for the analyses can be determined using the *calc_npc* function of the CIDR package (version 0.1.5) (Lin et al., 2017). Empty droplets can be detected and removed with the application of the DropletUtils tool (version 1.8.0) (Lun et al., 2019).

Dimensionality Reduction

Interactive visualisation of samples in scatter plots requires a dimensionality reduction process, which is performed using the following methods: 1) PCA, which is generated with the *prcomp* function of the *stats* R package (version 4.0.3); 2) Classic multidimensional scaling (cMDS), which is performed with the *cmdscale* function of the *stats* R package using camberra as distance method; 3) Nonmetric multidimensional scaling

(isoMDS), which is performed using the *isoMDS* function of the *MASS* R package (version 7.3); 4) t-distributed stochastic neighbor embedding (t-SNE), which is performed using the *Rtsne* function of the *Rtsne* R package (version 0.15); 5) Uniform manifold approximation and projection (UMAP), which is performed using the *umap* function of the *uwot* R package (version 0.1.9); 6) and Non-negative matrix factorisation (NMF), which is performed using the *nnmf* function of the *NNLM* package (version 0.4.3). Collectively, application of these methods provides users with a multi-perspective assessment of the relationships between data.

Unsupervised Clustering

Determination of clusters within the interactive web interface is supported by the results provided by unsupervised clustering methods. At present, SingleCAnalyzer applies the following unsupervised clustering methods: 1) k-means, which is computed using the *kmeans* function of the *stats* R package (with *iter_max* = 15); 2) partition around medoids (PAM), which is computed using the *pam* function of the *cluster* R package (version 2.1.0); 3) hierarchical clustering, which is performed using the *hclust* function of the *stats* R package; 4) leiden clustering and pseudotime analysis, which is performed using Monocle3 R package (version 0.2.3) (Qiu et al., 2017). The user can specify input parameters such as the desired number of groups, the distance metric used by pam and hclust functions and the agglomeration parameter of hclust.

Pseudotime Analysis

Trajectory and pseudotime analyses are performed using the Monocle3 R package (Qiu et al., 2017). It calculates possible trajectories between leiden clusters over the UMAP projection. The pipeline calculates the pseudotime prediction for each cluster centroid and a scale colour which represent the time is applied over the points when an origin cluster is selected. The function *preprocess_cds* uses PCA or LSI output based on user options with the following parameters: *norm_method* = log and *scaling* = true. The function *reduce_dimension* uses the following parameters: *max_components* = 2, *reduction_method* = UMAP, *umap.metric* = cosine, *umap.min_dist* = 0.1, *umap.n_neighbors* = 15L, *umap.nn_method* = annoy. The function *cluster_cells* uses the following parameters: *k* = 20, *cluster_method* = Leiden, *num_iter* = 2, *partition_qval* = 0.05. The function *learn_graph* uses *use_partition* and *close_loop* as true.

Comparison Between Clusters

Groups of samples can be compared by applying different methods to assess differential expression. Reviews of the use of methods have concluded that no single method outperforms the others under all circumstances, and suggest that it is necessary to determine the optimal method or pipeline for each analysis performed (Seyednasrollah et al., 2013; Sonesson and Delorenzi, 2013). However, researchers have acknowledged that DESeq2 (version 1.28.1) (Love et al., 2014), EdgeR (version 3.30.3) (Robinson et al., 2010) and limma (version 3.44.3) (Law et al., 2014) are the most widely used methods

and consistently performed well when their reliability was assessed. We have integrated all of the methods within a SingleCAnalyzer that can be adjusted to apply customised parameters to individual tests.

SingleCAnalyzer performs a functional enrichment analysis and a gene set enrichment analysis (GSEA) for each comparison result. The enrichment analysis is performed with the R package Goseq (version 1.40) (Young et al., 2010) and the GSEA is performed with the R package fgsea (version 1.14) (Korotkevich et al., 2016). Functional annotation database used by these methods was downloaded from the NCBI BioSystems repository (Geer et al., 2009). Resulting graphs are generated with the package RJSplot (version 2.6) (Barrios and Prieto, 2018).

Data Management

Analyses can be launched as anonymous or registered users. Anonymous accounts are regularly deleted, and registered users can require the cancellation of their account. Data of registered users are protected by their personal password which is encrypted on our system. Users can freely download or delete their processed data and analysis results without any limitation. Raw data files uploaded by users (FASTQ, HDF5) are deleted once they are processed. This deletion avoids storage limitations and the presence of sequences in our system.

DISCUSSION

Single-cell platforms provide computational methods which enable the transformation of sequences into expression values of genes in each cell (Zheng et al., 2017; Shum et al., 2019). Further steps can be performed by the application of bioinformatics methods which are available on code repositories or analysis servers. These methods are connected in series to compose an analysis workflow. The development of pipelines is a complex work which involves the installation, test, setting up and integration of computational methods. In addition, full processing of scRNA-Seq data requires an intensive computational processing and the knowledge of programming languages for the execution of the pipeline. On the other hand, cloud servers are designed to avoid the development and execution of pipelines by the analysts, but its use also implies limitations such as additional data uploading time, uncertain server loads and limited customization of the analysis. Previous works have provided web servers for the analysis of scRNA-Seq data from a matrix with gene counts of cells (Gardeux et al., 2017; Zhu et al., 2017; Scholz et al., 2018; Chen et al., 2019; Monier et al., 2019). In this work we have developed the first cloud server which allow a complete analysis from sequences to pathways in a fully integrated platform. It was possible with the integration of low computational cost methods for the demultiplexing and quantification of reads which supports Drop-seq and 10x Chromium single-cell protocols (Srivastava et al., 2019).

Another approach for the analysis of scRNASeq sequences is the use of workflow management systems. A popular option is Galaxy which offers a web-based system for the pipeline

construction and the execution of bioinformatic analyses (Jalili et al., 2021). A recent study has presented Galaxy workflows for the analysis of scRNASeq data (Moreno et al., 2021). One of the workflows allows the uploading of FASTQ files for processing into an annotated cell matrix with Alevin. Then, post processing is done with Scanpy (Wolf et al., 2018) and the interactive visualization with the UCSC CellBrowser (Speir et al., 2021). This workflow has similar limitations to cloud solutions, as customization and uploading time, and requires of a computational cluster account and training about Galaxy workflows. Regarding the integration of results, the application of standard visualization tools avoids the creation of custom interfaces which integrate different nature of results, and the execution of new analysis based on the user interaction with the graph cannot be performed.

Visualization is a key aspect on the interpretation of scRNA-Seq results (Cakir et al., 2020). An adequate and interactive representation facilitates the correct classification and characterization of cells. This issue has been extensively approached by analysis techniques of cytometry and visualization methods have been adapted to the specific characteristics of single-cell such as the lower number of cells and the increment on the number of variables (transcripts/proteins). Two dimensional plots have been traditionally used for the representation of fluorescent makers on Cytometry. At present, flow cytometry panels can include dozens of makers and its representation as scatterplots are performed by a dimensional reduction technique. Similar strategy is followed for single cell visualization, but the lower number of cells allows its representation with web-based technologies which avoids software installation and platform dependencies. SingleCAnalyzer has developed its graphical interface with D3 and JavaScript technologies which allows the user-graph interaction on a Web browser. This solution has efficiently tested for the representation of 6,000 cells on six simultaneous scatterplots and allows a full interaction with clustering, cell classification, transcript quantification and cell trajectory results. Regarding the differential expression interface, it can handle 60,000 transcripts and perform six interconnected representations (MA-plot, volcano plot, scatterplot, boxplot and heatmap) on user interaction. The scalability of the platform will depend on the optimization of Web Browsers in the storage, representation and processing of interactive HTML Canvas and Scalable Vector Graphics. Current browsers have memory management and multiprocessing limitations. However, these technologies are becoming popular, and browsers are adapting their rendering engines for improving their performance (e.g. RenderingNG technology of chrome).

Future implementations of SingleCAnalyzer will be directed to the integration of novel analysis methods for scRNA-Seq and to the compatibility with new platforms and experimental protocols. At present, we provide semi-automated analysis of scRNA-Seq data on the cloud with analytical and interactive graphs, which enable the comprehensive analysis of results. It is freely available for scientists to explore the potential of their scRNASeq studies running quick analysis on an easy-to-use interface.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: <https://singlecanalyzer.eu>.

AUTHOR CONTRIBUTIONS

CP contributed to conception and design of the study. DB developed the web server. CP and AV designed and implemented the analysis workflow. DB and CP performed the platform test and system optimization. CP wrote the first draft of the manuscript. CP, DB, and AV wrote sections of the manuscript. All authors contributed to manuscript revision, read, and approved the submitted version.

REFERENCES

- Barrios, D., and Prieto, C. (2018). RJSplot: Interactive Graphs with R. *Mol. Inf.* 37, 1700090. doi:10.1002/minf.201700090
- Cakir, B., Prete, M., Huang, N., van Dongen, S., Pir, P., and Kiselev, V. Y. (2020). Comparison of Visualization Tools for Single-Cell RNAseq Data. *Nar. Genomics Bioinform.* 2, lqaa052. doi:10.1093/nargab/lqaa052
- Chen, S., Zhou, Y., Chen, Y., and Gu, J. (2018). Fastp: An Ultra-fast All-In-One FASTQ Preprocessor. *Bioinformatics* 34, i884–i890. doi:10.1093/bioinformatics/bty560
- Chen, H., Albergante, L., Hsu, J. Y., Lareau, C. A., Lo Bosco, G., Guan, J., et al. (2019). Single-cell Trajectories Reconstruction, Exploration and Mapping of Omics Data with STREAM. *Nat. Commun.* 10, 1903. doi:10.1038/s41467-019-09670-4
- Cunningham, F., Achuthan, P., Akanni, W., Allen, J., Amode, M. R., Armean, I. M., et al. (2019). Ensembl 2019. *Nucleic Acids Res.* 47, D745. doi:10.1093/nar/gky1113
- Gardeux, V., David, F. P. A., Shajkofci, A., Schwalie, P. C., and Deplancke, B. (2017). ASAP: A Web-Based Platform for the Analysis and Interactive Visualization of Single-Cell RNA-Seq Data. *Bioinformatics* 33, 3123–3125. doi:10.1093/bioinformatics/btx337
- Geer, L. Y., Marchler-Bauer, A., Geer, R. C., Han, L., He, J., He, S., et al. (2009). The NCBI BioSystems Database. *Nucleic Acids Res.* 38, D492–D496. doi:10.1093/nar/gkp858
- Guo, M., Wang, H., Potter, S. S., Whitsett, J. A., and Xu, Y. (2015). SINCERA: A Pipeline for Single-Cell RNA-Seq Profiling Analysis. *PLoS Comput. Biol.* 11, e1004575. doi:10.1371/journal.pcbi.1004575
- Haque, A., Engel, J., Teichmann, S. A., and Lönnberg, T. (2017). A Practical Guide to Single-Cell RNA-Sequencing for Biomedical Research and Clinical Applications. *Genome Med.* 9, 75. doi:10.1186/s13073-017-0467-4
- Hwang, B., Lee, J. H., and Bang, D. (2018). Single-cell RNA Sequencing Technologies and Bioinformatics Pipelines. *Exp. Mol. Med.* 50, 1–14. doi:10.1038/s12276-018-0071-8
- Jalili, V., Afgan, E., Gu, Q., Clements, D., Blankenberg, D., Goecks, J., et al. (2021). The Galaxy Platform for Accessible, Reproducible and Collaborative Biomedical Analyses: 2020 Update. *Nucleic Acids Res.* 48, W395. doi:10.1093/NAR/GKAA434
- Kiselev, V. Y., Kirschner, K., Schaub, M. T., Andrews, T., Yiu, A., Chandra, T., et al. (2017). SC3: Consensus Clustering of Single-Cell RNA-Seq Data. *Nat. Methods* 14, 483–486. doi:10.1038/nmeth.4236
- Korotkevich, G., Sukhov, V., Budin, N., Shpak, B., Artyomov, M. N., and Sergushichev, A. (2016). An Algorithm for Fast Preranked Gene Set Enrichment Analysis Using Cumulative Statistic Calculation. *bioRxiv*, 60012. doi:10.1101/060012

FUNDING

DB and AV was supported by Operational Programme of Youth Employment, European Social Fund (ESF), Junta de Castilla y Leon (JCyL). DB was supported by the PGC project (grant number PGC2018-093755-B-I00) of the Spanish Ministry of Science, Innovation and Universities. CP was supported by the PTA fellowship (grant number PTA2015-10483-I) of the Spanish Ministry of Economy, Industry and Competitiveness (MINECO).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fbinf.2022.793309/full#supplementary-material>

- Law, C. W., Chen, Y., Shi, W., and Smyth, G. K. (2014). Voom: Precision Weights Unlock Linear Model Analysis Tools for RNA-Seq Read Counts. *Genome Biol.* 15, R29. doi:10.1186/gb-2014-15-2-r29
- Lin, P., Troup, M., and Ho, J. W. (2017). CIDR: Ultrafast and Accurate Clustering through Imputation for Single-Cell RNA-Seq Data. *Genome Biol.* 18, 59. doi:10.1186/s13059-017-1188-0
- Love, M. I., Anders, S., and Huber, W. (2014). Differential Analysis of Count Data - The DESeq2 Package. *Genome Biol.* 15, 550. doi:10.1186/s13059-014-0550-8
- Lun, A. T. L., Riesenfeld, S., Andrews, T., Dao, T. P., Gomes, T., Marioni, J. C., et al. (2019). EmptyDrops: Distinguishing Cells from Empty Droplets in Droplet-Based Single-Cell RNA Sequencing Data. *Genome Biol.* 20, 63. doi:10.1186/s13059-019-1662-y
- Monier, B., McDermaid, A., Wang, C., Zhao, J., Miller, A., Fennell, A., et al. (2019). IRIS-EDA: An Integrated RNA-Seq Interpretation System for Gene Expression Data Analysis. *PLoS Comput. Biol.* 15, e1006792. doi:10.1371/journal.pcbi.1006792
- Moreno, P., Huang, N., Manning, J. R., Mohammed, S., Solovyyev, A., Polanski, K., et al. (2021). User-friendly, Scalable Tools and Workflows for Single-Cell RNA-Seq Analysis. *Nat. Methods* 18, 327–328. doi:10.1038/s41592-021-01102-w
- Patro, R., Duggal, G., Love, M. I., Irizarry, R. A., and Kingsford, C. (2017). Salmon Provides Fast and Bias-Aware Quantification of Transcript Expression. *Nat. Methods* 14, 417–419. doi:10.1038/nmeth.4197
- Perraudeau, F., Risso, D., Street, K., Purdom, E., and Dudoit, S. (2017). Bioconductor Workflow for Single-Cell RNA Sequencing: Normalization, Dimensionality Reduction, Clustering, and Lineage Inference. *F1000Res* 6, 1158. doi:10.12688/f1000research.12122.1
- Prieto, C., and Barrios, D. (2019). RaNA-Seq: Interactive RNA-Seq Analysis from FASTQ Files to Functional Analysis. *Bioinformatics* 36, 1955–1956. doi:10.1093/bioinformatics/btz854
- Qiu, X., Mao, Q., Tang, Y., Wang, L., Chawla, R., Pliner, H. A., et al. (2017). Reversed Graph Embedding Resolves Complex Single-Cell Trajectories. *Nat. Methods* 14, 979–982. doi:10.1038/nmeth.4402
- Robinson, M. D., McCarthy, D. J., and Smyth, G. K. (2010). edgeR: A Bioconductor Package for Differential Expression Analysis of Digital Gene Expression Data. *Bioinformatics* 26, 139–140. doi:10.1093/bioinformatics/btp616
- Scholz, C. J., Biernat, P., Becker, M., Baßler, K., Günther, P., Balfer, J., et al. (2018). FASTGenomics: An Analytical Ecosystem for Single-Cell RNA Sequencing Data. *bioRxiv*. doi:10.1101/272476
- Seyednasrollah, F., Laiho, A., and Elo, L. L. (2013). Comparison of Software Packages for Detecting Differential Expression in RNA-Seq Studies. *Brief. Bioinform.* 16, 59–70. doi:10.1093/bib/bbt086
- Shum, E. Y., Walczak, E. M., Chang, C., and Christina Fan, H. (2019). Quantitation of mRNA Transcripts and Proteins Using the BD Rhapsody Single-Cell Analysis System. *Adv. Exp. Med. Biol.* 1129, 63–79. doi:10.1007/978-981-13-6037-4_5

