frontiers | Frontiers in Signal Processing

TYPE Review
PUBLISHED 02 September 2022
DOI 10.3389/frsip.2022.984169

OPEN ACCESS

EDITED BY
Frederic Dufaux,
CNRS, France

REVIEWED BY
Benedetta Tondi,
University of Siena, Italy
William Puech,
Université de Montpellier, France

*CORRESPONDENCE
Mihai Mitrea,
mihai.mitrea@telecom-sudparis.eu

SPECIALTY SECTION
This article was submitted to Image Processing,
a section of the journal
Frontiers in Signal Processing

RECEIVED 01 July 2022
ACCEPTED 02 August 2022
PUBLISHED 02 September 2022

CITATION
Allouche M and Mitrea M (2022), Video fingerprinting: Past, present, and future.
Front. Sig. Proc. 2:984169.
doi: 10.3389/frsip.2022.984169

Video fingerprinting: Past, present, and future

Mohamed Allouche[1,2] and Mihai Mitrea[1]*

[1]Telecom SudParis, ARTEMIS Department, SAMOVAR Laboratory, Evry, France, [2]VIDMIZER, Paris, France

The last decades have seen video production and consumption rise significantly: TV/cinematography, social networking, digital marketing, and video surveillance incrementally and cumulatively turned video content into the predilection type of data to be exchanged, stored, and processed. Belonging to video processing realm, *video fingerprinting* (also referred to as *content-based copy detection* or *near duplicate detection*) regroups research efforts devoted to identifying duplicated and/or replicated versions of a given video sequence (query) in a reference video dataset. The present paper reports on a state-of-the-art study on the past and present of video fingerprinting, while attempting to identify trends for its development. First, the conceptual basis and evaluation frameworks are set. This way, the methodological approaches (situated at the cross-roads of image processing, machine learning, and neural networks) can be structured and discussed. Finally, fingerprinting is confronted to the challenges raised by the emerging video applications (*e.g.*, unmanned vehicles or fake news) and to the constraints they set in terms of content traceability and computational complexity. The relationship with other technologies for content tracking (*e.g.,* DLT - Distributed Ledger Technologies) are also presented and discussed.

KEYWORDS

Video fingerprinting, review, machine learing, neural network, visual feature, DLT (distributed ledger technologies)

1 Introduction

Nowadays, TV/cinematography, social networking, digital marketing, and video surveillance incrementally and cumulatively turned video content into the predilection type of data to be exchanged, stored, and processed. As an illustration, according to Statista, 2022, the TV over Internet traffic tripled between 2016 and 2021, reaching a monthly 42,000 petabytes of data.

Such a tremendous quantity of information, coupled to myriad of domestic/professional usages should be backboned by strong scientific and methodological video processing paradigms, and video fingerprinting is one of these. *Video*

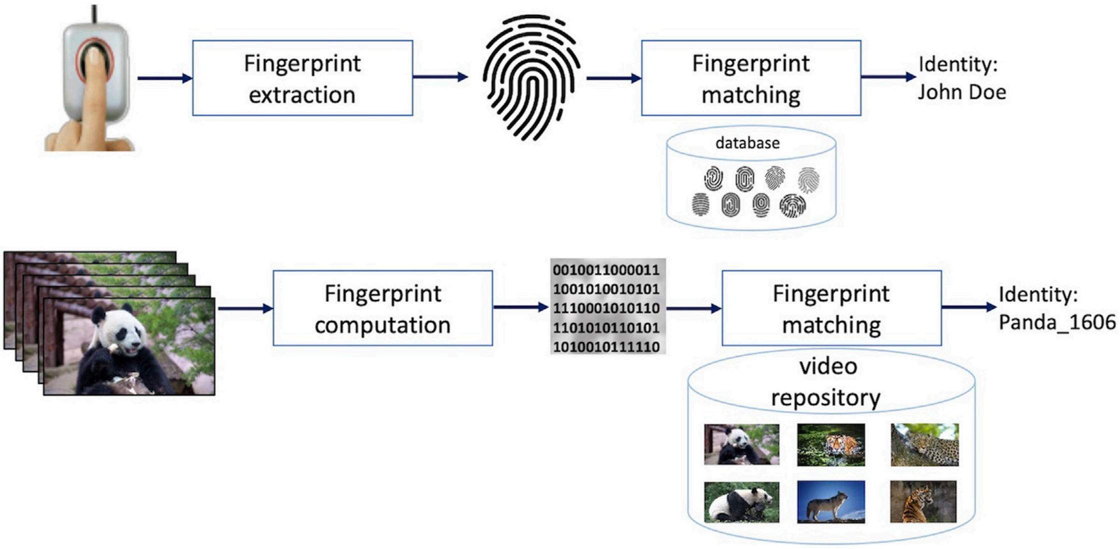

FIGURE 1
Human fingerprinting versus video fingerprinting.

fingerprinting identifies duplicated, replicated and/or slightly modified versions of a given video sequence (query) in a reference video dataset Douze et al., 2008, Lee and Yoo, 2008, Su et al., 2009, Wary and Neelima, 2019. It is also referred to as *near duplicate detection*, or *content-based copy detection* Law-To et al., 2007a. The term *video hashing*[1] (or *perceptual video hashing*) is also in use for fingerprinting applications applied to very large video database search Nie et al., 2015, Liu, 2019, Anuranji and Srimathi, 2020.

Video fingerprint principle can be illustrated in relation to the human fingerprints Oostveen et al., 2002, Figure 1. The patterns of dermal ridges on human fingertips are natural identifiers for humans, as disclosed by Sir Francis Galton in 1893. Although they are tiny when compared to the entire human body, human fingerprints can uniquely identify a person regardless of their physiognomy changes and potential disguises. Analogously, video fingerprints are meant to be video identifiers that shall uniquely identify videos even if their contents undergo a predefined, application dependent set of transformations.

The conceptual premise being generic, the underlying research studies are very different, from both methodological and applicative perspectives. The present paper reports on a state-of-the-art study on the past and present of video fingerprinting while trying to identify trends for its future development. It solely considers the video component and leaves the multimodal approaches (video/audio, video/annotations, video/depth, *etc.*) outside its scope.

The paper is structured as follows. First, Section 2 identifies the *fingerprinting* scope with respect to two related yet complementary applicative frameworks, namely *video indexing* and *video watermarking*. The fingerprinting evaluation framework is set in Section 3. This way, the methodological approaches (situated at the cross-roads of image processing, ML—machine learning and NN—neural networks) can be objectively structured and presented in Section 4. Finally, fingerprinting is confronted to the challenges raised by emerging video processing paradigms in Section 5. Conclusions are drawn in Section 6. A list of acronyms (unless they are commonly known and/or unambiguous) is included after References.

2 Applicative scope

The applicative scope of video fingerprint can be identified through synergies and complementarities with *video indexing* Idris and Panchanathan, 1997 and *video watermarking* Cox et al., 2007. To this end, this section will incrementally illustrate the principles of these three paradigms and will identify their relationship.

Video indexing might be considered as the first framework for content-based video searching and retrieval Idris and Panchanathan, 1997, Coudert et al., 1999. Assuming a video repository, the objective of video indexing is to find all the video

[1] This term may be prone to confusion as *robust video hashing* generally denotes a related yet different research field, devoted to forensics applications Fridrich and Goljan, 2000, Zhao et al., 2013, Ouyang et al., 2015.