- Soneson, C., and Delorenzi, M. (2013). A Comparison of Methods for Differential Expression Analysis of RNA-Seq Data. *BMC Bioinform.* 14, 91. doi:10.1186/1471-2105-14-91
- Speir, M. L., Bhaduri, A., Markov, N. S., Moreno, P., Nowakowski, T. J., Papatheodorou, I., et al. (2021). UCSC Cell Browser: Visualize Your Single-Cell Data. *Bioinformatics* 37, 4578–4580. doi:10.1093/bioinformatics/btab503
- Srivastava, A., Malik, L., Smith, T., Sudbery, I., and Patro, R. (2019). Alevin Efficiently Estimates Accurate Gene Abundances from dscRNA-Seq Data. *Genome Biol.* 20, 65. doi:10.1186/s13059-019-1670-y
- Stuart, T., Butler, A., Hoffman, P., Hafemeister, C., Papalexi, E., Mauck, W. M., et al. (2019). Comprehensive Integration of Single-Cell Data. *Cell* 177, 1888–1902. doi:10.1016/j.cell.2019.05.031
- VIZBI (2022). Posters. Available at: <https://vizbi.org/Posters/2021/vC15> (Accessed March 4, 2022).
- Wagner, F., and Yanai, I. (2018). Moana: A Robust and Scalable Cell Type Classification Framework for Single-Cell RNA-Seq Data. *bioRxiv*. doi:10.1101/456129
- Wolf, F. A., Angerer, P., and Theis, F. J. (2018). SCANPY: Large-Scale Single-Cell Gene Expression Data Analysis. *Genome Biol.* 19, 15. doi:10.1186/s13059-017-1382-0
- Young, M. D., Wakefield, M. J., and Smyth, G. K. (2010). Goseq : Gene Ontology Testing for RNA-Seq Datasets Reading Data. *Gene* 11, 1–21. Available at: <http://cobra20.fhrc.org/packages/release/bioc/vignettes/goseq/inst/doc/goseq.pdf>.
- Zheng, G. X., Terry, J. M., Belgrader, P., Ryvkin, P., Bent, Z. W., Wilson, R., et al. (2017). Massively Parallel Digital Transcriptional Profiling of Single Cells. *Nat. Commun.* 8, 14049. doi:10.1038/ncomms14049
- Zhu, X., Wolfgruber, T. K., Tasato, A., Arisdakessian, C., Garmire, D. G., and Garmire, L. X. (2017). Granatum: A Graphical Single-Cell RNA-Seq Analysis Pipeline for Genomics Scientists. *Genome Med.* 9, 108. doi:10.1186/s13073-017-0492-3

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Prieto, Barrios and Villaverde. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



BioViz Connect: Web Application Linking CyVerse Cloud Resources to Genomic Visualization in the Integrated Genome Browser

Karthik Raveendran[†], Nowlan H. Freese[†], Chaitanya Kintali, Srishti Tiwari, Pawan Bole, Chester Dias and Ann E. Loraine^{*}

Department of Bioinformatics and Genomics, University of North Carolina at Charlotte, Charlotte, NC, United States

OPEN ACCESS

Edited by:

Jim Procter,
University of Dundee, United Kingdom

Reviewed by:

William C. Ray,
Nationwide Children's Hospital,
United States
Ram Vinay Pandey,
Karolinska University Hospital,
Sweden

*Correspondence:

Ann E. Loraine
aloraine@uncc.edu

[†]These authors have contributed
equally to this work

Specialty section:

This article was submitted to
Data Visualization,
a section of the journal
Frontiers in Bioinformatics

Received: 25 August 2021

Accepted: 28 April 2022

Published: 23 May 2022

Citation:

Raveendran K, Freese NH, Kintali C,
Tiwari S, Bole P, Dias C and Loraine AE
(2022) BioViz Connect: Web
Application Linking CyVerse Cloud
Resources to Genomic Visualization in
the Integrated Genome Browser.
Front. Bioinform. 2:764619.
doi: 10.3389/fbinf.2022.764619

Genomics researchers do better work when they can interactively explore and visualize data. Due to the vast size of experimental datasets, researchers are increasingly using powerful, cloud-based systems to process and analyze data. These remote systems, called science gateways, offer user-friendly, Web-based access to high performance computing and storage resources, but typically lack interactive visualization capability. In this paper, we present *BioViz Connect*, a middleware Web application that links CyVerse science gateway resources to the Integrated Genome Browser (IGB), a highly interactive native application implemented in Java that runs on the user's personal computer. Using *BioViz Connect*, users can 1) stream data from the CyVerse data store into IGB for visualization, 2) improve the IGB user experience for themselves and others by adding IGB specific metadata to CyVerse data files, including genome version and track appearance, and 3) run compute-intensive visual analytics functions on CyVerse infrastructure to create new datasets for visualization in IGB or other applications. To demonstrate how *BioViz Connect* facilitates interactive data visualization, we describe an example RNA-Seq data analysis investigating how heat and desiccation stresses affect gene expression in the model plant *Arabidopsis thaliana*. The RNA-Seq use case illustrates how interactive visualization with IGB can help a user identify problematic experimental samples, sanity-check results using a positive control, and create new data files for interactive visualization in IGB (or other tools) using a Docker image deployed to CyVerse via the Terrain API. Lastly, we discuss limitations of the technologies used and suggest opportunities for future work. *BioViz Connect* is available from <https://bioviz.org>.

Keywords: SR45a, AT1G07350, Arabidopsis, abiotic stress, Integrated Genome Browser, CyVerse, visualization, Terrain API

Abbreviations: API, Application Programming Interface; CSS, Cascading Style Sheets; HTML, HyperText Markup Language; IGB, Integrated Genome Browser; REST, REpresentational State Transfer.

INTRODUCTION

Science gateways are Web sites that implement user-friendly interfaces to high performance computing and storage systems (Wilkins-Diehr et al., 2008). Science gateways typically assemble and curate discipline-specific, command-line, Unix-based tools within a single, easy-to-use interface, enabling users to run compute-intensive processing on datasets too large for a personal computer (Giardine et al., 2005; Goff et al., 2011; Merchant et al., 2016). In a typical use case, domain researchers upload their “raw” (unprocessed) data to the gateway site and then operate the gateway’s Web-based interface to create custom processing and analysis pipelines, where a pipeline is defined as tasks performed in sequence by non-interactive tools which emit and consume well-understood file types and formats. Common pipeline tasks in genomics include aligning RNA-Seq sequences onto a reference genome to produce BAM (binary alignment) format files (Li et al., 2009), generating scaled RNA-Seq coverage graphs from the “BAM” files using tools such as deepTools bamCoverage (Ramirez et al., 2016), or searching promoter regions for sequence motifs common to sets of similarly regulated genes using tools such as DREME (Bailey, 2011).

A science gateway aims to provide a single point of access for tools needed to process and analyze data from a research project. However, native visualization tools with their own graphical user interfaces separate from a Web browser are difficult to use with Web-based science gateway systems. The Integrated Genome Browser from BioViz.org (Nicol et al., 2009; Freese et al., 2016) and the Broad Institute’s Integrative Genomics Viewer (Robinson et al., 2011) exemplify this problem. Both tools require that data files reside on the user’s local file system or that they be accessible *via* HTTP (hypertext transfer protocol) and addressable *via* a file-specific URL (Uniform Resource Locator). If the gateway system does not allow URL-based access to data, then users must download the data files onto their local computer file system, which may not be practical or allowed.

Related problems confront visualization systems implemented as Web applications, deployed on Web hosts and not the user’s local computer. Using Web applications to visualize data can be even more challenging for users, because these applications often require hard-to-set-up data storage and delivery mechanisms specialized to the application. To view one’s data using the Web-based UCSC Genome Browser software, for example, users can either deploy their own copy of the software, which is difficult, or they can instead set up a UCSC Track Hub server, which is less technically challenging but nonetheless requires Track Hub-specific meta-data files to be created and configured (Raney et al., 2014). Similarly, using the JBrowse Web-based genome browser requires deploying data in JBrowse-compatible formats (Buels et al., 2016).

Another typical requirement for science gateways is extensibility, meaning they require a way for gateway developers or users to add new tools to the system to accommodate or even potentiate new directions for research. The CyVerse science gateway, the focus of this article, supports

extensibility by allowing developers to create and deploy CyVerse Apps, which are user-contributed container images that run within a CyVerse-provided container environment (Devisetty et al., 2016). Users create containers using Docker and then contribute their container image along with metadata specifying input parameters and accepted data types to CyVerse. Once accepted and deployed, the container is configured to run as an asynchronous “job” within the CyVerse infrastructure *via* a queuing system. Thus, Apps run non-interactively and therefore are not well-suited to providing interactive, exploratory visualization. However, these Apps do provide a means to create new input data for visualization, as we explore here.

In this paper, we introduce BioViz *Connect*, a Web application that overcomes limitations described above to add genome visualization capability to the CyVerse science gateway system. Previously called iPlant, the CyVerse science gateway is a United States National Science Foundation funded cyberinfrastructure project with the aim of providing computational resources for life sciences researchers (Goff et al., 2011; Merchant et al., 2016). We chose to work with CyVerse in this study because it features a rich Application Programming Interface (API), the Terrain REST API, that supports secure computational access to CyVerse data storage and analysis resources.

Using this API, we implemented a new visualization-focused interface to these resources, called BioViz *Connect*, using the Integrated Genome Browser (IGB) as the demonstration application. We selected IGB because it offers one of the richest feature sets for visual analysis in genomics [for descriptions of IGB functionality, see (Nicol et al., 2009; Gulledge et al., 2014; Loraine et al., 2015; Freese et al., 2016; Mall et al., 2016)] and because we are members of the core IGB development team. Therefore, we possessed insider’s knowledge of the featured visualization application that allowed us to modify IGB as needed for the project.

BioViz *Connect* enables users of Integrated Genome Browser to visually analyze their CyVerse data without having to download entire files to their local computer or migrate their data into application specific data stores. BioViz *Connect* lets users annotate their data sets with metadata, which control how the data will look when imported into the IGB and also indicate the genome version referenced in the data. Finally, BioViz *Connect* lets users run compute-intensive visual analytics algorithms, implemented as CyVerse Apps.

In the following sections, we describe how BioViz *Connect* is implemented, explaining the technology stack used and how BioViz *Connect* interacts with the CyVerse science gateway resources *via* its Terrain API. Next, we describe how BioViz *Connect* enables flow of data into the IGB desktop software by activating a REST API endpoint residing in IGB itself. To illustrate the functionality, we describe an example use case scenario for BioViz *Connect* in which a hypothetical analyst uses visualization and visual analytics tools within IGB in conjunction with their CyVerse account to quality-check and analyze an RNA-Seq data set from *Arabidopsis thaliana* plants undergoing desiccation and heat stresses. Lastly, we discuss

Integrated Genome Browser REST Endpoint

Integrated Genome Browser is a free, open-source desktop software program written in Java which users download and install on their local computer systems (IGB, RRID: SCR_011792) (Nicol et al., 2009; Freese et al., 2016). Installers for Linux, MacOS, and Windows platforms are available at <https://bioviz.org>.

The IGB source code resides in a git repository hosted on Atlassian's bitbucket.org site (<https://bitbucket.org/lorainelab/integrated-genome-browser>). When viewed on the BitBucket git repository's Web site, changes to the code called "commits" link to pages on the project management Web site documenting the motivation for the change and/or technical challenges encountered, thus making the source code easier to manage and understand. The project management Web site uses Jira from Atlassian Software, with URL <https://jira.bioviz.org>. IGB version 9.1.4 or greater is required for IGB to connect to BioViz Connect.

IGB contains a simple Web server configured to respond to REST-style queries on an IGB-specific port on the user's local computer. JavaScript code downloaded into the Web browser when users visit BioViz Connect pages enables requesting URLs addressed to "localhost", the user's computer, using the IGB-specific port. IGB intercepts these requests and performs actions dictated by parameters embedded in the URL text. This mechanism repurposes a REST endpoint dating from the earliest releases of IGB from the early 2000s. The IGB Users' Guide hosted at <https://wiki.bioviz.org/confluence> describes these and other features.

BioViz Connect Metadata

BioViz Connect uses the Terrain Metadata API to manage and obtain IGB-specific metadata for files and folders. The Terrain API represents metadata items as triplets containing Attribute, Value, and Unit. A metadata item's Attribute attaches meaning to what the metadata contains, and application developers can create their own custom Attributes to support diverse purposes. For example, since BioViz Connect is concerned with genomic data visualization, we created custom Attributes signaling genome assembly version, visual style information such as foreground color and background color, and free text comments on the data provided by the user, which are displayed in BioViz Connect's Web interface. A metadata item's Value is specific to the file or folder being tagged. BioViz Connect uses the Unit value to indicate that the metadata element concerns IGB and the BioViz Connect application.

The genome identifier attribute requires further explanation, as matching genome version names across systems has caused many problems for genome browsers and their users. Integrated Genome Browser, like many other systems, uses an application-specific scheme for naming genome versions, and contains a listing of synonyms matching these IGB-specific names onto genome version names from other systems. For example, the IGB genome version named H_sapiens_Feb_2009 is the same as UCSC genome version name hg17, which is the same as NCBI version 35. The

BioViz Connect user interface includes components for users to view, designate, or change the genome version metadata associated with individual files. To ensure compatibility with IGB, BioViz Connect uses a list of IGB-formatted genome identifiers hosted on the IGB Quickload site (<http://igbquickload.org/quickload/>) to configure the genome version selection components, implemented as menus. When users operate the interface to view data within IGB, the genome version metadata, along with style metadata, are passed to IGB via its localhost REST endpoint. This ensures that the data appear in the context of the correct genome assembly, alongside other data already loaded from BioViz Connect or other sources, while also enabling the user to specify in advance how the data will look once it appears in IGB. In addition, if other users load the same files, the data will look the same.

Enabling Access to Data via Public URLs

The flow of data from CyVerse into IGB depends on two key technical features of the CyVerse data storage and hosting system. First, the Terrain API enables users to create publicly accessible URLs for data files in their accounts, and these URLs can be enabled or disabled at will. In the current implementation, URLs created in this way are accessible to any internet user. Second, the CyVerse infrastructure supports HTTP range requests for these URLs, enabling clients such as IGB to request subsets of data, thus avoiding having to download or transfer an entire data file.

The BioViz Connect interface is designed to make the process of managing these URLs as easy as possible, similar to commercial cloud storage systems such as Dropbox and Google Drive that let users create, destroy, and manage public links to individual files and folders. Within the BioViz Connect interface, users create URLs for individual files by right clicking the file and selecting the "Manage Link" option. Selecting this option opens a right panel display in which the current status of the file is shown, and users can toggle between making the file public or private (Figures 2A,B).

As shown in Figure 2B, the text of this public URL is visible to the user, and users can copy it to their system clipboard by clicking the "copy" icon. The Terrain API determines the link text, and currently, it always contains the user's chosen name for the file and the path to the file within the virtual file system, preceded by the prefix shown in Figure 2B. We expose this detail to users because increasingly many researchers are using their CyVerse accounts to host files, and the current transparency and predictability of these URLs seems important for them to know about. Likewise, if the pattern ever changes, they will need to know this, as well.