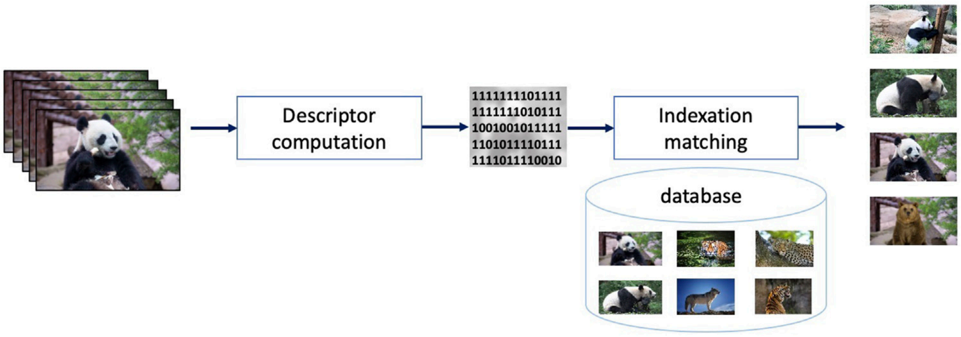

FIGURE 2
Video indexing principle: a binary descriptor is extracted from a query video to retrieve any other related visual content in the dataset.

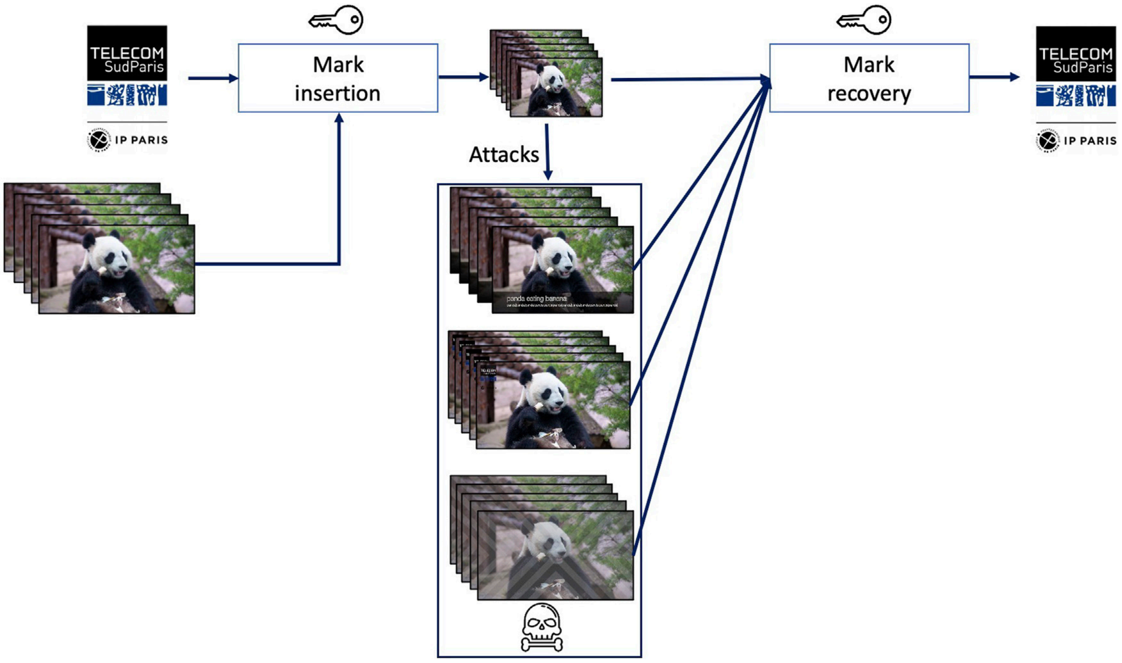

FIGURE 3
Video watermarking principle: a binary watermark is imperceptibly inserted (embedded) in the video sequence; this way, the watermarked sequence can be subsequently identified even when its content is modified (maliciously or not).

sequences that are visually related to a query. For instance, assuming the query is a video showing some Panda bears and the repository consist of some wild animal sequences, a video indexing solution searches for all sequences in the repository that contain Panda bears, as well as images containing the same type of background, as illustrated in Figure 2. To this end, salient information (referred to as *descriptor*) is extracted from the query and compared to the *descriptors* of all the sequences in that repository (that were *a priori* computed and stored). Such a comparison implicitly assumes that a similarity measure for the visual proximity between two video sequences is defined and that a threshold according to which two descriptors can be matched is set.

Digital watermarking Cox et al., 2007 deals with the identification of any modified version of video content, Figure 3. For instance, assuming again a video sequence representing some Panda bears is displayed on a screen and that the screen content is recorded by an external camera, the

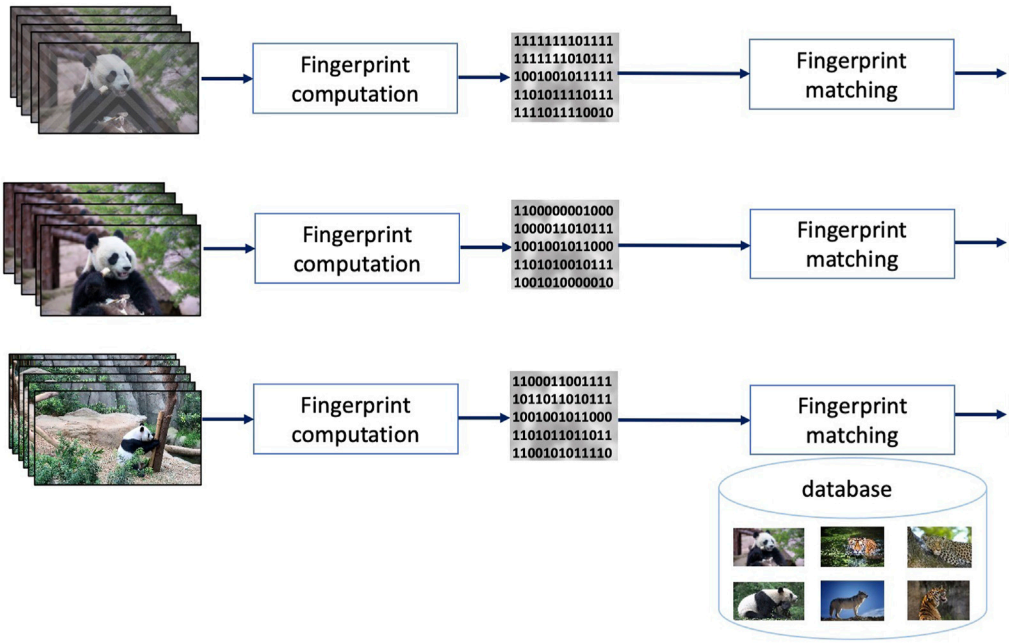

FIGURE 4
Video fingerprinting principle: a binary descriptor extracted from a query video (*fingerprint*) can unambiguously identify all the near-duplicated versions of that content.

original content should be identifiable from the camcordered version. To this end, according to the digital watermarking framework, extra information (referred to as *mark* or *watermark*) is imperceptibly *inserted* (or, as a synonym, *embedded*) into the video content prior to its release (distribution, storage, display, …). By detecting the watermark in a potentially modified version of the watermarked video content, the original content shall be unambiguously identified. Of course, the watermark shall not be recovered from any unmarked content (be it visually related to the original content or not).

Video fingerprinting also deals with identifying slightly modified (replicated, or near duplicated) content, yet its approach is different with respect to both indexing and watermarking, as illustrated in Figure 4. Coming back to the previous two examples, video fingerprinting shall also track a near-duplicated video sequence (*e.g.*, a screen recorded Panda sequence) back to its original (*e.g.*, the Panda original sequence) that is stored in a video repository. Yet, unlike indexing, any other sequence, even visually related to it (*e.g.*, the same Panda bear at a different time of the day and/or in different postures) shall not be detected as identical. To this end, some salient information (referred to as *fingerprint* or *perceptual hash*) is extracted from the query video sequence (note that this information is not previously inserted in the content, as in case of watermarking sequences). By comparing (according to a similarity measure and a preestablished threshold) the query fingerprint to the reference sequence fingerprints, a decision on the visual identity between the video sequences shall be made.

Three main properties are generally considered for fingerprinting.

First, the *unicity* (or *uniqueness*) property assumes that different contents (*i.e.*, content that is neither the query nor one of its near-duplicated versions) result in different fingerprints (in the sense of the similarity measure and of its related threshold).

Secondly, the *robustness* property relates to the possibility of identifying as similar sequences that are near-duplicated. The transformations a video can undergo will be further referred to as *modifications*, *distortions*, or *attacks*, be them malicious or mundane. The video that is obtained through transformations, modifications, distortions, or attacks will be denoted as a *copy*, a *replica video*, a *near duplicated video* or an *attacked video*. While these terms are conceptually similar, fine distinction among them can be made for some specific applicative fields. For instance, Liu et al., 2013 mention at least four different definitions related to near duplicated video content, ranging from "*Identical or approximately identical videos close to the exact duplicate of each other, but different in file formats, encoding parameters, photometric variations (color, lighting changes), editing operations (caption, logo and border insertion), different lengths, and certain modifications (frames add/remove)*" Wu et al., 2007a, Wu et al., 2007b to "*Videos of the same scene (e.g., a person riding a bike) varying viewpoints, sizes, appearances, bicycle type, and camera* *motions. The same semantic concept can occur under different illumination, appearance, and scene settings, just to name a few.*" Basharat et al., 2008. Our study will stay at a generic level and will use these terms as referred to in the cited studies.

Finally, a fingerprinting method is said to feature *dataset search efficiency* if the computation of the fingerprints and the matching procedure ensure low, application dependent computation time. The dataset search efficiency is assessed by the average computation time needed to identify a query in the context of a considered video fingerprinting use case (that is, execution time on a given processing environment and on a given repository).

By comparing among them these three methodological frameworks, it can be noted that:

- *Indexing* and *fingerprinting* share the concept of tracking content thanks to information directly extracted from that content (that is, both *indexing* and *fingerprinting are passive* tracking technique); yet, while fingerprinting tracks the content *per se*, indexing rather tracks a whole semantic family related to that content. From the applicative point of view, indexing and fingerprinting differ in the unicity property.
- *Watermarking* and *fingerprinting* share the possibility of tracking both an original content and its replicas modified under a given level of accepted distortion; yet watermarking requires the insertion of additional information (that is, watermarking is an *active* tracking technique) while fingerprinting solely exploits information extracted from the very content to be tracked.

Moreover, note that *video fingerprinting* is also sometimes referred to as (*perceptual*) *video hashing* Nie et al., 2015, Liu, 2019, Anuranji and Srimathi, 2020. Yet, distinction should be made with respect to *robust video hashing* Fridrich and Goljan, 2000, Zhao et al., 2013, Ouyang et al., 2015 that belongs to the security and/or forensics applicative areas and generally refers to applications where distinction between content preserving and content manipulation attacks should be made. Robust video hashing is out of the scope of the present study.

These properties turn fingerprinting in a paradigm with potential impact in large variety of applicative fields. The ability to identify and retrieve video even under distortions is a powerful tool for automatic video filtering and retrieval, copyright infringement prevention, media content broadcast monitoring over multi-broadcast channels, contextual advertising, or business analytics, to mention but a few Lefebvre et al., 2009, Lu, 2009, Seidel, 2009, Yuan et al., 2016, Wary and Neelima, 2019, Nie et al., 2021.

The analogy between the human and video fingerprints brings to light two key aspects. First, from the conceptual point of view, it implicitly assumes that video fingerprinting exists, that is, that a reduced set of information extracted from the video content makes it possible for the content to be tracked. As this concept cannot be *a priori* proved, it requires comprehensive *a posteriori* validation in a consensual evaluation framework, as discussed in Section 3. Secondly, from the methodological point of view, any video fingerprinting processing pipeline is composed of two main components: the *fingerprint extractor* (that is, the method for computing the fingerprint) and the *fingerprint detector* (that is, the method for searching similar content based on that fingerprint). Consequently, the state-of-the-art studies in Section 4 will be presented according to these two items.

3 Evaluation framework

In a nutshell, the performances of a video fingerprinting system can be objectively assessed by *evaluating its properties (uniqueness, robustness, and dataset search efficiency) on a consensual, statistically relevant dataset*, and this section is structured accordingly. Section 3.1 presents the quantitative measures that are most often considered in state-of-the-art studies, alongside with their statistical grounds. Section 3.2 deals with the datasets to be processed in video fingerprinting experiments and presents the principles for their specification as well as some key examples that will be further referred to in Section 4.

3.1 Property evaluation

The evaluation of the uniqueness and the robustness properties can be achieved by considering fingerprinting as a statistical binary decision problem. Be there a query sequence whose identity is looked up in a reference dataset with the help of a video fingerprinting system.

According to the binary decision principle, when comparing a query to a given sequence in the dataset, two hypotheses can be stated:

○*H0: the query is a replica of a video sequence identified though the tested fingerprint.*
○*H1: the query is not a replica of the video sequence identified through the tested fingerprint.*

The output of the system can be of two types: *positive*, when the query is identified as replica of a video sequence and *negative* otherwise.

When confronted to the ground truth, the statistical decisions can be labeled as: *true*, when the result provided by the test is correct and as *false* otherwise.

Consequently, four types of decisions are made:

○*False positive* (or *false alarm*, denoted by fp): the system erroneously accepts the query as a copy of a reference video sequence.
○*False negative* (or *missed detection*, denoted by fn): the system erroneously rejects a query as a copy of a reference video sequence.
○*True positive* (denoted by tp): the system correctly accepted a query as a copy of a reference video sequence.
○*True negative* (denoted by tn): the system correctly rejected a query as a copy of a reference video sequence.

The objective evaluation of a video fingerprinting system is achieved by deriving performance indicators from the four measures above.

To evaluate the uniqueness property, two measures are generally considered: the P_{fa} (Probability of False Alarm), and the *Prec* (Precision) rate, Su et al., 2009, Lee and Yoo, 2008:

$$P_{fa} = \frac{fp}{fp + tn} \quad Prec = \frac{tp}{tp + fp}$$

P_{fa} and *Prec* are also referred to as *FPR* (False Positive Rate) and *TPR* (True Positive Rate), respectively.

To evaluate the robustness property, the P_{md} (Probability of Missed Detection), and the *Rec* (Recall) rate, are generally considered:

$$P_{md} = \frac{fn}{tp + fn} \quad Rec = \frac{tp}{tp + fn}$$

An efficient fingerprinting method (featuring both unicity and robustness) should jointly ensure low values for P_{fa} and P_{md} while having *Prec* and *Rec* values close to 1. The actual thresholds for these entities depend on the specific use case.

Although *Prec* and *Rec* are two measures commonly used in the evaluation of any information retrieval system, they are not statistical measures as they do not consider the true negative results. Hence, to comprehensively present the properties of a system, P_{fa} and P_{md} should also be considered.

In practice, several other derived and/or complementary performance indicators can be considered, such as the *F1 score*, the *ROC* (Receiver Operating Characteristic), the *AUC* (Area Under the Curve), or the *mAP* (mean Average Precision).

From a theoretical point of view, the dataset search efficiency can be expressed by the computational complexity, that expresses the number of elementary operations required for computing and matching fingerprints as a function of video sequence parameters (frame size, frame rate) and repository size. As such an approach is limitative for NN-based algorithms, the dataset search efficiency property is commonly assessed by the average processing time required by the video fingerprinting system to identify the query within the reference dataset and to output the result for a query. The average processing time can be obtained by averaging the processing time required by the system for the considered collection of queries. Of course, such an evaluation implicitly assumes that detail description is available about the computing configuration (CPU, GPU) performances as well as about the size of the dataset.