BioViz Connect Deployment

BioViz Connect is managed using ansible roles and playbooks publicly available in a git repository from <https://bitbucket.org/lorainelab/bioviz-connect-playbooks>. The playbooks contain two sets of tasks. One set of tasks creates a virtual machine using the Amazon EC2 Web service. Once the host is created and running, a second set of ansible tasks installs and configures software on the host, including an Apache2 Web server, a MySQL database, and the BioViz Connect code base. Playbook users can specify the BioViz Connect repository and branch they wish to deploy, which facilitates rapid testing of proposed new code. During the provisioning process, a call is made to a Terrain endpoint that provides a list of all CyVerse

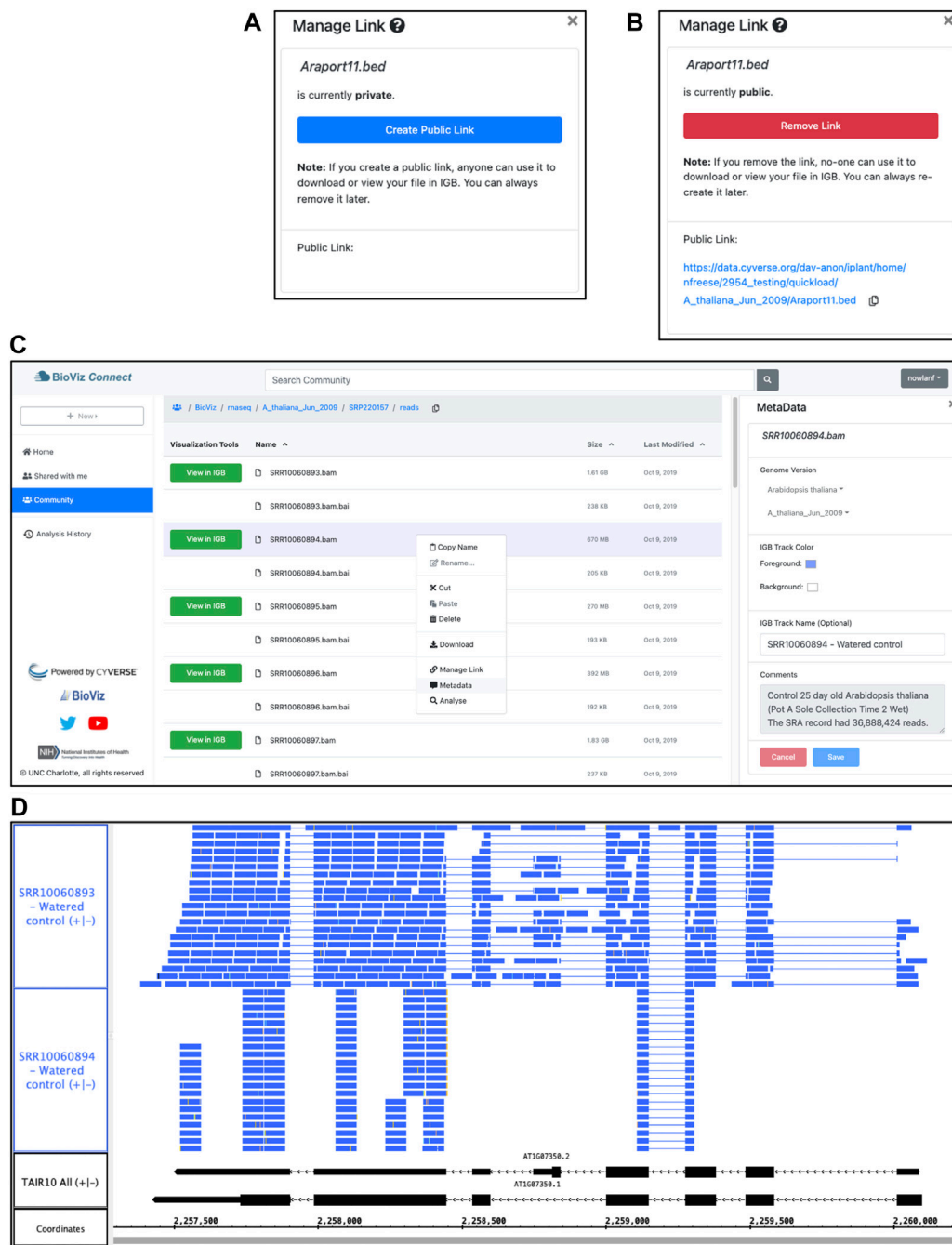


FIGURE 2 | BioViz Connect interface and IGB data visualization. **(A,B)** BioViz Connect “Manage Link” interface, from the right panel display. By default, files are not publicly accessible, and the interface appears as in **(A)**. Clicking the button labeled “Create Public Link” creates a public link, switching the display to the image shown in **(B)**. **(C)** BioViz Connect main page. The left panel shows shortcuts to home, shared, community folders. The middle panel lists files and folders. The right panel shows the selected file’s metadata. **(D)** SRR10060893.bam and SRR10060894.bam files viewed in IGB overlapping the SR45a gene of *A. thaliana*. The track labeled TAIR10 mRNA shows SR45a gene models AT1G07350.1 and AT1G07350.2.

asynchronous analysis apps that can produce output visible to IGB. These data are then used to construct the “analysis” sections of the user interface, and are stored in the BioViz Connect relational database, co-located on the same host.

BioViz Connect Interface for Running Visual Analysis Apps

When users right-click a file name in BioViz Connect, a context menu appears with an option labeled “Analyse.” Information

about IGB-compatible Apps, the file types they can accept, and App parameters are stored in the relational database configured during deployment as described above. When a user selects this option, BioViz *Connect* queries the database to identify IGB Community Apps that accept the file as input, and these are then displayed to the user. Once the user has selected an App, another query retrieves additional information about it, such as user-friendly description of what the App does, which is then displayed to the user.

The CyVerse ecosystem contains many hundreds of Apps, many of which are redundant or obsolete, and so the BioViz Team controls which ones are shown to users by adding them to the IGB Community, a CyVerse organizing concept that groups resources (such as Apps) according to which users can use or modify them. BioViz *Connect* only shows Apps that have been added to the IGB Community.

RNA-Seq Data

RNA-Seq data presented in the use case scenario are from Sequence Read Archive Bioproject PRJNA509437 (Leinonen et al., 2011), an experiment in which Arabidopsis plants underwent either a 3-h, non-lethal heat stress or a multi-day desiccation stress. Two post-treatment sample time points were collected for treated plants and their untreated control counterparts, with two to four replicates per sample type and 23 samples in total. Sample libraries were sequenced in single-end runs of the Illumina platform and are identified by their run identifiers. BAM files were generated by aligning sequence reads to the Arabidopsis June 2009 reference genome assembly using TopHat2 (TopHat, RRID: SCR_013035) (Kim et al., 2013). The data are available in the Community folder of publicly accessible datasets, represented as a folder in the left-side panel of the BioViz *Connect* display.

RESULTS

Understanding and Navigating the BioViz *Connect* Interface

Our design goal in creating BioViz *Connect* was to give the user a feeling of almost limitless computational power and space by integrating seamlessly with the CyVerse “cloud.” Doing so requires that users identify themselves to the system by entering a username and password, but how this process takes place can easily destroy the illusion of seamless access. To avoid this, we used an OAuth-style Terrain API endpoint that delegates logins to CyVerse infrastructure, preventing BioViz *Connect* from learning the user’s password.

To begin a session with BioViz *Connect*, the user opens the BioViz.org website in a Web browser, selects the link labeled BioViz *Connect*, and then clicks the link labeled “Sign in with your CyVerse ID”. This action opens a Central Authentication Service (CAS) page, hosted by CyVerse, where users enter their CyVerse username and password, or sign up for a new account if they do not already have one. The Web browser then returns to a “call-back” URL on the BioViz.org site, which displays the BioViz *Connect* user interface, a browsable, sortable, paginated view of the user’s CyVerse home directory and its contents (Figure 2C).

This view of files and data resembles the interface for commercial, consumer-focused cloud storage systems, a deliberate design choice aimed at building on many users’ familiarity with Google Drive, the Dropbox Web interface, and others. This interface displays a sortable, table-based view of the user’s home directory within the CyVerse file storage system, displaying a listing of files and folders the user has uploaded to their account or created using CyVerse Apps, including BioViz *Connect* Apps described in later sections. Single-clicking a file or folder selects it, double-clicking a folder opens it and displays the contents, and double-clicking a file opens a metadata display showing information about the file (Figure 2C). A bread crumb display at the top of the page shows the path from the root folder to the currently opened folder, and a copy icon next to the breadcrumb allows the user to copy the folder name and path. The browser forward and back buttons work as expected, and users can bookmark individual screens for faster navigation. The URLs displayed in the browser’s URL bar match the currently opened folder’s location, making the interface feel more polished and user-friendly by ensuring that every user-facing detail, including the URL, mimic and reinforce how the user has organized their data within the CyVerse virtual file system.

The top part of every BioViz *Connect* page also features a search bar that can be used to find files and folders with names matching a user-entered query string. Matches are returned in a list view similar to the original table view, and users can sort the results list by name, size, or date modified. Only files for which the user has read access and that reside in the currently visible section (Home, Community, or Shared with me) are returned. On the left side of every page, BioViz *Connect* displays icons representing shortcut links to the user’s home directory, a publicly available community data folder, and other destinations. The “Community” folder contains data published for all CyVerse users, including the example RNA-Seq data set for the use case scenario described in the next section.

Using BioViz *Connect* to View Data in Integrated Genome Browser

To demonstrate BioViz *Connect* functionality, we next describe an example use case scenario in which a hypothetical researcher visually analyzes data from a typical RNA-Seq experiment. The use case focuses on two main tasks: visually checking data quality and then confirming differential expression of a control gene known to be regulated by the treatment.

The experimental design included two treatments, heat and desiccation stress, their controls, and two time points, totaling six sample types, each with two to four replicates. The RNA-Seq sequences are available in the Sequence Read Archive, and the researcher has obtained the data, aligned it to the reference genome, and then contributed the files to the Community folder. Alignment files are stored in the file path “BioViz/rnaseq/A_thaliana_Jun_2009/SRP220157/reads”. The user has also annotated each file using the BioViz *Connect* interface, adding the genome version, visual style information, and notes describing each sample.

Now that the data are organized and annotated, the researcher uses the BioViz *Connect* interface to import the data into

Integrated Genome Browser for visualization and proceeds to look at each file, one by one, to check the quality of the alignments and confirm file identity. BioViz *Connect* makes this task easy to perform. To illustrate, we discuss RNA-Seq alignment files SRR10060893.bam and SRR10060894.bam, replicate control samples from time point one of the heat stress treatment. A quick scan of files listed in the BioViz *Connect* table view shows that SRR10060893.bam has size 1.61 GB, about twice the size of SRR10060894.bam, which is 0.669 GB. The user has annotated the files with the number of sequence reads obtained per sample, around 37 million for each. Because the samples were sequenced to about the same depth, their resulting alignment files ought to have similar sizes. Visualizing the sequence read alignments will help explain the discrepancy.

To visualize the alignments, the user launches Integrated Genome Browser, which is already installed on the local computer, downloaded from the BioViz.org Web site. Once IGB is running, the user clicks the “View in IGB” button available in the “Visualization Tools” column in the BioViz *Connect* table view, repeating this action for each file (**Figure 2C**). This action causes JavaScript code running within the Web browser to request data from a local URL (domain “localhost”) corresponding to a REST endpoint implemented within IGB. The URL includes parameters such as the publicly accessible URL for the data file, the IGB name of its reference genome, and visual style information indicating how the file should look once loaded into IGB. In response, IGB opens the requested genome version associated with the file and adds the file as a new track to the display.

To check assumptions about a new data set, it is useful to visualize a gene of known behavior, such as a gene already known to be regulated by the experimental treatment. Prior work from our lab and others have shown that SR45a, encoding an RNA-binding protein, is upregulated by heat and desiccation stresses, making it a good choice for this purpose (Yoshimura et al., 2011; Gullede et al., 2012). To find the gene, the analyst enters SR45a into IGB’s search interface at the top left of the IGB window, which zooms and pans the display to the gene’s position in the genome. Next, the user loads the alignments into the display by clicking the “Load Data” button at the top right of the IGB window. Once the data load, the user customizes track appearance by modifying vertical zoom setting and changing the number of sequences that can be shown individually in a track (stack height), creating the view shown in **Figure 2D**.

This customized view makes problems with SRR10060894 obvious at a glance. The alignments for this sample appear to stack on top of each other in orderly, uniform towers covering only 30% of the gene’s exonic sequence. By contrast, the alignments for sample SRR10060893 cover most of the exonic sequence and also include many spliced reads split across introns. The sparser pattern observed in SRR10060894 typically arises when the library synthesis process included too many polymerase chain reaction amplification cycles, reducing the diversity of resulting sequence data. This pattern indicates that the user should exclude SRR10060894 from further analysis, but the other file appears to be fine.

Comparing Sequencing Depth and Complexity Using Integrated Genome Browser Visual Analytics

Repeating the preceding process with other samples in the dataset, the user identifies another problematic pair of files. The files are replicates, but like the previous example, the files sizes differ. The alignments file SRR10060911.bam is 1.83 Gb, but its replicate SRR10060912.bam is only 0.454 Gb. Opening and viewing the alignment files in IGB, the user confirms that one file appears to contain more data than the other (**Figure 3A**). To quantify this observation, the user takes advantage of a simple, interactive visual analytics feature within IGB: selection-based counting. As with PowerPoint and many other graphical applications, IGB users can click-drag the mouse over graphical elements to select a group of items and then single-click while pressing SHIFT or CTRL-SHIFT keys to add or remove items from selection group. IGB reports the number of currently selected items in the Selection Info box at the top right of the IGB window. Using this feature, the researcher finds that sample SRR10060912 contains 1,925 alignments covering SR45a, and sample SRR10060911 has 10,867 alignments, nearly five times as many.

By further configuring track height and appearance settings, and operating IGB’s dynamic vertical and horizontal zoom controls, the user can stretch the display in each dimension independently to reveal more detail about the alignments (**Figures 3B,C**). From this new view of the data, the user can tentatively conclude that alignment pattern diversity is similar in each sample, but the depth of sequencing was greater in SRR10060911. To confirm the finding, the user then applies a visual analytics function (called a “Track Operation” within IGB) that creates coverage graphs, also called depth graphs, using data from the read alignment tracks (**Figure 3D**). To make a coverage graph, the user right-clicks a track label for a read alignment track and chooses option “Track Operations > Depth Graph (All).” This generates a new track showing a graph in which the y-axis indicates the number of sequences aligned per x-axis position, corresponding to base pair positions. After modifying the y-axis lower and upper boundary values (using controls in IGB’s Graph tab), the user again can observe that the pattern of alignments is similar between the two samples, but the overall level of sequencing was different. Thus, the file size difference most likely is due to a difference in sequencing depth rather than a problem with the library synthesis, as was the case in the previous example.