3.2 Evaluation dataset

Regardless the evaluated property, the dataset plays a central role, and its design is expected to observe to three constraints: statistical relevance, application completeness, and consensual usage.

The statistical relevance (and implicitly, the reproducibility of the results) mainly relates to the size of the dataset that should ensure the statistical error control (*e.g.*, the sizes of P_{fa}, P_{md}, the related relative errors, . . .) during the algorithmic evaluation and comparison. From this point of view, fingerprinting properties are expected to be reported with statistical precision (*e.g.*, confidence limits for the abovementioned entities).

The application completeness mainly relates to the type of content included in the dataset, that is expected to serve and to cover the applicative scope of the developed method.

The consensual usage relates to the acceptance of the dataset by the research community: this item relates to the possibility of objectively comparing results reported in different studies.

Of course, each dataset and each application evaluated on a specific dataset reach a different trade-off among these three desiderata. Table 1 provides a comparative view about some of the most often considered datasets (see Section 4); some of these corpora are introduced here-after.

TRECVID (TREC Video Retrieval Evaluation) framework Trecvid, 2022, Douze et al., 2008 is a key example in this respect, as it provides consequent benchmarking datasets. Sponsored by the NIST (National Institute of Standards and Technology) with additional support from other US governmental agencies, TRECVID is structured around different "tasks" focused on a particular aspect of the multimedia retrieval problem, as *ad-hoc* video search, instance search, and event detection, to mention but a few. TRECVID datasets consider video copies that are generated under video transformations, such as blurring, cropping, shifting, brightness changing, noise addition, picture-in-picture, frame removing, or text inserting Wang et al., 2016, Mansencal et al., 2018.

Related efforts are also carried out under different frameworks, such as Muscle-VCD (or simply Muscle) Law-To et al., 2007b or VCDB (Large-Scale Video Copy Detection Database) Jiang and Wang, 2016. Research institutes active in the field, like INRIA in France, also created INRIA Copy Days dataset Jegou et al., 2008. National and/or international research projects are also prone to generate datasets Open Video, 2022, Garboan and Mitrea, 2016.

TABLE 1 Examples of datasets processed for fingerprinting evaluation. The lower part (last 5 rows) corresponds to corpora processed by fingerprinting method exploiting NN.

Dataset	No. of video clips	Total duration	Average clip duration	Attacks	Additional info
Muscle-VCD ***2007***	**101** (15 originals)	**80 h** (2.5 h originals)	47 min 30 s	change of color/brightness blur recording with an angle logos/subtitles insertion vertical shift flipping	
CC_WEB_vVIDEO ***2007***	**13,129** (9,300 originals)	**551 h** (387.5 h originals)	2 min 30 s	compression photometric variations postproduction content modification (frame add/remove) frame rate modification	
TRECVID ***2011***	**11,256** (201 originals)	**400 h** (6.7 h originals)	2 min	camcording picture in picture insertions of patterns compression change of gamma decrease in quality postproduction	
VCDB ***2014***	**9,236** (528 originals)	**2,030 h** (27 h originals)	73 s	insertion of patterns camcording scale changes picture in picture	100,000 additional videos in option (to serve as background distraction)
CCV ***2011***	**9,317**	**210 h**	80 s		20 semantic labels (bird, soccer, baseball, . . .)
UCF101 ***2012***	**13,320**	**27 h**	7 s		action recognition data set of realistic action videos
ActivityNet ***2015***	**19,994**	**849 h**	2 min 30 s		specialized for human activity understanding
YLI-MED ***2015***	**50,000**	**625 h**	45 s		specialized for research in multimedia event detection
*Youtube-8M**2018***	**6,100,000**	**350,000 h**	3 min 30 s		

With the advent of NN approaches, research groups affiliated to popular multimedia platforms operators organized and made available large datasets, as presented in the last 5 rows in Table 1. For instance, YouTube-8M Segments dataset Abu-El-Haija et al., 2016 includes human-verified labels on about 237K segments and 1,000 classes, summing-up to more than 6 million video ID or more than 350,000 h of video. The dataset is organized in about 3,800 classes with an average of 3 labels per video. Of course, several other AI datasets coexist. For instance, Zhixiang et al., 2018 points to three of them: CCV (Columbia Consumer Video) Jiang et al., 2011, YLI-MED (YLI Multimedia Event Detection) Bend, 2015, Thomee, 2016, and ActivityNet Heilbron et al., 2015. Note that unlike the TRECVID datasets, the datasets mentioned in this paragraph are not specifically designed for fingerprinting applications but for general video tracking applications (including indexing): hence, the near duplicated content is expected to be created by the experimenter, according to the application requirements and the principles above.

4 Methodological frameworks

While today any fingerprinting state-of-the-art study cannot be either exhaustive or detailed, this section rather focusses on illustrating the main trends than on the impressive variety of studies. It is structured according to the two main steps in a generic fingerprinting computing pipeline: fingerprinting extraction (that is, spatio-temporal salient information extraction) and fingerprinting matching (that is, comparing salient information extracted from two different video sequences). These two basic steps are, in their turn, composed of several sub-steps Douze et al., 2008, Lee and Yoo, 2008, Su et al., 2009. On the one hand, the fingerprinting computation generally includes video pre-processing (*e.g.*, letterboxing removal, frame resizing, frame dropping and/or key-frame detection), local feature extraction, global feature extraction, local/global feature description, temporal information retrieval, and the means for accelerating the search in the dataset (inversed file, *etc.*). On the other hand, the detection procedure generally includes some time-alignment operations (time origin synchronization, jitter cancelation, …), followed by information matching.

Significant differences occur in the ways these steps are implemented. Hence, this section will be structured into two categories, further referred to as *conventional* (Section 4.1) and *NN-based fingerprinting* (Section 4.2) methods. The former category relates to the earliest fingerprinting methods (e.g., 2009–2019) and stems from image processing and machine learning, being backboned by information theory concepts. The latter category is incremental with respect to the former one, as it (partially) considers concepts and tools belonging to the NN realm for achieving fingerprint extraction and matching. Of course, studies combining conventional and NN tools also exist Nie et al., 2015, Nie X. et al., 2017, Duan et al., 2019, Zhou et al., 2019 that will be discussed in Section 4.2.

4.1 Conventional methods

4.1.1 Main directions

As a common ground, these methods stem from image processing, machine learning, and information theory concepts and leverage the fingerprinting extraction on three incremental levels Garboan and Mitrea, 2016.

First, in an attempt to get to frame aspect distortion invariance, the fingerprinting is extracted from derived representations such as 2D-DWT (2D Discrete Wavelet Transform) coefficients Garboan and Mitrea, 2016, 3D-DCT (3D Discrete Cosine Transform) coefficients Coskun et al., 2006, pixel differences between consecutive frames, temporal ordinal measure of average intensity blocks in successive frames Hampapur and Bolle, 2001, visual attention regions Su et al., 2009, quantized block motion vectors, ordinal ranking of average gray level of frame blocks, quantized compact Fourier–Mellin transform coefficients, ordinal histograms of frames Kim and Vasudev, 2005, Sarkar et al., 2008, color layout descriptor, ...

Secondly, frame content distortion invariance can be achieved by the complementary between global features incorporating geometric information (*e.g.*, centroid of gradient orientations of keyframes Lee and Yoo, 2008 or invariant moments of frames edge representation) and local features based on interest points (corner features, Hessian-Affine, Harris points, SIFT (Scale-Invariant Feature Transform), SURF (Speeded Up Robust Features)) generally described under the BoVW (Bag of Visual Words) framework Douze et al., 2008, Jiang et al., 2011.

Thirdly, video format distortion invariance is generally handled by using a large variety of additional synchronization mechanisms, pair designed with the feature selection, from synchronization block, based on wavelet coefficients to K-Nearest Neighbors matching Law-To et al., 2007a of interest points or Viterbi-like algorithms Shikui et al., 2011.

These main directions as well as their mutual combinations will be considered in the next section as structuring elements. They will be illustrated by a selection of 15 studies, published between 2009 and 2019, that will be presented in chronological order. The functional synergies among and between these studies are synoptically presented in Figure 5 that is structured in three layers, shaped as hemicycles:

- the outer blue layer relates to local feature description, exemplified through MPEG-CDVS (Compact Descriptors for Visual Search), ORB (Oriented Fast and Rotated BRIEF), SURF, Transformed domains, SIFT, CS-LBP (Center-symmetric Local Binary Patterns), and HOG (Histogram of Oriented Gradient).
- the middle gray layer relates to global features, exemplified through luminance component, color histograms, and BoVW.
- the inner blue layer relates to the temporal features, exemplified through luminance spectrogram, motion vectors, histogram correlation, optical flow, and TIRI (Temporal Informative Representative Image).

The order of the classes in each hemicycle is chosen to allow for a better visual representation of the synergies among them. The studies represented in gray-shadowed rectangles correspond to conventional methods while the studies represented in white rectangles correspond to NN-based method that also include conventional modules.

4.1.2 Methods overview

The complementarity between visual similarity and temporal consistency is exploited in Tan et al. (2009) to achieve scalability during the detection and localization of video content replicas. The video content synchronization is modeled as a network flow problem. Specifically, the chronological matching of the frames between the two video sequences is replaced by the search for a maximal path that carries the maximum capacity in transmission network, under constraints of type *must-link* and *cannot-link*. As the theoretical solution thus obtained can feature a large complexity, the study also suggests an *a posteriori* simplification based 7 heuristic constraints. The study exploits the idea that the temporal alignment leverages the constraints on visual feature effectiveness and to prove this, a Hessian-Affine detector and PCA-SIFT (Principal Component Analysis) feature are considered in the experiments. On the detection side, key-

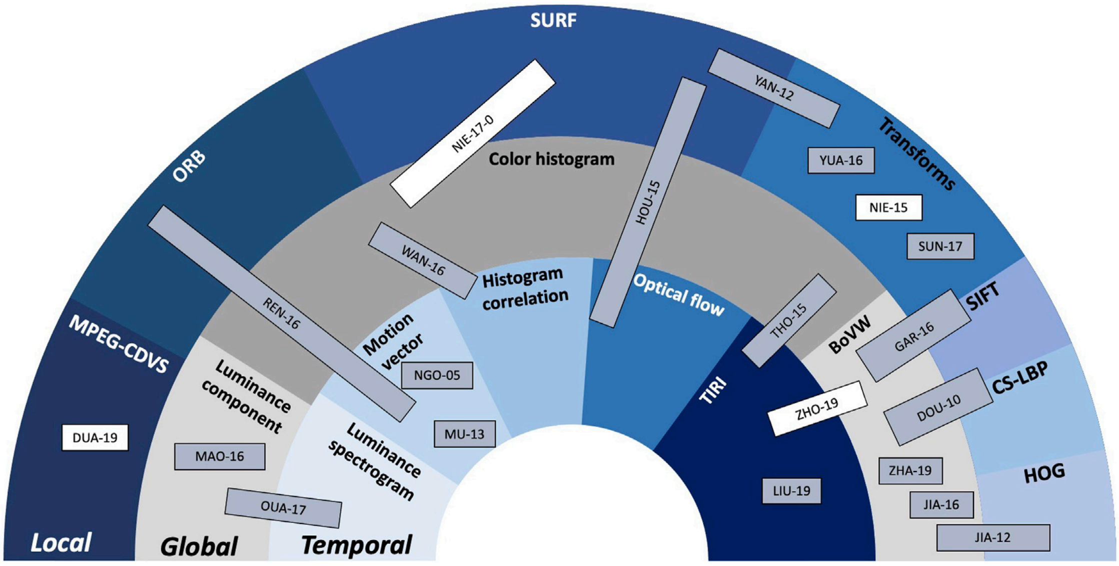

FIGURE 5
Conventional fingerprinting method synopsis: the hemicycles (areas) related to the local, global, and temporal features are located at the outer, middle, and inner parts of the figure, respectively. Inside each hemicycle, examples of state-of-the-art solutions are presented. Conventional methods are presented in gray-shadowed rectangles while NN-based methods that also include conventional modules are represented in white rectangles.

point matching is considered. The experiments are structured at four levels: partial segments of full-length movies to videos crawled from YouTube, detection of near-duplicates in a dataset of more than 500 h, near-duplicate shot detection and copy detection on TRECVID and Muscle-VCD-2007 datasets, respectively.

A fingerprinting method that is optimized for searching of strongly modified sequences in reduced-size video datasets is presented in Douze et al. (2010). The fingerprints are computed from a subset of frames, either periodically sampled from the video sequence or chosen according to a visual content rule (*key frames*). The local visual information is extracted through Hessian-Affine detectors followed by SIFT and CS-LBP descriptors Heikkila et al., 2009. The descriptors are subsequently clustered by a bag of words approach combined to a Hamming Embedding procedure. To improve search efficiency, an inverted file structure is finally considered. For the fingerprinting retrieval, a spatio-temporal verification is performed to reduce the number of potential candidates. The experiments are carried out on the TRECVID 2008 dataset and show how the method parameters can be adjusted to reach a trade-off between accuracy and efficiency.

The study reported in Yang et al. (2012) is based on SURF points Bay et al., 2008 that are first extracted at the frame level. After dividing the frame into 16 even square blocks, the number of SURF points in each quadrant is traversed to build a third-order Hilbert curve that will pass through each quadrant, resulting in an adjacent grid that keeps the same neighborhood as the original image. Finally, the hash bits are computed as the differences of SURF points. To match two fingerprints, the CSR (Clip Similarity Rate) or the SSR (Sequence Similarity Rate) are calculated when the query and the reference videos have the same length, or different length, respectively. The former (CSR) relates to the mean of the matching distances between the 2 hashes while the latter (SSR) represents a weighted average of matched, mismatched, and re-matched frames in query video. The experiments select 40 source videos from TRECVID 2011 framework and 60 of their replicas (logo insertion, picture in picture, video flipping, Gaussian noise). Three types of metrics are used to evaluate the method: *Prec*, *Rec* and *ROC*. For a preestablished *Prec* value (set at 0.8 in the experiments), the advanced algorithm has the best *Rec* value (0.92) compared to the solutions advanced in Zhao et al. (2008) (*Rec* = 0.78) and in Kim and Vasudev (2005) (*Rec* = 0.57).

Aiming at obtaining fingerprint invariance against rotations, Jiang et al., 2012 suggests the joint use of HOG and RMI (Relative Mean Intensity) to express the visual characteristics in the frames. The fingerprinting matching is based on the Chi-square statistics. The experimental results are obtained by processing the Muscle corpus and are expressed in terms of *matching quality*, computed as the ratio of correct answers to total number of queries.

An early work presented in Ngo et al. (2005) considers an approach to video summarization that models the video as a temporal graph, by detecting its highlights based on analyzing motion vectors. That work is the backbone of the fingerprinting technique presented in Li and Vishal (2013) with a focus on the compactness of the fingerprint. The key steps of the algorithm are preprocessing and segment extraction, computing the SGM (Structural Graphical Model), graph partitioning using the graph normalized cuts method, fingerprint extraction, and fingerprint quantization by applying RAQ (Randomized Adaptive Quantizer). For the fingerprint extraction step, the authors selected the TIRI method, based on frame averaging followed by 2-D DCT, Esmaeili et al., 2011. The Hamming distance is used during the matching stage. To test the proposed method, 600 different videos were collected from YouTube, then copy videos were created using 8 different attacks of three major types: signal processing attacks, frame geometric attacks, and temporal attacks. The results present better accuracy particularly with restricted fingerprint length compared to TIRI Esmaeili et al., 2011, CGO (Centroids of Gradient Orientations) Lee and Yoo, 2006, and RASH (Radial hASHing) Roover et al., 2005.