Normalizing Coverage Graphs to Compare Gene Expression Visually

Coverage graphs set to the same scale allow comparing gene expression across sample types, but only if the libraries were sequenced to approximately the same depth. If not, then coverage graphs need to be normalized before comparing them. Scaling coverage graphs within IGB is impractical, however, as it would require downloading, reading, and processing the entire bam-format alignments file. A better approach is to off-load

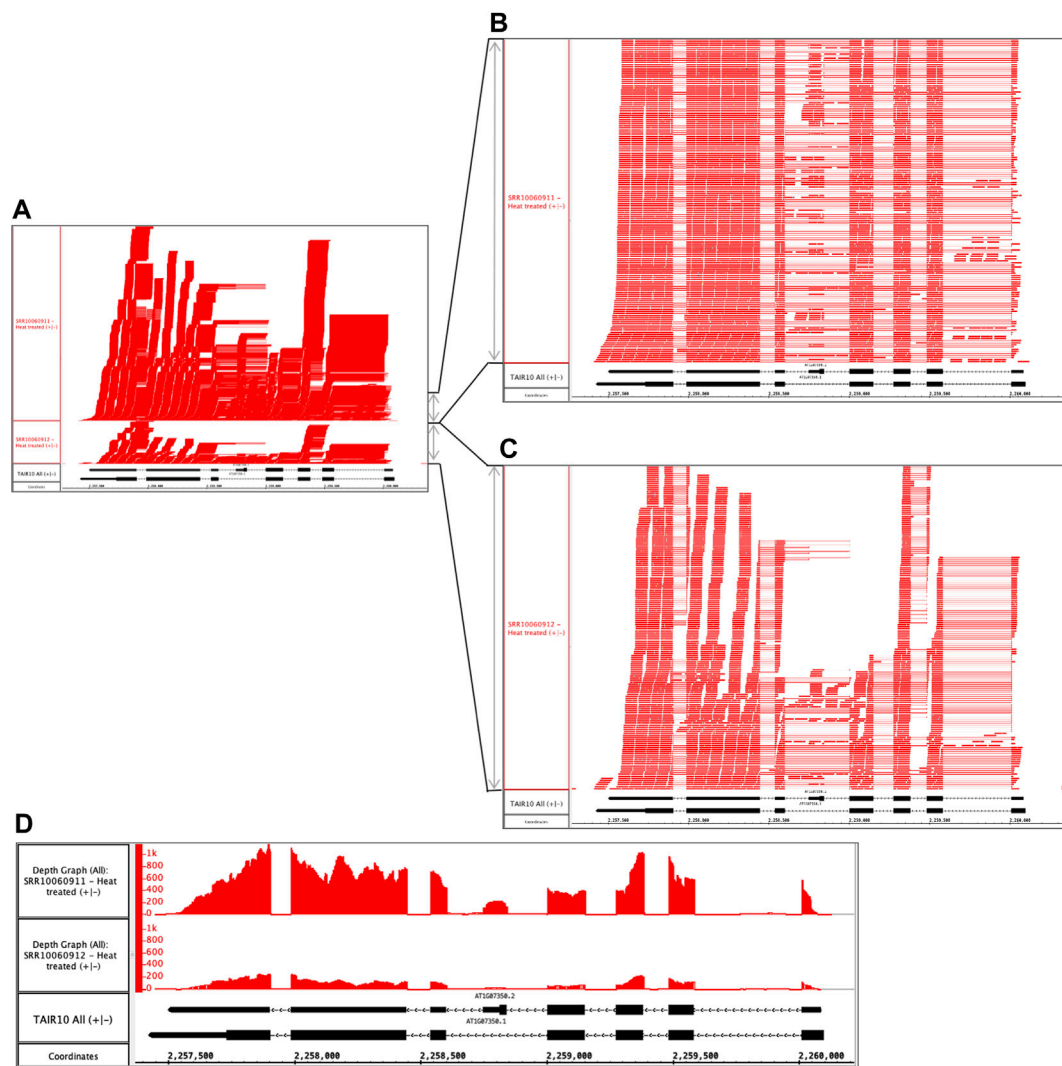


FIGURE 3 | Heat treated samples viewed in IGB. **(A)** Vertical dimension is compressed to show all alignments. **(B)** SRR10060911 and **(C)** SRR10060912 tracks stretched vertically to reveal alignment patterns in more detail. **(D)** Alignment coverage graphs calculated within IGB using alignments from **(A)**. The y-axis values represent the number of aligned sequences per base pair position indicated on the coordinates track. The track labeled TAIR10 mRNA shows SR45a gene models AT1G07350.1 and AT1G07350.2.

computationally intensive visual analytics tasks to CyVerse cloud computing resources. To demonstrate the value of this strategy, we deployed the deepTools genomeCoverage command line tool from the deepTools suite (DeepTools, RRID: SCR_016366) as a new IGB-friendly CyVerse App (Ramirez et al., 2016).

To create a scaled coverage graph, the user returns to BioViz Connect, right-clicks a bam format file, and chooses “Analyse.” This opens the Analysis right-panel display, which lists all IGB-compatible CyVerse Apps that can accept the selected file type as input (**Figure 4A**). Selecting “Make scaled coverage graph” opens a form with options for creating the graph using the genomeCoverage algorithm (**Figure 4B**). The interface includes a place for the user to enter names for the analysis and for the output file that will be produced. The user then clicks “Run Analysis” button, which calls upon the CyVerse analysis

API to run the App with specified parameters using CyVerse computing resources. The request to run the App and the work it performs are called “jobs,” and jobs are carried out asynchronously, running and completing only when resources they require become available, as with other systems set up for high-performance computing. Users can check job status by using the Analyses History in the BioViz Connect interface (**Figure 4C**), where Analyses are listed as Queued (waiting to run), Running, Failed, or Completed. The length of the time to complete a job is dependent on the size of the queue, the analysis being carried out, and the size of the file. When we ran these analyses ourselves, the “Make scaled coverage graph” job took 7 min and 12 s for the SRR10060911.bam as its file size is 1.83 GB, whereas SRR10060912.bam took only 5 min and 52 s, most likely due to its smaller file size of 455 MB. Larger files may take longer,

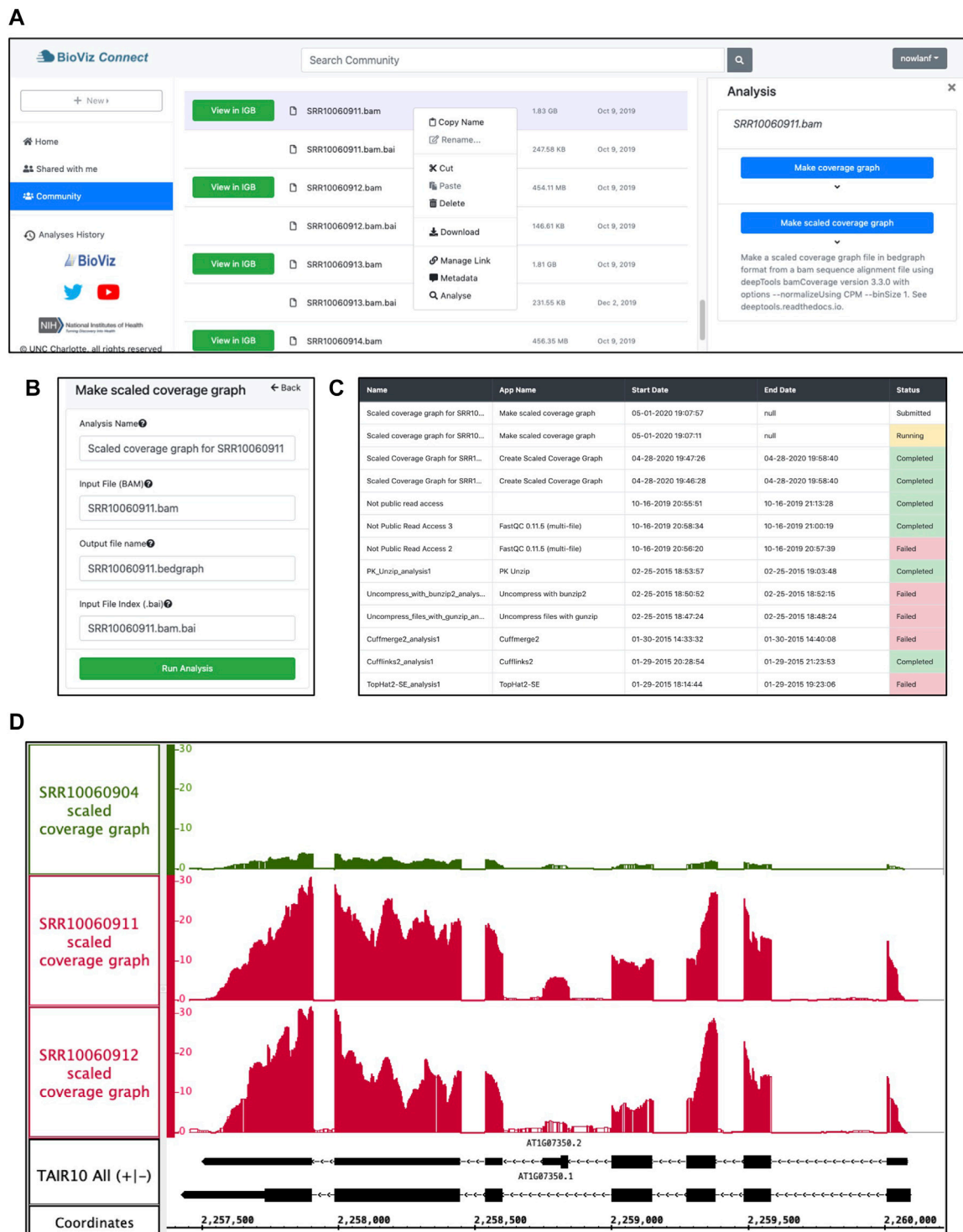


FIGURE 4 | Example analysis in BioViz Connect with output visualized in IGB. **(A)** BioViz Connect main page with analysis right panel open. **(B)** Scaled coverage graph analysis options for naming the analysis, selecting input file, output file name, and index file selection. **(C)** Analyses History showing the status of current and previous jobs. **(D)** SRR10060904 (control), SRR10060911 (heat treated), and SRR10060912 (heat treated) scaled coverage graphs viewed in IGB overlapping the SR45a gene of *A. thaliana*.

for example, an 8.89 GB file took 38 min and 54 s to complete. Independent of BioViz *Connect*, the CyVerse infrastructure sends an email to users when jobs finish. When a job finishes, any files or folders it creates appear in the analyses folder in the user's home directory, or in the same location as the input files, if those are stored in a location where the user has permission to modify or add to the folder. To quickly navigate to results, users can click the analysis name in the Analyses History, opening the folder where the output data files are stored.

Figure 4D shows sample App output, a visualization of the SR45A region with three scaled coverage graphs loaded from bigwig data files, a compact binary format for representing numeric values associated with base pairs in a genome map. Two heat-treated and one control sample are shown. The three coverage graphs have been configured to use the same y-axis scale, making it obvious that the heat treatment elevated SR45A gene expression, consistent with previously published reports. The image presents a clear visual argument in favor of this conclusion, and it also shows the user how much the expression level measurement varies across the gene body, something a single summary statistic cannot provide.

DISCUSSION

BioViz *Connect* introduces and demonstrates innovations in the field of science gateway development and research, while providing useful functionality for researchers seeking to understand and visualize genomic data. BioViz *Connect* enables users of the CyVerse science gateway to visualize genomic data files from their CyVerse accounts using Integrated Genome Browser, a desktop application. To our knowledge, BioViz *Connect* is the first and only resource that integrates remote CyVerse file storage and computational resources with a genome browser native to the local computer, achieving this cross-application communication *via* localhost REST endpoints.

We implemented BioViz *Connect* using the CyVerse Terrain API, a collection of remote REST endpoints that form a comprehensive computational interface to CyVerse resources. However, CyVerse and its Terrain API were not the first cloud system we considered. Our larger goal was to expand users' experience of genome browsing by connecting the interactivity and speed of a native, desktop genome browser (such as IGB) with the vast resources of cloud-based, remote storage and computing systems, making it easier for users to store and share their data with others and also run compute-intensive visual analytics algorithms that would never be possible using just the user's personal computer. To achieve this, we considered several commercial and public-sector systems, but selected CyVerse because of its focus on supporting scientific research, its free cost for users, and its early support for computational interfaces *via* APIs (Dooley et al., 2012).

At first, we proposed to use the CyVerse Agave API, which was well-documented and well-supported at the time. Since then, at least two other groups have published workflow management sites that use Agave, justifying our original choice (Wang et al.,

2018; Hubbard et al., 2020). However, several months after launching our project, we discovered that Agave's manipulation of user data conflicted with CyVerse's own Discovery Environment interface, then a Web interface resembling a personal computer desktop. We also learned that Agave lacked support for HTTP range requests against data files, an essential feature from our perspective, and that this feature was unlikely to be added, as Agave's maintainers were in the process of migrating to a new version to be called "Tapis." Realizing our problem, they recommended we instead use Terrain, the API that powers the Discovery Environment interface. After consulting with developer teams working on Terrain and Discovery Environment, both based at University of Arizona, we decided to use Terrain.

We chose to use Integrated Genome Browser as the visualization component of BioViz *Connect* for several reasons. The first was that we wanted to demonstrate and explore a connection between cloud-based resources and a pre-existing, native, desktop application already in wide use, and IGB satisfied this requirement. The second major reason was convenience. As the core development group for IGB, we understand its architecture and capabilities, reducing our learning curve when connecting this local application to the cloud. IGB already contained a localhost REST interface that we could repurpose for BioViz *Connect*, an endpoint was first developed in the early 2000s to enable a connection between the Affymetrix NetAffx Web site and IGB. Since then, we used this same endpoint to implement IGB's internal region and data bookmarking system. IGV, the only other native genome browser application in wide use, has a similar REST endpoint used to trigger loading of data files from the Galaxy Web site and others, but this endpoint lacks features such as the ability to specify track appearance. The third reason was that the IGB interface decouples navigation and data loading, thus making it easier for users to control when data are requested from the remote host. We surmised that this would make possible delays in data loading less onerous than for other browsers, such as IGV (Robinson et al., 2011), UCSC Genome Browser (Kent et al., 2002), Jbrowse (Buels et al., 2016), and Ensembl (Howe et al., 2021), all of which load data automatically when users navigate to a new region. However, since we first released BioViz *Connect*, the CyVerse development team have improved data throughput, making those tools' design less problematic.

Our success in linking IGB to the cloud, along with the abovementioned improvement in CyVerse infrastructure, suggests an interesting next step for BioViz *Connect*: adding other genome browser systems to the interface. Anticipating this possibility, the first column in the BioViz *Connect* file browser table is labeled "Visualization Tools," a generic heading that suggests adding other tools. Doing this would be valuable because although genome browsers often recapitulate each other's features, all have capabilities unique to them, and users who prefer them. For example, IGB offers fast navigation through a genome, the ability to interact directly with data, access to shared data *via* IGB Quickload sites, and visual analytics functions called "Operations" that aid exploratory analysis. Unique features of the Broad Institute's IGV include a sashimi

plot view for detecting differential splicing (Katz et al., 2015) and a bisulfite sequencing view for understanding DNA methylation. The UCSC Genome Browser excels at offering a multitude of data sets in distinct tracks, while the Ensembl browser and associated informatics system famously support nearly every reference assembly known to science, including many plant genomes not supported by UCSC. And the Jalview system provides a host of features for examining the deep details of alignments, the heart of genomic analysis (Procter et al., 2021). BioViz *Connect* could make these systems easier to use and compare, allowing us to study how different approaches to visualization affect understanding.

To our knowledge, BioViz *Connect* is the first application developed using the Terrain API by a group outside the CyVerse development team. Because our work is open source, developed entirely in public, other groups can use our implementation as a guide or inspiration for their own work. BioViz *Connect* further demonstrates to the larger community of biologists, developers, and funders that modern, feature-rich REST interfaces to powerful computational resources stimulate and enable innovation and progress.

The scaled coverage graphs described in the use case scenario offer a useful, practical example of how remote resources can power interactive visual analytics on the desktop, an idea that has been explored in diverse fields and settings, but not often applied to genome visualization as was done here. The example we presented used a pre-existing algorithm, developed by others, but it shows how developers can harness a more powerful gateway system to develop and deploy all-new interactive genome data visualizations. Offloading compute-intensive visual analytics functions to science gateway systems will likely become more appealing and important as the size and complexity of genomic data continue to increase.

LIMITATIONS AND WAYS TO OVERCOME THEM

However, at least two important technical limitations remain, providing opportunities for future work. The first technical limitation has to do with how data flows from the CyVerse back end data store and into the desktop genome browser application. Integrated Genome Browser as currently implemented can display data from users' CyVerse accounts because the Terrain API can assign publicly accessible URLs to individual data files, which makes them available for visualization but exposes them to everyone on the internet. This problem of public accessibility could perhaps be addressed by adding password protection to these URLs, using Basic Authentication headers defined by the HTTP protocol. IGB already supports logging into password-protected Web servers, and so this solution would require little or no changes on the client side.

Another problem has to do with the data file formats themselves and how they can sometimes expose more information than anticipated. IGB, along with every other genome visualization system we are aware of, uses random access, indexed file formats to retrieve subsets of data corresponding to genomic regions. For example, BAM (binary alignment) files are typically large,

impractical to download in their entirety. The data stored in these files are sorted by genomic location and therefore can be indexed by genomic location. When retrieving data for a desired genomic region, IGB and other programs use the BAM file's index, stored separately in a smaller "bai" file, to look up the range of bytes where those data reside in the target file, and then read and process only the data for that region, ignoring the rest. This idea of mapping genomic coordinates to physical file coordinates has been in heavy use for decades, for as long as IGB has existed. Indeed, the original IGB development team at Affymetrix implemented one of the first indexed file formats, called "bar" for "binary array format", used for storing and accessing data from Affymetrix genome tiling arrays, one of the first technologies invented to survey transcription across an entire genome in an unbiased way. However, in some situations, the index can sometimes serve as a genomic map, providing an overview of an entire dataset that could identify an individual. For example, as shown in (Pedersen et al., 2017), one can use the BAM index to detect chromosome abnormalities from whole genome sequencing data, exposing more information about a person or an experiment than anticipated.

The second technical limitation concerns how to flow data from remote sites, *via* a Web browser, into other programs running natively on the desktop, such as Integrated Genome Browser. Web browser development communities are constantly changing and improving their security models, essential to keeping users and their data safe in an increasingly adversarial and dangerous digital environment. Most Web pages are now loaded over encrypted channels, using HTTPS, the secure version of HTTP, and this includes BioViz *Connect*. This means that the JavaScript code responsible for interacting with IGB's localhost endpoint is also loaded *via* HTTPS. However, when this code interacts with IGB *via* its localhost endpoint, it does so *via* unencrypted HTTP, because there is currently no robust way to support HTTPS for the localhost domain. The Chrome and Firefox browser allow BioViz *Connect* code to access the localhost IGB endpoint using HTTP because the communication channel is limited to the user's own computer, presumed to be secure. The MacOS Safari Web browser does not allow it, however. This means that BioViz *Connect*'s "View in IGB" feature fails for Safari users. We handle this by advising the user to switch to a different browser on MacOS. This issue exemplifies a more general problem with connecting the desktop to the cloud. The methods used to communicate with remote computers are always changing, usually becoming more restrictive, which means that developers need to constantly test, revise, and update their software, more so perhaps than developers who create stand-alone, independent applications that rarely need to interoperate with anything other than the host computer's operating system.

Architectures using Web-based REST APIs may help solve these problems. For example, CyVerse or BioViz *Connect* could add new endpoints that themselves support region-based retrieval of genomic data, as with the XML-based Distributed Annotation Service (Dowell et al., 2001; Jenkinson et al., 2008), the newer JSON-based University of Santa Cruz Genome Informatics REST interface (UCSC, 2021), or the BEACONS network API, which supports multiple layers of user authentication (<https://beacon-project.io/>). Rather than deliver data in new JSON or XML formats that would require modifying the client software, these new endpoints could simply stream the data in their

native formats, requiring minimal or no change to the client software. Another way to achieve this would be to design APIs using the facade design pattern, in which an application translates an incompatible interface to a compatible one, expanding the range of clients able to access a resource. For example, developers could create a novel API that provides all the services required for accessing BAM files and their indexes, by creating and destroying secure URLs as users open and load data file resources during a session. Many variations are possible, and as cloud computing infrastructures become easier and cheaper to build upon, more bioinformatics groups will attempt even more daring and exciting innovations, amplifying their users' ability to investigate biological systems.

Finally, we highlight aspects of the BioViz *Connect* interface and functionality that could be further developed to help users find useful tools and help developers find users for their tools. First, we note that the "View in IGB" button in the BioViz *Connect* table view occupies a column labeled "Visualization Tools," a space where links to other visualization tools could also be added, based on the input data they accept. To make space for these other tools, we could replace the button with an IGB logo, and use tooltips to provide documentation or link to videos describing how to use the tools. Second, we could enhance BioViz *Connect* search capabilities to query MetaData tags or other file properties and attributes. Third, we could collaborate with the CyVerse team and other users to design and implement data registries, which data providers and users could use to publish, publicize, and locate data sets relevant to their work. As we hope the name suggests, BioViz *Connect* will connect researchers with data and tools, and will help tool developers connect with their intended audience, improving scientific practice for everyone.

REFERENCES

- Bailey, T. L. (2011). DREME: Motif Discovery in Transcription Factor ChIP-Seq Data. *Bioinformatics* 27 (12), 1653–1659. doi:10.1093/bioinformatics/btr261
- Buels, R., Yao, E., Diesh, C. M., Hayes, R. D., Munoz-Torres, M., Helt, G., et al. (2016). JBrowse: A Dynamic Web Platform for Genome Visualization and Analysis. *Genome Biol.* 17 (1), 66. doi:10.1186/s13059-016-0924-1
- Devisetty, U. K., Kennedy, K., Sarando, P., Merchant, N., and Lyons, E. (2016). Bringing Your Tools to CyVerse Discovery Environment Using Docker. *F1000Res* 5, 1442. doi:10.12688/f1000research.8935.1
- Dooley, R., Vaughn, M. W., Stanzione, D. C., and Terry, S. (2012). "Software-as-a-Service: The iPlant Foundation API," in 5th IEEE Workshop on Many-Task Computing on Grids and Supercomputers (MTAGS), Salt Lake City, Utah, USA.
- Dowell, R. D., Jokerst, R. M., Day, A., Eddy, S. R., and Stein, L. (2001). The Distributed Annotation System. *BMC Bioinform.* 2, 7. doi:10.1186/1471-2105-2-7
- Freese, N. H., Norris, D. C., and Loraine, A. E. (2016). Integrated Genome Browser: Visual Analytics Platform for Genomics. *Bioinformatics* 32 (14), 2089–2095. doi:10.1093/bioinformatics/btw069
- Giardine, B., Riemer, C., Hardison, R. C., Burhans, R., Elnitski, L., Shah, P., et al. (2005). Galaxy: A Platform for Interactive Large-Scale Genome Analysis. *Genome Res.* 15 (10), 1451–1455. doi:10.1101/gr.4086505
- Goff, S. A., Vaughn, M., McKay, S., Lyons, E., Stapleton, A. E., Gessler, D., et al. (2011). The iPlant Collaborative: Cyberinfrastructure for Plant Biology. *Front. Plant Sci.* 2, 34. doi:10.3389/fpls.2011.00034
- Gulledge, A. A., Roberts, A. D., Vora, H., Patel, K., and Loraine, A. E. (2012). Mining *Arabidopsis thaliana* RNA-Seq Data with Integrated Genome Browser

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: <https://www.ncbi.nlm.nih.gov/sra/?term=PRJNA509437>.

AUTHOR CONTRIBUTIONS

NF and AL conceived of and supervised the project. KR, CK, ST, and PB planned and developed BioViz *Connect*. NF, AL, KR, CK, ST, PB, and CD tested and debugged BioViz *Connect*. NF, KR, CK, and AL wrote the draft manuscript. All authors read and approved the final manuscript.

FUNDING

Research reported in this publication was supported by the National Institute of General Medical Sciences of the National Institutes of Health under award numbers 5R01GM121927 and R35GM139609. Funding was used to plan, design, and develop the software reported in the article.

ACKNOWLEDGMENTS

We thank Paul Sarando, Sarah Roberts, Sriram Srinivasan, Ian McEwen, Ramona Walls, and Reetu Tuteja for their assistance with the Terrain API and publishing CyVerse apps, which was made possible through CyVerse's External Collaborative Partnership program.

- Reveals Stress-Induced Alternative Splicing of the Putative Splicing Regulator SR45a. *Am. J. Bot.* 99 (2), 219–231. doi:10.3732/ajb.1100355
- Gulledge, A. A., Vora, H., Patel, K., and Loraine, A. E. (2014). A Protocol for Visual Analysis of Alternative Splicing in RNA-Seq Data Using Integrated Genome Browser. *Methods Mol. Biol.* 1158, 123–137. doi:10.1007/978-1-4939-0700-7_8
- Howe, K. L., Achuthan, P., Allen, J., Allen, J., Alvarez-Jarreta, J., Amode, M. R., et al. (2021). Ensembl 2021. *Nucleic Acids Res.* 49 (D1), D884–D891. doi:10.1093/nar/gkaa942
- Hubbard, A., Bomhoff, M., and Schmidt, C. J. (2020). fRNAkenseq: A Fully Powered-By-CyVerse Cloud Integrated RNA-Sequencing Analysis Tool. *PeerJ* 8, e8592. doi:10.7717/peerj.8592
- Jenkinson, A. M., Albrecht, M., Birney, E., Blankenburg, H., Down, T., Finn, R. D., et al. (2008). Integrating Biological Data-Tthe Distributed Annotation System. *BMC Bioinforma.* 9 (Suppl. 8), S3. doi:10.1186/1471-2105-9-S8-S3
- Katz, Y., Wang, E. T., Silterra, J., Schwartz, S., Wong, B., Thorvaldsdóttir, H., et al. (2015). Quantitative Visualization of Alternative Exon Expression from RNA-Seq Data. *Bioinformatics* 31 (14), 2400–2402. doi:10.1093/bioinformatics/btv034
- Kent, W. J., Sugnet, C. W., Furey, T. S., Roskin, K. M., Pringle, T. H., Zahler, A. M., et al. (2002). The Human Genome Browser at UCSC. *Genome Res.* 12 (6), 996–1006. doi:10.1101/gr.229102
- Kim, D., Pertea, G., Trapnell, C., Pimentel, H., Kelley, R., and Salzberg, S. L. (2013). TopHat2: Accurate Alignment of Transcriptomes in the Presence of Insertions, Deletions and Gene Fusions. *Genome Biol.* 14 (4), R36. doi:10.1186/gb-2013-14-4-r36
- Leinonen, R., Sugawara, H., and Shumway, M. (2011). The Sequence Read Archive. *Nucleic Acids Res.* 39, D19–D21. doi:10.1093/nar/gkq1019

- Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., et al. (2009). The Sequence Alignment/Map Format and SAMtools. *Bioinformatics* 25 (16), 2078–2079. doi:10.1093/bioinformatics/btp352
- Loraine, A. E., Blakley, I. C., Jagadeesan, S., Harper, J., Miller, G., and Firon, N. (2015). Analysis and Visualization of RNA-Seq Expression Data Using RStudio, Bioconductor, and Integrated Genome Browser. *Methods Mol. Biol.* 1284, 481–501. doi:10.1007/978-1-4939-2444-8_24
- Mall, T., Eckstein, J., Norris, D., Vora, H., Freese, N. H., and Loraine, A. E. (2016). ProtAnnot: An App for Integrated Genome Browser to Display How Alternative Splicing and Transcription Affect Proteins. *Bioinformatics* 32 (16), 2499–2501. doi:10.1093/bioinformatics/btw068
- Merchant, N., Lyons, E., Goff, S., Vaughn, M., Ware, D., Micklos, D., et al. (2016). The iPlant Collaborative: Cyberinfrastructure for Enabling Data to Discovery for the Life Sciences. *PLoS Biol.* 14 (1), e1002342. doi:10.1371/journal.pbio.1002342
- Nicol, J. W., Helt, G. A., Blanchard, S. G., Jr., Raja, A., and Loraine, A. E. (2009). The Integrated Genome Browser: Free Software for Distribution and Exploration of Genome-Scale Datasets. *Bioinformatics* 25 (20), 2730–2731. doi:10.1093/bioinformatics/btp472
- Pedersen, B. S., Collins, R. L., Talkowski, M. E., and Quinlan, A. R. (2017). Indexcov: Fast Coverage Quality Control for Whole-Genome Sequencing. *Gigascience* 6 (11), 1–6. doi:10.1093/gigascience/gix090
- Procter, J. B., Carstairs, G. M., Soares, B., Mourão, K., Ofoegbu, T. C., Barton, D., et al. (2021). Alignment of Biological Sequences with Jalview. *Methods Mol. Biol.* 2231, 203–224. doi:10.1007/978-1-0716-1036-7_13
- Ramírez, F., Ryan, D. P., Grüning, B., Bhardwaj, V., Kilpert, F., Richter, A. S., et al. (2016). deepTools2: a Next Generation Web Server for Deep-Sequencing Data Analysis. *Nucleic Acids Res.* 44 (W1), W160–W165. doi:10.1093/nar/gkw257
- Raney, B. J., Dreszer, T. R., Barber, G. P., Clawson, H., Fujita, P. A., Wang, T., et al. (2014). Track Data Hubs Enable Visualization of User-Defined Genome-Wide Annotations on the UCSC Genome Browser. *Bioinformatics* 30 (7), 1003–1005. doi:10.1093/bioinformatics/btt637
- Robinson, J. T., Thorvaldsdóttir, H., Winckler, W., Guttman, M., Lander, E. S., Getz, G., et al. (2011). Integrative Genomics Viewer. *Nat. Biotechnol.* 29 (1), 24–26. doi:10.1038/nbt.1754
- UCSC (2021). REST API Data Interface [Online]. Available: <https://genome.ucsc.edu/goldenPath/help/api.html> (Accessed August 20, 2021).
- Wang, L., Lu, Z., Van Buren, P., and Ware, D. (2018). SciApps: a Cloud-Based Platform for Reproducible Bioinformatics Workflows. *Bioinformatics* 34 (22), 3917–3920. doi:10.1093/bioinformatics/bty439
- Wilkins-Diehr, N., Gannon, D., Klimeck, G., Oster, S., and Pamidighantam, S. (2008). TeraGrid Science Gateways and Their Impact on Science. *Computer* 41 (11), 32–41. doi:10.1109/MC.2008.470
- Yoshimura, K., Mori, T., Yokoyama, K., Koike, Y., Tanabe, N., Sato, N., et al. (2011). Identification of Alternative Splicing Events Regulated by an Arabidopsis Serine/arginine-Like Protein, atSR45a, in Response to High-Light Stress Using a Tiling Array. *Plant Cell Physiol.* 52 (10), 1786–1805. doi:10.1093/pcp/pcr115

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Raveendran, Freese, Kintali, Tiwari, Bole, Dias and Loraine. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



ShapoGraphy: A User-Friendly Web Application for Creating Bespoke and Intuitive Visualisation of Biomedical Data

Muhammed Khawatmi, Yoann Steux, Saddam Zourob and Heba Z. Sailem*

Institute of Biomedical Engineering, Department of Engineering, University of Oxford, Oxford, United Kingdom

OPEN ACCESS

Edited by:

Sean O'Donoghue,
Garvan Institute of Medical Research,
Australia

Reviewed by:

Anamaria Crisan,
Salesforce, United States
Jan Aerts,
Amador Bioscience, Belgium

*Correspondence:

Heba Z. Sailem
heba.sailem@eng.ox.ac.uk

Specialty section:

This article was submitted to
Data Visualization,
a section of the journal
Frontiers in Bioinformatics

Received: 02 October 2021

Accepted: 23 May 2022

Published: 04 July 2022

Citation:

Khawatmi M, Steux Y, Zourob S and
Sailem HZ (2022) ShapoGraphy: A
User-Friendly Web Application for
Creating Bespoke and Intuitive
Visualisation of Biomedical Data.
Front. Bioinform. 2:788607.
doi: 10.3389/fbinf.2022.788607

Effective visualisation of quantitative microscopy data is crucial for interpreting and discovering new patterns from complex bioimage data. Existing visualisation approaches, such as bar charts, scatter plots and heat maps, do not accommodate the complexity of visual information present in microscopy data. Here we develop ShapoGraphy, a first of its kind method accompanied by an interactive web-based application for creating customisable quantitative pictorial representations to facilitate the understanding and analysis of image datasets (www.shapography.com). ShapoGraphy enables the user to create a structure of interest as a set of shapes. Each shape can encode different variables that are mapped to the shape dimensions, colours, symbols, or outline. We illustrate the utility of ShapoGraphy using various image data, including high dimensional multiplexed data. Our results show that ShapoGraphy allows a better understanding of cellular phenotypes and relationships between variables. In conclusion, ShapoGraphy supports scientific discovery and communication by providing a rich vocabulary to create engaging and intuitive representations of diverse data types.