The method presented in Thomas and Sumesh (2015) stands for a simple yet robust color-based video copy detection technique. The first step consists of summarizing the video by extracting the key frames, then generating the TIRI, thus including temporal information in the fingerprint. The second step extracts the color correlation group of each pixel of the TIRIs. The color correlation is clustered into 6 groups, by comparing the intensity of each component in the RGB color space (*e.g.*, group 1 corresponds to $R_{xy} \geq G_{xy} \geq B_{xy}$). Finally, the histogram of the color correlation values is considered as the fingerprinting representation of the video. The matching is done by calculating the normalized distance of the histograms representing the source and the query video clips. The experiments are run on a dataset of 22 source videos and of some of their basic near-duplicated versions (letterboxing, pillarboxing, rotations, . . .). Compared to other state-of-the-art techniques like SIFT or basic color histogram, the advanced color correlation histogram system shows better performances (for the considered modifications) but remains sensitive to color changes (such as grey scale conversion or contrast changes).

A multifeatured video fingerprinting system, designed to jointly improve the accuracy and the robustness is advanced in Hou et al. (2015). The fingerprint computation starts by extracting spatial features from the key frames that have been preprocessed (size and frame rate uniformization), then partitioned into $N_x \times N_y$ non overlapping blocks. Both global and local features are extracted as the mix of the color histogram of all sub images and SURF points, respectively. Additionally, an optical flow feature is extracted as a temporal domain feature: it is represented as a two-dimensional vector that reflects the motion among successive frames. The fingerprinting detection is based on a multiple feature detection matching method that combines the local color histogram feature and the optical flow of SURF points. For the experiments, 30 videos from TRECVID 2010 dataset are processed. Compared to other video fingerprinting algorithms based on local descriptor, such as CGO Hong et al., 2010 and Harris Lee and Yoo, 2008, the video detection *Prec* and *Rec* are slightly improved.

Belonging to the DWT-based fingerprinting family, the study presented in Nie et al. (2015) also focusses on the fingerprint dimensionality. The advanced fingerprinting scheme consists of two types of coefficients, intra-cluster and inter-cluster, thus preserving both global and local information. After normalizing the video clips (at 300 × 240 pixels, 500 frames), the first step is to cluster the frames, according to a graph model based on the K-means algorithm, whose parameters are estimated from the relationships among frames. To select the feature that represents a frame, the fourth order Cumulant of Luminance Component is computed, thus ensuring invariance to different types of distortions (Gaussian noise addition, scaling, lossy compression, and low-pass filtering). The next step reduces the dimensionality while preserving the local and global structures thanks to an algorithm referred to as DOP (Double Optimal Projection): the dimensionality reduction is obtained by multiplying the cumulant coefficient matrix by a mapping matrix. The distance vector thus obtained results in two types of fingerprints: the statistical fingerprint, represented by the kurtosis coefficient of the distance vector and the geometrical fingerprint, represented by the binarization of the distance vector. The matching procedure is performed in two steps: first, according to the distance between the statistical fingerprints and then according to the Hamming distance between the geometric fingerprints (an empiric threshold of 0.18 is considered for the binary decision). The experiments are performed on a dataset of 300 original video clips and of some of their replicas (MPEG compression, letterboxing, frame change, blur, shifting, rotations) and result in both *Prec* and *Rec* larger than 0.95.

The technique presented in Mao et al. (2016) assumes that the probability that five identical successive scene frames occur in two different videos is very low. The fingerprint computation starts by frame resizing (down to *108* × *132*) and division (into *9* × *11* sub-regions). For each sub-region, two types of information are extracted from the luminance component: the mean value of the sub-region and 4 differential elements of the sub-region sub-blocks. This process generates 720 elements in total counting 144 mean values and 576 differential values. The fingerprint is subsequently quantized and clustered. A matching technique based on binary search of inverted file is implemented. A test dataset was created by collecting 510 Hollywood film clips and 756 of their replicas (re-encoding, logo addition, noise addition, picture in picture, . . .). An average detection rate of 0.98 is obtained.

The Shearlet transform is a multi-scale and multi-dimensional transform that is specifically designed to address anisotropic and directional information at different scales. This property can by be exploited in fingerprinting applications, as demonstrated in Yuan et al. (2016), where a 4-scale Shearlet transform with 6 directions is considered. The fingerprinting definition considers both low and high frequency coefficients and is defined under the form of the normalized sum of SSCA (sub-band coefficient amplitudes). Low frequency coefficients are supposed to feature invariance with respect to common distortions, hence, to ensure the fingerprinting robustness. The high frequency coefficients, on their side, are supposed to keep visual content inner information, hence, to contribute to the method uniqueness. Such frame-level fingerprint is coupled to a TIRI of the video. The search efficiency is based on the use of IIF (Invert Index File) mechanism. The experimental results are carried out on visual content sampled from TRECVID 2010 and form INRIA Copy Day dataset Jegou et al., 2008. The replicas are obtained through geometrical distortions (letterboxing, rotation), luminance distortions, noise addition (salt and pepper, Gaussian), text insertion, and JPEG compression. The quantitative results are expressed in terms of *TPR*, *FPR*, and *F1* score and consider as ground two state-of-the-art methods based on the DCT on Ordinal Intensity Signature (OIS). The method main advantage is given by its resilience to geometric transformations (gains of about 0.3 in *F1* score).

The study Wang et al., 2016 is centered around the usage of the temporal dimension expressed as the temporal correlation among successive frames in a video sequence. To this end, the video sequence is structured into groups of frames centered on some key frames (that is, the temporal context for a key frame is computed based on both preceding and succeeding frames). A fingerprint is subsequently extracted from each group of frames. From a conceptual standpoint, the fingerprint is based on the color correlation histogram computed on the frame sequence. Yet, to enhance the overall method speed, this visual information is processed through several types of operations. First, the dimensionality is reduced by projection on a random, bipolar (+1/−1) matrix. Secondly, a binary code is defined based on a weighted addition of the color correlation histogram elements. Finally, the search speed is accelerated by an LSH (Locality Sensitive Hashing) algorithm Datar et al., 2004. The matching algorithm is based on LCS (Longest Common Subsequence) algorithm. The experiments consider 8 transformations included in the TRECVID 2009 dataset and report results (expressed in terms of *Prec* and *Rec*) that are compared against a solution relaying on BoVW and SIFT Zhao et al., 2010: according to the type of attacks, absolute gains between 5 and 14% in *Prec* and between 6 and 12% in *Rec* are shown. Although the method was optimized for reducing the search time, no experimental result is reported in this respect.

A fingerprinting system based on contourlet HMT (Hidden Markov Tree) model is designed in Sun et al. (2017). The contourlet is a multidirectional and multiscale transform that is expected to handle the directional plane information Do and Vetterli, 2004 better than the well-known wavelets transform. HMT generates links between the hidden state of the coefficients and their respective children. Before the extraction of the fingerprint, a normalization phase takes place. It unifies the frame rate, the width, and the height, and converts the frames to grayscale. Once normalized, each frame is partitioned into equal blocks, thus preserving the local features. The contourlet transform is then applied to each block to obtain the contourlet coefficients which are fed to the HMT model to generate the standard deviation matrices. Finally, the SVD (Singular Value Decomposition) is used to reduce the dimension of the resultant standard deviation matrices. The video fingerprint is created by concatenating the fingerprints extracted from all the frames. This study adopts a 2-step matching algorithm. In the first step, the fingerprint of a random frame is used to compute its distance to all the fingerprints present in the dataset. The *N* best matches are further investigated in the second step where the squared Euclidean distance between all the frames presenting the query clip and a referenced clip is calculated. The reference video with the minimum distance is identified as the matching result. Compared to the CGO based method Lee and Yoo, 2008, the Sun et al., 2017 method achieves better performances in terms of the probability of false alarm and the probability of true detection.

Ouali et al., 2017 extends some basic concepts from audio to video fingerprinting. To this end, the video sequence is considered as a sequence of frames that are first resized. The fingerprint encodes the positions of several *salient* regions in some binary images generated from the luminance spectrogram; in this study, the term *salient* designates the regions featuring the highest spectral values. The selection of the salient areas can be done at the level of the frame or at the level of successive frames. The former considers a window of spectrogram coefficients centered on the related median while the latter considers the regions that have the highest variations compared to the same regions in the previous frame. The experimental results are carried out on the TRECVID 2009 and 2010 datasets and show that the fingerprint extracted on sequences of frames outperforms the fingerprint extracted at the level of frames.

The study presented in Liu (2019) addresses the issue of reducing the complexity and the execution time of the fingerprint matching in large datasets. The method to extract the fingerprint is referred to as *rHash* and it is derived from the *aHash* method Yang et al., 2006. First, a pre-processing step reduces the frame rate to 10, uniformizes the resolution to *144x176*, and generates the TIRIs Esmaeili and Ward, 2010. Secondly, the *rHash* involves 4 steps: image resizing, division into blocks, block-wise local mean computation, and the binarization of each pixel based on the correspondent block mean value. The *rHash* outputs a fingerprint composed of 12 words of 9 bits each. For the matching process, an algorithm based on a look-up table, word counting, and ordering operations is advanced. The TRECVID 2011 and the VCDB Jiang and Wang, 2016 datasets are processed when benchmarking the advanced method against *aHash* and *DCT-2ac hash* Esmaeili et al., 2011 methods: higher accuracy as well as increased searching speed are thus brought to light.

As video content is preponderantly recorded, stored, and transmitted in compressed formats, fingerprints extracted directly from the compressed stream will beneficially eliminate the need for decoding operations. While early studies Ngo et al., 2005, Li and Vishal, 2013 already considered MPEG motion vectors as a partial information in fingerprinting applications, Ren et al., 2016 can be considered as an incremental step: the fingerprinting computation combines information extracted from the decompressed (pixel) domain to information extracted at the MPEG-2 stream level. First, from the decompressed *I* frames, key frames are selected according to their visual saliency. To this end, histogram-based contrast is computed for each *I* frames alongside with the underlying image entropy. Then, key frames are selected according to the Person's coefficient. For any selected key frame, both global and local features are extracted as the color histograms and ORB descriptors, respectively. Finally, motion vectors directly extracted from the MPEG-2 stream serve local temporal information: specifically, motion vectors angle histograms are computed. Hence, the key frame fingerprint is a combination of the color histograms, ORB descriptors and motion vector normalized histogram. The video fingerprint is computed as the set of key frame fingerprints. The matching procedure is individually performed at the level of the three components (*i.e.*, based on their individual appropriate matching criteria) and the overall decision is achieved through fusing decisions made on multiple features by a weighted additive voting model. In experiments, the color histogram, ORB descriptors and motion vector histograms weights are set to 0.2, 0.4, and 0.4, respectively. The experimental results are obtained by processing the TRECVID 2009 dataset and consider one state of the art measure based on SIFT. The gains of the advanced algorithm have been evaluated in terms of NDCR (Normalized Detection Cost Rate), *F1* score, and copy detection processing time.

4.1.3 Discussion

The previous section brings to light that the fingerprinting conventional methods form a fragmented landscape. While the general methodological framework is unitary (*cf*. Section 3), each study ambitions to take a different applicative challenge, from searching of strongly modified sequences in reduced-size video datasets to reducing the complexity and the execution time of the fingerprint matching. The evaluation criteria are different, with a preponderancy of *Prec*, *Rec* and *F1* that are generally computed on datasets sampled from the corpora presented in Table 1; yet, the criteria of sampling the reference datasets are not always precised. In this context, no general and/or precise conclusion about the pros and the cons of the state-of-the-art methods can be drawn.

However, the value of these research efforts can be collectively judged by analyzing their steadily evolution, as illustrated in Figure 6. This figure covers the 2009—2019 time span and presents, for each analyzed year, the key conceptual ideas (the dark-blue, left block) as well as the methodological enablers in fingerprinting extraction (the blue, right-upper block) and matching (the light-blue, right-lower block)[2].

Figure 6 and Section 4.1.2 show that the state-of-the-art is versatile enough to pragmatically offer solutions to specific applicative fields, without being able to provide the ultimate fingerprinting method. As an attempt in reaching such a solution, NN—based solutions are considered for some or all of the blocks in the fingerprinting scheme, as explain ion Section 4.2.

4.2 NN-based methods

4.2.1 Main directions

The class of NN-based video fingerprinting methods can be considered as an additional direction with respect to the conventional fingerprinting methods presented in Section 4.1. They inherit its basic conceptual workflow: pre-processing video sequence, extracting spatial and temporal information, eventually aggregating them into various derived representations (be them binary or not), matching.

However, NN-based video fingerprinting methods rely (at least partially) on various types of NN, from AlexNet Krizhevsky et al., 2012 and ResNet (Residual neural network) (He et al., 2016) to CapsNet (Capsule Neural Network) Sabour et al., 2017 and LSTM Hochreiter and Schmidhuber, 1997, sometimes requiring specifically designed architectures Zhixiang et al., 2018. Yet, such an approach does not exclude the usage of partial conventional solutions in conjunction with NN, *e.g.*, BoVW can be considered as an aggregation tool of visual features extracted by CNN (Convolutional Neural Network) Zhang et al., 2019. Moreover, the matching algorithm generally comes across with the NN considered in the extraction phase.

These main directions will be illustrated by a selection of 20 studies, published since 2016, that will be presented in chronological order. The relationship among and between them is depicted in Figure 7, that is also structured in three hemicycles (as Figure 5), yet their meanings are slightly different:

- the outer blue layer corresponds to the spatial features, exemplified through: CRBM(Conditional Restricted Boltzmann Machine), ResNet, NIP (Nested Invariance Pooling), VGGNet, AlexNet, GoogleNet, new structures designed to the fingerprinting purpose, RetinaNet, and Tracked HetConv-MK (heterogeneous convolutional multi-kernel).

[2] For a specific year, the information presented in Figure 6 may correspond to several references.