Keywords: microscopy, multiplexed imaging, morphology, glyph-based visualisation, high dimensional data, graph editor, single cell data, science communication

1 INTRODUCTION

Biomedical imaging generates large amounts of data capturing biological systems at different scales ranging from single molecules to organs and organisms (Walter et al., 2010). Inspection of individual images is not feasible when hundreds of images are acquired, particularly when they are composed of multiple layers, channels, or planes. Automated image analysis allows quantifying image data resulting in large multiparametric datasets (Sero et al., 2015; Natrajan et al., 2016). Effective data visualisation is essential for interpreting analysis results and unleashing the hidden patterns locked in image data (Heer et al., 2010; Cairo, 2013).

Intuitive representations can improve the effectiveness of visualisation tools as they support identifying and understanding the complex relationships in image data. By intuitive we mean that the depicted representations are semantically relevant where the used visual channel resembles the concept or the represented phenotypic feature. For example, it is easier to associate measurements of cell size to the size of the object and the protein levels to the colour of the object. This has many advantages especially when multiple variables are plotted simultaneously. First, the pictorial representation facilitates remembering and interpreting the data. Second, the natural mapping

between the measured objects and the representation makes it easier to investigate the relationship between the measured variables.

Visualising complex imaging data has been mostly limited to general-purpose tools that do not take into account the structural nature of image data. Due to their scalability to a large number of data points, heat maps and dimensionality reduction, such as UMAPs and t-SNE, are the most used approaches for visualising high dimensional data, including image-based measurements (McInnes et al., 2018). Several methods have been developed for visualising bioimage data with an emphasis on interactive linkage of raw image data, cell features, and identified quantitative phenotypes using linked scatter plots combined with supervised and unsupervised learning approaches including t-SNE plots. These include Facetto, histoCAT, and mineotaur (Antal et al., 2015; Schapiro et al., 2017; Krueger et al., 2020). ImaCytE (Somarakis et al., 2019) is another tool for visualising multiplexed image cytometry data that takes the interactive aspect a step further by developing custom two-layered pie charts to represent the proportion of different phenotypes. While these tools are useful in interactive and data exploration tasks, they heavily rely on the user interpretation of identified phenotypes based on the appearance of a handful of cells which can be a subjective and daunting task. Therefore, new visualisation techniques for representing multiparametric image data are desperately needed to aid data analysis and result interpretation.

Glyph-based visualisation is another approach to visual design where quantitative information is mapped to illustrative graphics referred to as glyphs. They provide a flexible way of representing multidimensional data (Ropinski et al., 2011; Borgo et al., 2013; Fuchs et al., 2017). For example, we have previously developed PhenoPlot, a glyph-based visualisation approach that plots cell shape data as cell-like glyphs (Sailem et al., 2015). PhenoPlot was built as is a MatLab toolbox and incorporates two ellipsoid glyphs to represent the cell and nucleus. It uses a variety of visual elements such as stroke, colour and symbols to encode up to 21 variables. The key focus of PhenoPlot is to allow for natural data mapping by selecting graphic features that resemble data attributes. For instance, the extent that a jagged border around the cell ellipse can be used to represent the irregularity of cell shape, and the proportion of “x” symbols filling the cell ellipse can be mapped to endosome abundance. However, the shape configuration in PhenoPlot is limited to two ellipse-shaped objects and the feature mapping is hard-coded which does not accommodate the diversity of biomedical images data.

To support knowledge discovery tasks from microscopy data, we propose a new framework for creating glyph-based representations by combining geometrical shapes that can systematically encode several predefined visual elements. We implemented this framework as a user-friendly web interface that can automatically and swiftly map data to the created glyph representations. To our knowledge, ShapoGraphy is the first method that allows creating new glyph-based visualisation by combining different shaped objects and custom mapping of their properties, such as colour, symbols, stroke, and dimensions, to data attributes. The user can choose from a basic set of shapes or

draw their own. The effectiveness and utility of ShapoGraphy are illustrated by using various image datasets where we show that it facilitates the understanding of cellular phenotypes and interactive exploration of the data. This includes multiplexed image data where single cell activities of tens of proteins are measured simultaneously. In summary, ShapoGraphy allows the users to construct an infinite number of glyph-based representations in order to generate a quantitative and intuitive visualisation to aid pattern recognition from multiparametric data.

2 METHODS

2.1 Design and Concept of ShapoGraphy

To generate a quantitative pictorial representation of phenotypic data we created ShapoGraphy; a user-friendly web application (Figures 1A–D, 2A). ShapoGraphy maps data to visual properties of shapes where multiple shapes can be combined to define a biological structure. For example, a squared-shaped object can be used to represent cell context, epithelial cell shape can be represented using a square for the cell body and a circle for the nucleus. We call such a configuration a template and provide multiple templates to represent a variety of microscopy data. A new template can be created by combining different shapes. The users have the option of selecting from a collection of predefined geometrical shapes or drawing their own. For example, the user can draw a cell or organ shape. The objects can be positioned relative to each other to create the desired structure (Figure 1D). ShapoGraphy is highly customisable where the property of any object in the template, such as colour, size or opacity, can be changed.

We developed various encodings that allow mapping continuous quantitative data to shapes by using different visual elements (Figure 1C). These include dimensions, size, and colour that are commonly used for visualising data. For the fill gradient element, we employed well-established colour maps from ColorBrewer (Brewer, 2022). We have previously proposed novel visual elements, such as partial overlaying the object outline or filling the object with symbols proportional to the variable value (Sailem et al., 2015). We introduce new features in ShapoGraphy, such as the mesh density (horizontal, vertical or grid), opacity, and rotation angle (Figure 1C). The use of various glyph shapes, positions and visual elements allows designing abstract and intuitive representations of a broad range of structures investigated in biomedical imaging to assist in understanding, summarising, and communicating results (Figure 1D). This type of design gives the user high flexibility when it comes to constructing new visual encodings that are more intuitive and engaging.

2.2 ShapoGraphy User Interface

We adopted a modular design that resembles other graphic design software such as Adobe Illustrator. Data import, saving results, figure export and other auxiliary functionalities such as viewing the data in a heat map or t-SNE plots are available from the top menu (Figure 2). Once a dataset is uploaded, the user can

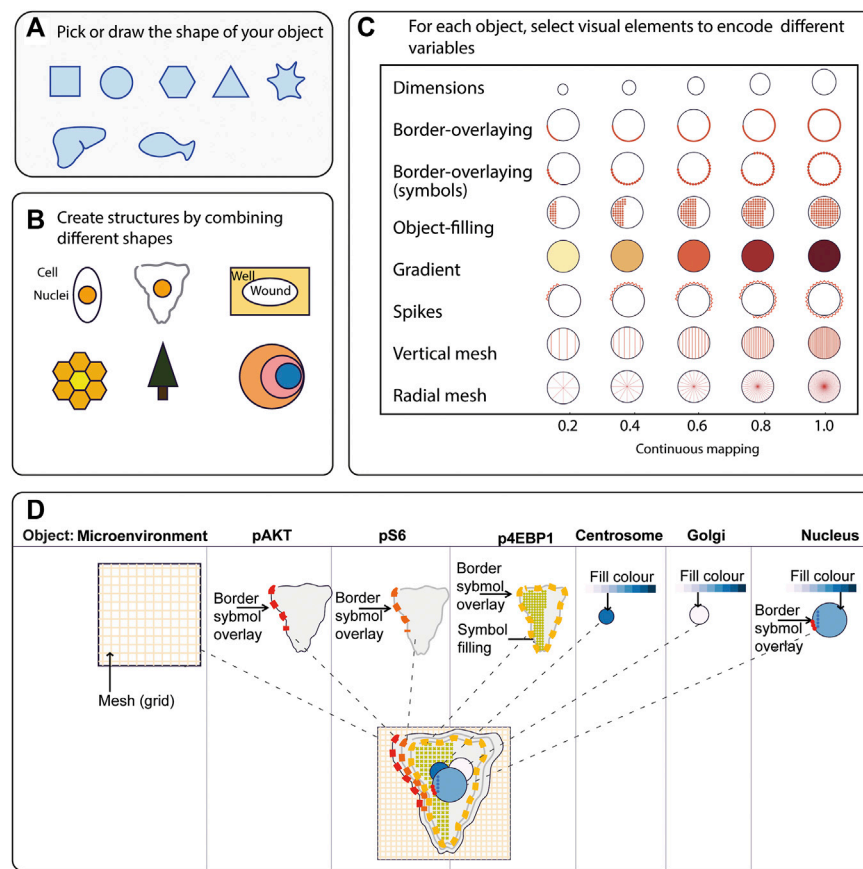


FIGURE 1 | ShapoGraphy provides a highly flexible framework for creating glyph-based visualisations. **(A)** Example of object shapes that can be created using ShapoGraphy. **(B)** Shapes can be combined to create structures that resemble the measured phenomena. **(C)** Various visual elements are defined for each object and can be selected by the user to encode several variables. **(D)** An example of how objects can be combined to represent a wide range of phenotypic information.

add various shapes from the left menu. This includes drawing a custom shape using the “draw shape” icon which opens a small canvas that the user can draw on. For this option, the user needs to draw the shape in one stroke as many elements, such as border symbols or overlay, will be mapped to the object outline. A list of the added shapes will appear on the right menu. The user can modify the name of each object using the pencil icon at the bottom of the objects list so that they can be easily identified. The user can also duplicate an object which can be useful to generate a new object with exact feature mapping or when a custom shape is used. The objects are laid on top of each other as layers. The object layer order can be modified using the upward and downward arrows on the left of the object name. For example, the nucleus should be positioned after the cell object, as it will be concealed otherwise. The object location can be changed from the Global Features sub-menu or by dragging and dropping the object in the canvas.

For each object, we recommend selecting visual channels in such a way that they metaphorically resemble the measured concepts. Different symbols can also be used to distinguish different variables. The user can customise the visual appearance of these channels and the variables that are bound

to them from the Data Mapping sub-menu. For example, for “Symbol filling” or “Border symbol” elements, the user can choose from the following symbols: { \times , *, -, \bullet , \square , \square } and specify their colour and size (**Table 1**). For the Mesh element, the user can choose vertical, horizontal, radial, grid-like or randomly oriented mesh (**Figure 1C**). The user can also specify the stroke size of the mesh and the colour of the mesh lines.

To facilitate the exploration of design space in ShapoGraphy, we offer a hide/show functionality of each of the objects or data-symbol mappings through the eye icon on the left of each object or element. We found this functionality very useful when assessing interactions between objects, decluttering the representation or determining relevant features.

On the right menu, there are also options for data normalisation which is discussed in **Section 2.4** and positional mapping of Shape Glyphs in 2D dimensional space.

We employ pagination to deal with a large number of data points. The user has the option to display more objects on the same page or browse them in multiple pages. This can be useful if combined with sorting functionality in the Positional Mapping sub-menu.

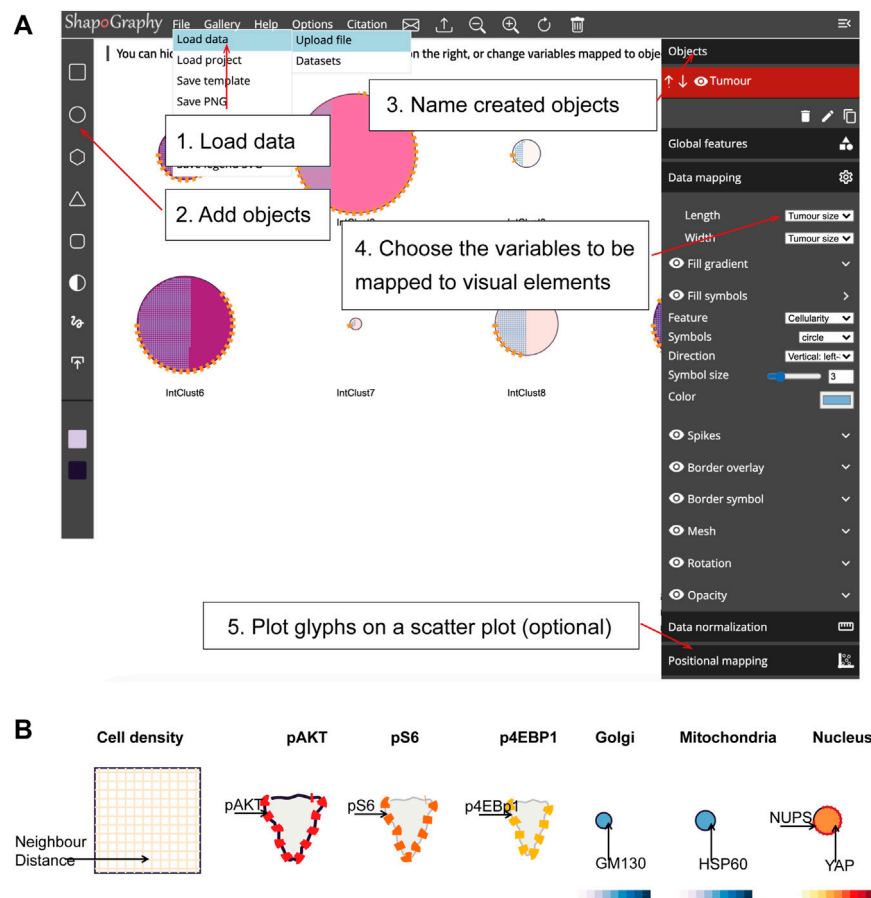


FIGURE 2 | ShapoGraphy user interface. **(A)** ShapoGraphy allows users to interactively construct and customise their plots using a flexible graphical user interface. The user 1) uploads the data from the file menu 2) creates objects 3) customises their properties 4) maps the selected object properties to the variables in the dataset. Positional mapping can be used to position the created objects in a scatter plot based on selected data variables. **(B)** Legend is generated automatically by ShapoGraphy where different objects are shown separately and variables mapped to the different visual elements for each object are labelled. Objects names chosen by the user are shown in bold. All other labels are the variable names that are mapped to the object properties or depicted marks.

2.3 Legend

The legend can be viewed from the top menu. Creating a legend for the resulting composite glyph can be challenging as we do not know in advance which objects or elements will be used and how they will overlap. We therefore employed a simple object-oriented strategy where we plot each object separately and automatically determine non-overlapping locations to label the used visual elements (**Figure 2B**). Long variable names are truncated and are displayed as a tooltip if the user hovers over them. An alternative option for generating a legend is manual labelling of one of the generated Glyph Shapes as we did for **Figures 3, 4**.

2.4 Data Normalisation

Like heat maps and other glyph-based approaches, our method requires normalising the data between 0 and 1 so they are mapped to the same scale (Sailem et al., 2015). If the uploaded data is not normalised, then it is automatically scaled. We note that some variables can be related (represent the same scale). For example, if the width and length of an object were scaled independently, their relative ratio will not provide a faithful representation of the

actual data. To tackle this problem, we introduce linked variable functionality in the Data Normalisation sub-menu on the right. Linked variables are mapped to the same scale. For instance, if the length of the largest cell is 100 pixels and its width is 60 pixels, then they will be scaled to 1 and 0.6 respectively when defined as linked variables but to 1 and 1 when scaled independently (assuming that this cell is also the widest cell).

2.5 Implementation

ShapoGraphy is developed using HTML5 and JavaScript. The shapes and their customisation are implemented using paper.js library. It is a client-side web application which means that all the processing happens at the user end and minimal data is uploaded to our server. This circumvents potential privacy issues.

We defined a portfolio of templates to accommodate different data (**Figures 3, 4** and **Supplementary Figures S1, S2**). The user can choose an existing template to map their data or modify an existing template by adding additional objects and changing shape-data mapping. They can also delete or hide unwanted objects for maximal flexibility.

TABLE 1 | Customisable properties of ShapoGraphy elements.

Visual element	Static properties
Length	No additional properties
Width	No additional properties
Fill gradient	Colour map
Fill symbols	Symbol: { X, *, -, •, □, □ }
	Fill direction: left- > right, right- > left, top- > bottom, bottom- > top
	Symbol colour
	Symbol size
Spikes	Stroke size
	Spike density
	Colour
Border overlay	Stroke size
	Stroke colour
Border symbol	Symbol: { X, *, -, •, □, □ }
	Symbol colour
	Symbol Size
Mesh	Orientation {vertical, horizontal, radial, grid, random}
	Colour
	Stroke size
Rotation	No additional properties
Opacity	No additional properties

2.6 Import and Export

We offer multiple options for exporting visualisation created in ShapoGraphy including Portable Graphics Format (PNG) or Scalable Vector Graphic (SVG). The latter is particularly useful if the user needs to tweak the design in a graphic editors. The user can export their template which will be saved as a JavaScript Object Notation (JSON) file. This can be then imported using the “Load Project” function from the File menu.

The File menu on top left allows the user to upload data, load demo data or load a project (data file and previously saved templates). If the variable names in the template and variable names in the data file do not match, then the user can remap these variables from the right menu.

2.7 Datasets

The datasets used in this manuscript are available as demo files from the file menu in ShapoGraphy.

2.7.1 Wound Scratch Data

Wound scratch data was obtained from an image-based siRNA screen measuring human dermal lymphatic endothelial cells migration into a scratch wound created in a cell monolayer²⁰. Cells were imaged at 0 and 24 h following wounding at 4x objective. Cells were detected and the wound area was segmented using DeepScratch¹⁵. Measurements of wound size and cell numbers at 24 h were normalised to timepoint 0 h and represented using ShapoGraphy.

2.7.2 Multiplexed Imaging Data

Multiplexed imaging data of 2000 HeLa cells was obtained from Gut et al. (2018) where immunofluorescence of different markers was performed in cycles to image the subcellular localisation of 40 proteins¹⁶. Ten variables were selected to showcase ShapoGraphy. Data was scaled and transformed using UMAP. K-means was

used to group phenotypically similar cells into six clusters. The average of UMAP dimension 1 and 2 was calculated for each cluster.

Three cell-shaped objects were created to represent PI3K/AKT/mTOR pathway (pAKT, p4EBp1 and pS6, where “p” denote protein phosphorylation) on the cell periphery as the proportion of symbols overlaid on the object outline (**Figure 4C**). The grid density in the square surrounding the cell object represents the local cell density. The abundance of late endosomes (CAV1) was represented as “x” symbols filling the cytosol. Golgi and centrosome organelles were abstracted as circles with a colour gradient reflecting their abundance. Three variables were mapped to the circle-shaped nucleus object: the value of nuclear pore protein (NUPS) was mapped to the border of the nucleus object, the level of YAP transcription factor was mapped to the colour of the nucleus object, and the abundance of cell proliferation protein PCNA was represented as dots filling the nucleus object. The position of each Shape Glyph is mapped to the cluster centre using the Positional Mapping sub-menu.

3 RESULTS

3.1 Case Studies

We created various templates to represent diverse image datasets. These include phenotypic data of breast tumours based on METABRIC study (Curtis et al., 2012) and cell shape data from our PhenoPlot study (**Supplementary Figures S1, S2**). Here, we discuss in detail the application of ShapoGraphy to multiplexed and wound healing data. Notably all these templates can also be used with any numerical data.

3.1.1 Visualising Scratch Assays Data

As a first use case, we used ShapoGraphy to visualise the effect of gene perturbations on cell migration into a wound scratch

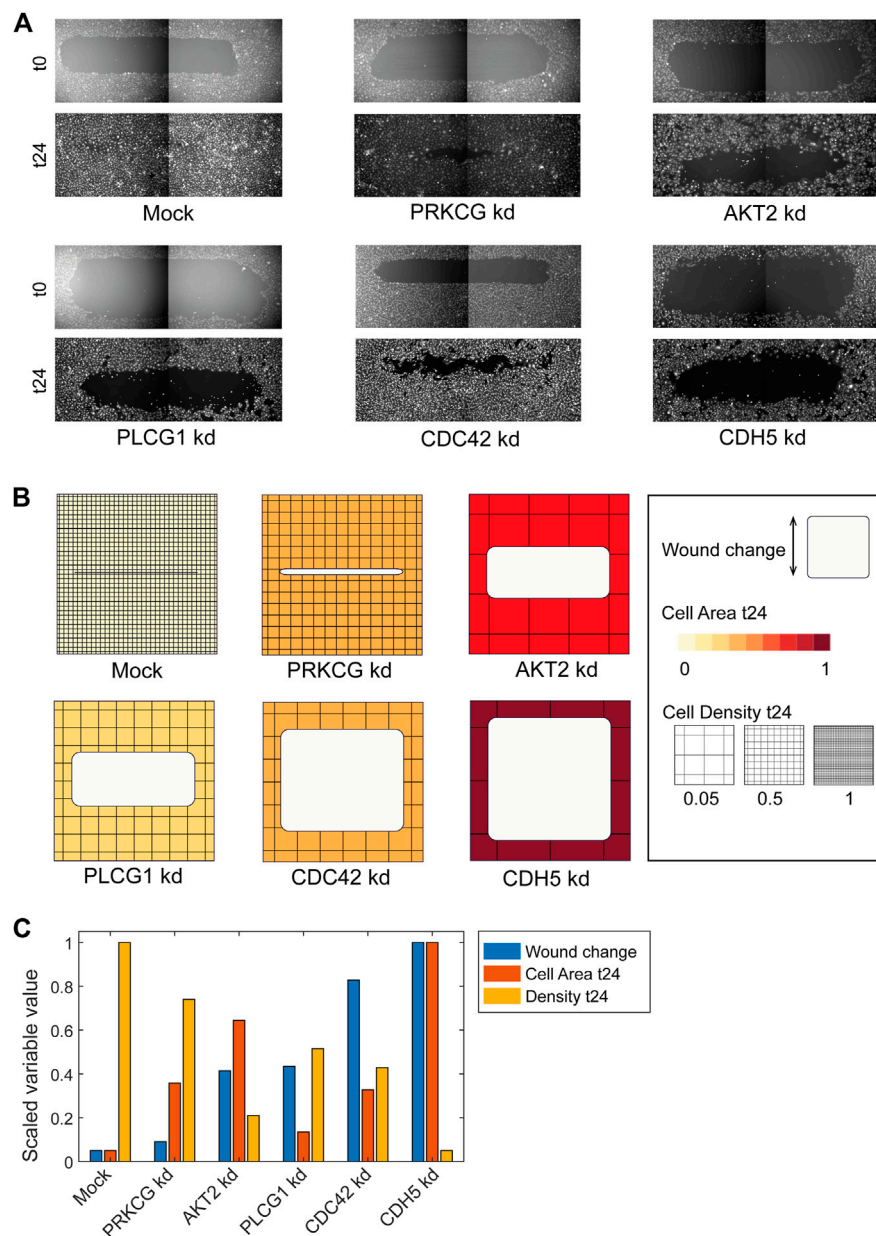
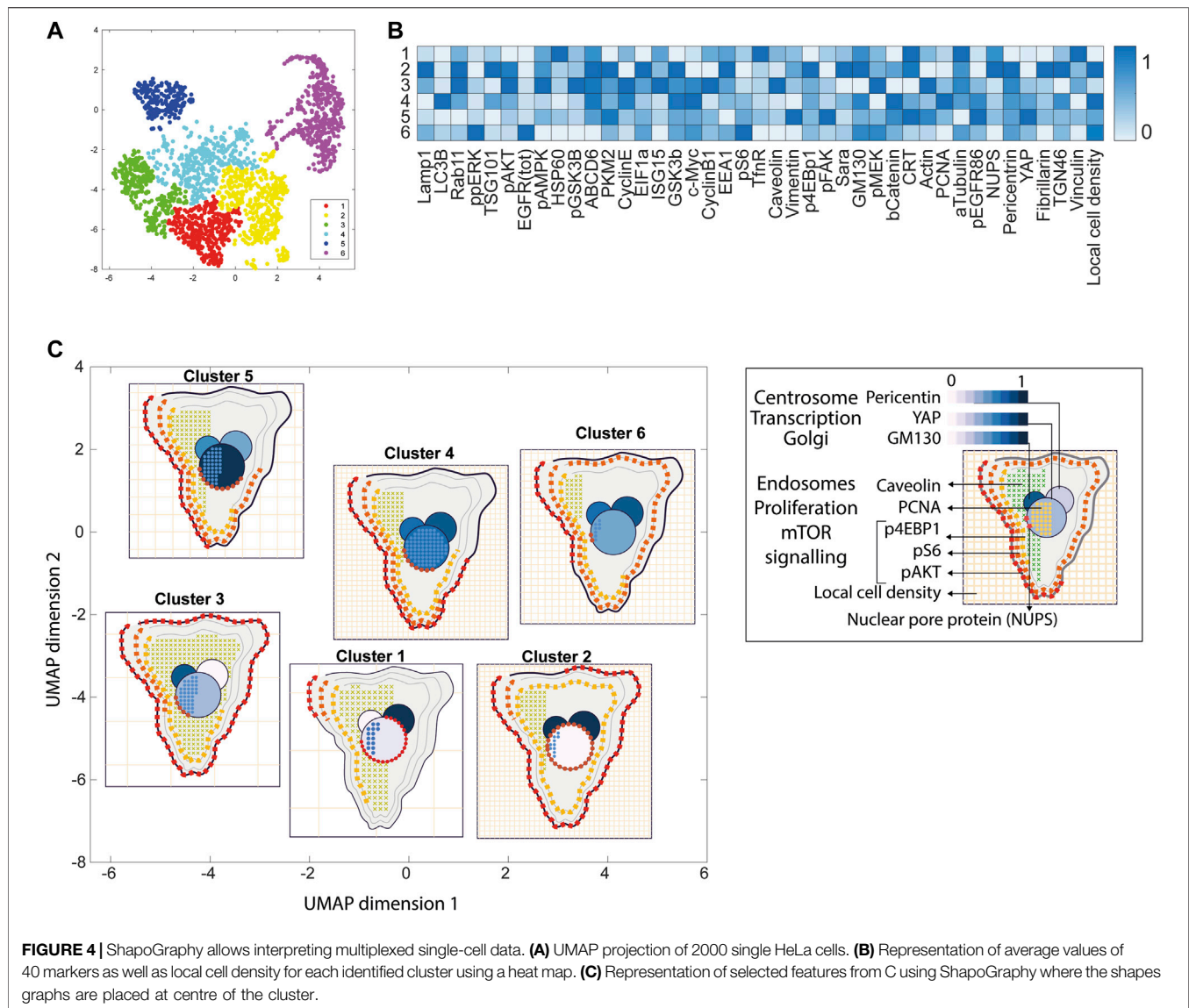


FIGURE 3 | Using ShapoGraphy to represent wound healing data. **(A)** Image data capturing the effect of various gene depletions on human lymphatic endothelial cells ability to migrate into scratch wounds [time-point 0h (t0) and 24h (t24)]. **(B)** Intuitive representation of wound area and cell number measurements using ShapoGraphy based on data in **(A)**. The outer square represents the well where lighter red hues indicate lower cell area while higher red hues indicate higher cell area. Cell number is mapped to grid density. The height of the inner square represents the normalised change in wound area. **(C)** Representation of the same data in **(B)** using a bar chart where numerical data are mapped to the bars' length

(Javer et al., 2020). In this dataset, the closure of an artificially made wound by human lymphatic endothelial cells is measured over a period of 24 h to determine how different gene knockdowns, using siRNA, affect cell migration (**Figure 3A**). In addition to the change in wound area, we measured the number and area of cells as they can affect the final wound area.

To represent this data using ShapoGraphy, the well and the wound were depicted as rectangles mimicking the shape of the

actual measured data. We chose to represent the cell area using the colour of the well object because it applies to most of the cells. We mapped the density of the cells to a mesh density element because they represent a similar concept, i.e., density, and therefore are easier to link. The height of the wound object represents the change in wound area which naturally corresponds to the healing process where cells migrate vertically to close the created wound (**Figure 3B**). Compared to a bar chart (**Figure 3C**), such representation reveals more

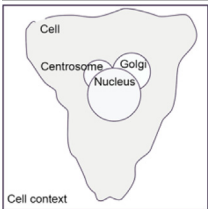
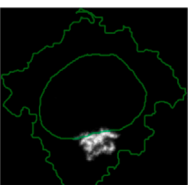
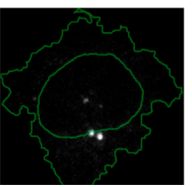
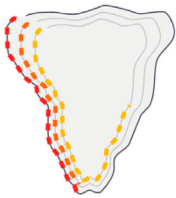
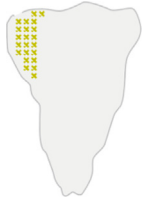

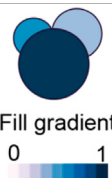
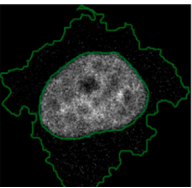
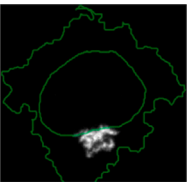
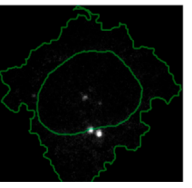


readily that depletion of AKT2 and PLCG1 genes results in a similar wound area and that AKT2 knockdown results in lower cell density and higher cell area than PLCG1. Therefore, their effects on cell motility are not equal. Similarly, depleting CDH5 and CDC42 significantly affects wound area, but CDH5 knockdown results in significantly lower cell number and very large cells suggesting that these two genes affect cell motility through different mechanisms (**Figure 3B**). This pattern is difficult to discern from raw images as wound measurements need to be normalised to the initial timepoint (0 h) (**Figure 3A**). A bar chart of these three variables, on the other hand, does not allow for metaphoric association between these variables making it difficult to identify the relationships between them. These results show that ShapoGraphy allows identifying interactions between variables as it provides a more intuitive representation which supports making scientific conclusions from complex phenotypic data.

3.1.2 Visualisation of Multiplexed Imaging Data

Next, ShapoGraphy was used to obtain high data density of single cell phenotypes in multivariate multiplexed imaging data measuring 40 markers (Gut et al., 2018). Multiplexed imaging allows simultaneous imaging of spatial protein activities, subcellular organisation as well as various cell identities (Zhang et al., 2013). Since tens of markers can be imaged, colour coding of the different proteins is no longer useful to visualise this information (Walter et al., 2010). To study the phenotypic heterogeneity of cancer HeLa cells, we analysed data from 2000 cells that were stained with markers highlighting various cellular organelles and signalling components including the AKT pathway (Methods). Using k-means and UMAP cells could be clustered to characterise different subpopulations but the specifics of the underlying phenotypic differences between the clusters could not be obtained (**Figure 4A** and Methods). Heat maps allow studying all the measured

TABLE 2 | List of design decisions (objects and visual elements) used in **Figure 3**.

Design	Description		
	<p>For intuitive mapping of multiplexed image data, we used different objects to create a hierarchy and represent features associated with different cellular compartments</p> <p>Golgi (GM130)</p>	<p>Centrosomes (Pericentrin)</p>	 
	<p>Signalling of AKT is represented as symbols overlaid on the cell object outline. Three cell-shaped objects are layered to represent additional information at the cell periphery. This configuration allows representing the signalling cascade pAKT -> p4EBP1 and pS6. Different colours are used for these different proteins so they can be distinguished easily</p>		
	<p>Endosome abundance, based on CAV1, is represented as symbols filling the inner cell-shaped object. This visual channel is well-suited to represent the punctate distribution of endosomes in the cell</p> <p>Caveolin (CAV1)</p>		
	<p>Multiple variables are mapped to the nucleus object. The border symbol (red dots overlying nucleus glyph) provides a faithful representation of nuclear pore protein (NUPS) that localises to the nucleus membrane. The nucleus colour is used to represent the level of YAP transcription factor. While the cell proliferation protein PCNA is represented using symbol filling due to its punctate appearance (blue dots)</p> <p>NUPS</p>	<p>PCNA</p>	<p>YAP</p>
	<p>We used colour gradient in a manner similar to a heat map to represent the value of proteins that localise to different organelles. For example, YAP transcription factor is mapped to the colour of the nucleus glyph and Pericentrin is mapped to the colour of the centrosome glyph where they localise. The same colour map is used to enable comparison. The colour provides a good choice when the objects are overlapping, and part of the object is concealed as it is uniform throughout the object</p>		
	<p>YAP (nuclear)</p>	<p>Golgi (GM130)</p>	<p>Centrosomes</p>
			

markers individually but require many cognitive calculations such as searching for the different variables and remembering their values to compare them (**Figure 4B**). This makes them challenging to interpret.

In order to facilitate the understanding of single cell phenotypes that are derived from multiplexed data, ShapoGraphy was used to design a template where the visual elements resemble the represented data attributes. We

combined several objects to create a structure that mimics the measured data and depicts the hierarchical nature of bioimage data (**Table 2**). For example, as cells are composed of multiple organelles, we used different circled objects inside the cell object to represent data of proteins localised to different organelles: nucleus, Golgi and centrosomes. On the cell object, we represented the AKT signalling cascade as consecutive layers on the cell periphery. We created a square around the cell object to represent its context based on local cell density. As in the first use case, we mapped the cell density to the mesh density as they can be easily associated. Symbol filling is well suited for representing endosomal abundance because of its punctate distribution in the cytosol. The rationale for the different design choices is explained in **Table 2**. This abstract representation of different components in the cell and their spatial arrangement provides a more intuitive representation where the various elements in the Shape Glyph can be easily linked to the measured variables.

A major advantage of using glyph representations is that the quantitative information is self-contained and therefore the position channel can be used to visualise additional dimensions. We positioned the composite glyphs based on the centre of identified clusters in the reduced UMAP space to help sorting these composite glyphs and comparing cluster phenotypes (Methods).

Figure 4C shows that Cluster 2, 4, and 6 on the right have high cell density (grid density) and low late endosome abundance (x symbols filling the cytosol). Cluster 6 and 4 are highly similar, but Cluster 6 has the highest pS6 levels across all clusters, while Cluster 2 has very high pAKT and p4EBp1, centrosomes (Pericentrin), nuclear pore proteins (NUPS), but low YAP values. Cluster 3 has also high pAKT and p4EBp1 like Cluster 2 but has lower cell density and the highest endosome abundance. Discussing our results with biologists, they found that these representations help them understand their data better as it is easier to identify and relate the differences between clusters to image data. In comparison, **Figure 4B** depicts the same information in a heat map which can complement our Shape Glyphs but does not help the user to build a mental picture of the data. Therefore, ShapoGraphy provides a more expressive representation of phenotypic classes and their biological relevance based on high dimensional single-cell data which allows scientists to uncover and study complex patterns and relationships in the data.

3.2 Guidelines for Designing Glyph-Based Representations Using ShapoGraphy

We reflect on our learning from developing various use cases using ShapoGraphy and our discussions with potential users. First, while the motivation of combining different objects is to create semantically relevant representations, it is possible that some object and/or element combinations can be perceived differently from what is intended or can result in undesirable properties. For instance, using a mesh element on a hierarchy of circles can create geometric patterns (**Supplementary Figure S4**). Here we propose that ShapoGraphy provides a fast approach for

assessing such interactions. Moreover, it allows experimenting with various designs that can inspire new visual representations.

We noticed that when creating composite glyphs, users tried to infer meaning from aspects of the element configuration which were not mapped to data as the user was looking for patterns in the plotted glyphs. This was the case when using the mesh element with random orientation. This problem did not arise when the user learned that this is a static configuration. As object colour can be either statically defined or dynamically mapped to the variable, we recommend using it consistently for all objects. For example, the coloured objects in **Figure 4C** (Golgi, centrosome, and nucleus) reflect the variable value and the same colour is used otherwise. We also experimented with assigning the same colour for all symbols/elements, however some users found this representation difficult to scan and using different colours helped the user in distinguishing and scanning these distinct elements (**Supplementary Figure S3**). Continuing the discussion of colour assignment, we found that using the same colour map for “Fill gradient” element is important to make comparisons across different objects easier.

Consideration should be given to the number of features when using Shape Glyphs as our working mental memory is limited and can handle only 5–10 variables at a time (Cairo, 2013). Selection of important features can be achieved through interactive exploration in ShapoGraphy and using the hide/show functionality to identify the most relevant information to be communicated to the reader.

Object occlusion is another aspect that needs to be considered when designing Shape Glyphs where objects are overlayed on top of each other or partially overlap. Visual elements such as colour and mesh density are less affected when part of the object is occluded. For example, the nucleus object lies on the top and occlude part of the Golgi and centrosome objects in **Figure 4C**, but does not affect the perceived quantitative mapping as colour is uniform throughout the object.

4 DISCUSSION

The human brain perceives information by converting visual stimuli to symbolic representations that are then interpreted based on our memories and previous knowledge. Visualisation approaches help our brain create a mental visual image of quantitative data in order to recognise patterns and identify interesting relationships that might be missed otherwise (Tufte, 2001). ShapoGraphy is a new visualisation approach that allows creating bespoke glyph-based representations by constructing composite glyphs that combine different shapes and symbols, each of which encodes multiple variables. To our knowledge, such an approach to data visualisation has not been explicitly proposed before and no tool is available to create such graphical representations automatically.

The main advantage of ShapoGraphy is that it enables the creation of a metaphoric quantitative representation of the data to aid the reader in interpreting, understanding, and communicating scientific results. This makes it perfectly suited for bioimages because of the structural and hierarchical nature of

these datasets. Nonetheless, ShapoGraphy is a very versatile tool and can be applied to any numerical data such as single cell RNA sequencing, proteomics, or non biological data. Another advantage of Shape Glyphs is that such pictorial representations can attract more attention from the reader as they stimulate more cognitive activity (Borgo et al., 2013). This can be beneficial when communicating data with a broad audience. Therefore, ShapoGraphy serves as a general-purpose methodology for creating more engaging and intuitive graphic representations.

ShapoGraphy complements existing visualisation methods such as heat maps, t-SNE and UMAPs. While the latter approaches provide a global picture of the major trends or structure in the data, ShapoGraphy allows a more detailed understanding of multiparametric phenotypes. It aims to represent quantitative data so the user can compare different variable values relative to each other, rather than generating an actual picture of the image data. Such distinction is necessary as image data are often normalised which make interpreting raw image data more challenging and subjective. Currently, our approach is best suited for summarising and providing higher information density of major phenotypes in the data, rather than individual data points. This is because the pictorial nature of the generated representations requires high resolution and more space. These phenotypes can be identified using clustering or classification tasks. A potential future direction is to extend our approach to gain multi-level summaries of the data enabling effective visualisation of a larger number of data points.

The high flexibility offered by ShapoGraphy to combine and position different Shape Glyphs and symbols, including hand-drawn shapes, provides an unprecedented opportunity to easily evaluate various designs. This is an important distinction from glyph-based visualisation methods that have been developed for medical images as they provide a very bespoke representation for the problem at hand making them hard to transfer to other types of images (Ropinski et al., 2011). Notably, it can take time to learn new visual encodings representing specific or complex domain knowledge (Borgo et al., 2013). Once learned, such glyph-based visualisations can become more effective for specialised users. Many examples can be found in the genomics domain including representations of gene variants or ideograms of chromosome structure (Wolfe et al., 2013; L'Yi et al., 2022). Redundant or alternative representations, that are more familiar to the user, can be used in parallel with ShapoGraphy when introducing new visual designs (Cairo, 2013).

An important future direction is to perform a user study for evaluating various aspects of glyph-based designs generated by ShapoGraphy. Given the infinite number of designs that can be generated using ShapoGraphy, such a study should be carefully planned and focused on the most recurring element combinations or designs that are most well-received in the community. Moreover, this assessment should align well with the purpose of the visualisation such as facilitating the discovery of complex patterns, communicating with a broad audience, interpretability, or effectiveness. The user study could advance our understanding of how various elements interact with each other and might highlight potential perturbations that can

be programmatically employed to improve future versions of ShapoGraphy. For example, multilevel glyphs can be used to minimise occlusion (Müller et al., 2014) or sequential highlighting of certain glyph elements selected by the user. This could also inform practices on visual elements that are most effective when combined and which combinations should be avoided which ultimately could accelerate the development of glyph-based visualisations.

Another interesting extension of ShapoGraphy would be the automation of the mapping between numerical features and shapes. One way to achieve that is to adopt a generative approach where multiple glyph-variable mappings are proposed for the user to choose from. Such an approach could inspire visualisation design (Brehmer et al., 2022). This would greatly improve the user experience as currently, the user needs to map features one by one. We tackle this limitation by enabling users to save their mapping along with their created composite glyph configuration as a JSON file for later use. We also offer a range of templates that can be directly used or adjusted by the user.

To conclude, ShapoGraphy can be used in all steps of data analysis to create intuitive pictorial representations of any data type. It can be used to summarise analysis results obtained from clustering or classification approaches, as well as an educational tool. We believe that the unique flexibility offered by ShapoGraphy will expand our visual vocabularies, accelerate the evolution of glyph-based visualisation, inspire creative design, and stimulate the development of new visual encoding schemas. Most importantly, ShapoGraphy is not restricted to image data but can be applied to any numerical data.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are available at www.shapography.com, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

HS conceived the study, devised the concept, designed the use-cases and wrote the paper. YS, MK, and SZ implemented ShapoGraphy web application.

FUNDING

HS is funded by a Sir Henry Wellcome Fellowship (Grant Number 204724/Z/16/Z).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fbinf.2022.788607/full#supplementary-material>

REFERENCES

- Antal, B., Chessel, A., and Carazo Salas, R. E. (2015). Mineotaur: a Tool for High-Content Microscopy Screen Sharing and Visual Analytics. *Genome Biol.* 16, 283. doi:10.1186/s13059-015-0836-5
- Borgo, R., Kehrer, J., Chung, D. H. S., Maguire, E., Laramée, R. S., Hauser, H., et al. (2013). Glyph-based Visualization: Foundations, Design Guidelines, Techniques and Applications. *Eurogr. State Art. Rep.*, 39–63. doi:10.2312/conf/EG2013/stars/039-063
- Brehmer, M., Kosara, R., and Hull, C. (2022). Generative Design Inspiration for Glyphs with Diatoms. *IEEE Trans. Vis. Comput. Graph.* 28, 389–399. doi:10.1109/TVCG.2021.3114792
- Brewer, C. A. (2022). Available at: www.ColorBrewer.org.
- Cairo, A. (2013). *The Functional Art, an Introduction to Information Graphics and Visualization*. United States: New Riders.
- Curtis, C., Shah, S. P., Chin, S. F., Turashvili, G., Rueda, O. M., Dunning, M. J., et al. (2012). The Genomic and Transcriptomic Architecture of 2,000 Breast Tumours Reveals Novel Subgroups. *Nature* 486, 346–352. doi:10.1038/nature10983
- Fuchs, J., Isenberg, P., Bezerianos, A., and Keim, D. (2017). A Systematic Review of Experimental Studies on Data Glyphs. *IEEE Trans. Vis. Comput. Graph.* 23, 1863–1879. doi:10.1109/TVCG.2016.2549018
- Gut, G., Herrmann, M. D., and Pelkmans, L. (2018). Multiplexed Protein Maps Link Subcellular Organization to Cellular States. *Science* 361, 7042. doi:10.1126/science.aar7042
- Heer, J., Bostock, M., and Ogievetsky, V. (2010). A Tour through the Visualization Zoo. *Commun. ACM* 8. doi:10.1145/1743546.1743567
- Javer, A., Rittscher, J., and Sailem, H. Z. (2020). DeepScratch: Single-Cell Based Topological Metrics of Scratch Wound Assays. *Comput. Struct. Biotechnol. J.* 18, 2501–2509. doi:10.1016/j.csbj.2020.08.018
- Krueger, R., Beyer, J., Jang, W. D., Kim, N. W., Sokolov, A., Sorger, P. K., et al. (2020). Facetto: Combining Unsupervised and Supervised Learning for Hierarchical Phenotype Analysis in Multi-Channel Image Data. *IEEE Trans. Vis. Comput. Graph.* 26, 227–237. doi:10.1109/TVCG.2019.2934547
- L'Yi, S., Wang, Q., Lekschas, F., and Gehlenborg, N. (2022). Gosling: A Grammar-Based Toolkit for Scalable and Interactive Genomics Data Visualization. *IEEE Trans. Vis. Comput. Graph.* 28, 140–150. doi:10.1109/TVCG.2021.3114876
- McInnes, L., Healy, J., and Melville, J. (2018). *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*. arXiv: 1802.03426.
- Müller, H., Reihls, R., Zatloukal, K., and Holzinger, A. (2014). Analysis of Biomedical Data with Multilevel Glyphs. *BMC Bioinforma.* 15 Suppl 6, S5–S12. doi:10.1186/1471-2105-15-S6-S5
- Natrajan, R., Sailem, H., Mardakheh, F. K., Arias Garcia, M., Tape, C. J., Dowsett, M., et al. (2016). Microenvironmental Heterogeneity Parallels Breast Cancer Progression: A Histology-Genomic Integration Analysis. *PLOS Med.* 13, e1001961. doi:10.1371/journal.pmed.1001961
- Ropinski, T., Oeltze, S., and Preim, B. (2011). Survey of Glyph-Based Visualization Techniques for Spatial Multivariate Medical Data. *Comput. Graph.* 35, 392–401. doi:10.1016/j.cag.2011.01.011
- Sailem, H. Z., Sero, J. E., and Bakal, C. (2015). Visualizing Cellular Imaging Data Using PhenoPlot. *Nat. Commun.* 6, 5825–5826. doi:10.1038/ncomms6825
- Schapiro, D., Jackson, H. W., Raghuraman, S., Fischer, J. R., Zanotelli, V. R. T., Schulz, D., et al. (2017). HistoCAT: Analysis of Cell Phenotypes and Interactions in Multiplex Image Cytometry Data. *Nat. Methods* 14, 873–876. doi:10.1038/nmeth.4391
- Sero, J. E., Sailem, H. Z., Ardy, R. C., Almuttaqi, H., Zhang, T., and Bakal, C. (2015). Cell Shape and the Microenvironment Regulate Nuclear Translocation of NF- κ B in Breast Epithelial and Tumor Cells. *Mol. Syst. Biol.* 11, 790–816. doi:10.15252/msb.20145644
- Somarakis, A., Van Unen, V., Koning, F., Lelieveldt, B., and Holtt, T. (2021). ImaCytE: Visual Exploration of Cellular Micro-environments for Imaging Mass Cytometry Data. *IEEE Trans. Vis. Comput. Graph.* 27, 98–110. doi:10.1109/tvcg.2019.2931299
- Tufte, E. R. (2001). *The Visual Display of Quantitative Information*. Second ed. Connecticut: Graphics Press LLC. doi:10.2307/3323797
- Walter, T., Shattuck, D. W., Baldock, R., Bastin, M. E., Carpenter, A. E., Duce, S., et al. (2010). Visualization of Image Data from Cells to Organisms. *Nat. Methods* 7, S26–S41. doi:10.1038/nmeth.1431
- Wolfe, D., Dudek, S., Ritchie, M. D., and Pendergrass, S. A. (2013). Visualizing Genomic Information across Chromosomes with PhenoGram. *BioData Min.* 6, 18–12. doi:10.1186/1756-0381-6-18

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Khawatmi, Steux, Zourob and Sailem. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Frontiers in Bioinformatics

Explores innovation in the analysis and interpretation of biological data

An innovative journal that provides a forum for new discoveries in bioinformatics. It focuses on how new tools and applications can bring insights to specific biological problems.

Discover the latest Research Topics

[See more →](#)

Frontiers

Avenue du Tribunal-Fédéral 34
1005 Lausanne, Switzerland
frontiersin.org

Contact us

+41 (0)21 510 17 00
frontiersin.org/about/contact



Frontiers in Bioinformatics