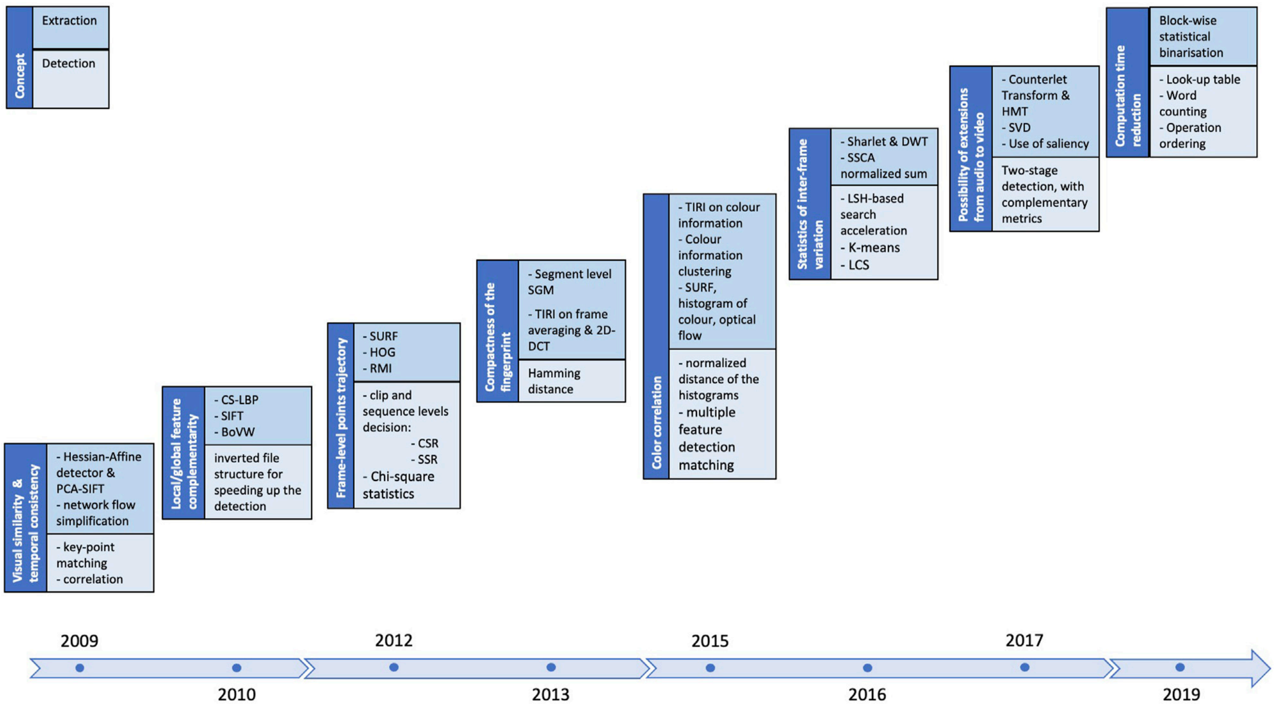
Concept
Extraction
Detection
Visual similarity & temporal consistency
- Hessian-Affine detector & PCA-SIFT
- network flow simplification
- key-point matching
- correlation
Local/global feature complementarity
- CS-LBP
- SIFT
- BoVW
inverted file structure for speeding up the detection
Frame-level points trajectory
- SURF
- HOG
- RMI
- clip and sequence levels decision:
- CSR
- SSR
- Chi-square statistics
Compactness of the fingerprint
- Segment level SGM
- TIRI on frame averaging & 2D-DCT
Hamming distance
Color correlation
- TIRI on colour information
- Colour information clustering
- SURF, histogram of colour, optical flow
- normalized distance of the histograms
- multiple feature detection matching
Statistics of inter-frame variation
- Sharlet & DWT
- SSCA normalized sum
- LSH-based search acceleration
- K-means
- LCS
Possibility of extensions from audio to video
- Counterlet Transform & HMT
- SVD
- Use of saliency
Two-stage detection, with complementary metrics
Computation time reduction
Block-wise statistical binarisation
- Look-up table
- Word counting
- Operation ordering
2009
2010
2012
2013
2015
2016
2017
2019

FIGURE 6
Incremental evolution of the conventional methods.

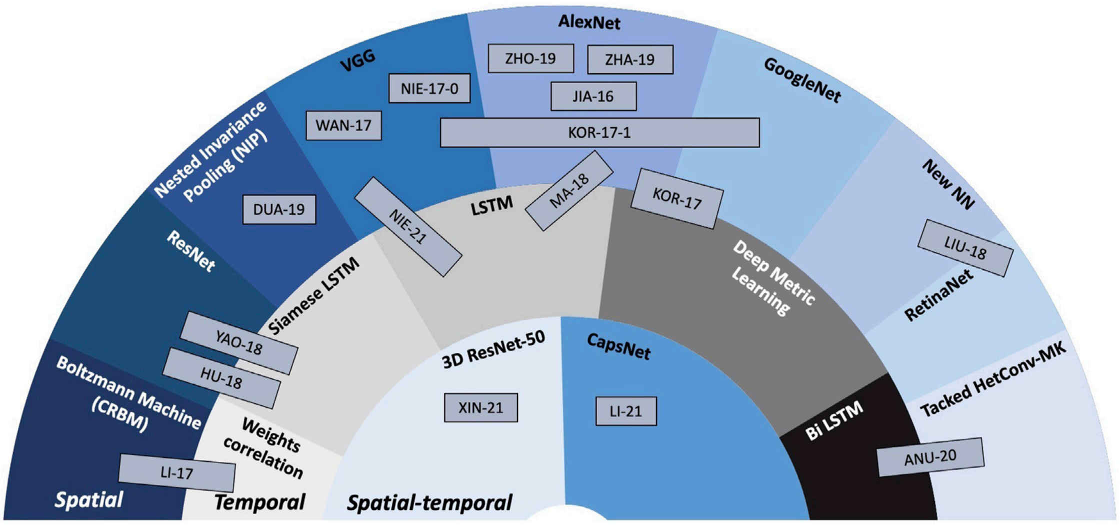

FIGURE 7
NN-based fingerprinting method synopsis: the hemicycles (areas) related to the spatial, temporal, and spatial-temporal features are located at the outer, middle, and inner parts of the figure, respectively. Inside each hemicycle, examples of state-of-the-art solutions are presented.

- the middle gray layer corresponds to temporal features, exemplified through: weight correlation, LSTM (Long-Short Term Memory), SiameseLSTM (Siamese LSTM), Deep Metric Learning, and BiLSTM (bidirectional LSTM).
- the inner blue layer corresponds to spatial-temporal features, exemplified through: 3D-ResNet50, and CapsNets structures.

The order of the classes in each hemicycle is again chosen to allow for a better visual representation of the synergies among them.

4.2.2 Methods overview

The work presented in Jiang and Wang (2016) is twofold. First, the VCDB is organized and presented by a comparison to other existing datasets (*e.g.*, Muscle-VCD) used to evaluate video copy detection algorithms. In parallel, a fingerprinting method referred to as SCNN (Siamese Convolutional Neural Network) is advanced. SCNN is composed of two identical AlexNet Krizhevsky et al., 2012 followed by a connection function layer that computes the Euclidean distance between the two AlexNet outputs, and finally a contrastive loss layer Hadsell et al., 2006. The information thus obtained is structured by BoVW. The experiments focus on the relationship between the dataset and the efficiency of the system. The rule of thumb that is thus stated is "*the bigger and the more heterogeneous the dataset, the harder for the systems to accurately detect copy videos*". Specifically, the SCNN achieves $F1 = 0.69$ on VCDB.

A two-level fingerprint approach is presented in Nie X. et al. (2017). First, LRF (Low-level Representation Fingerprint) is computed as a tensor-based model that fuses different visual features such as SURF and color histograms. Then, the DRF (Deep Representation Fingerprint) extracts the deep semantic features by using a pretrained VGGNet Karen and Andrew, 2014 containing five convolutional layers and 3 fully connected layers. The DRF takes 224 × 224 RGB images as input and outputs a 4096-dimension vector. The matching solution is also structured at two levels: the LRF component identifies a candidate set while the DRF further identifies the source video from the candidate set. The experiments consider both CC_WEB_VIDEO Wu et al., 2009 and Open Video Open Video, 2022 datasets, thus processing about 20,000 source clips. The method is benchmarked against four methods LRTA Li and Vishal, 2012, 3D DCT Baris et al., 2006, CGO Lee and Yoo, 2008, and CMF Nie et al., 2017b that it outperforms in terms of ROC curve.

The study in Schuster et al. (2017) discloses a video stream fingerprinting. The method takes advantage of the loophole in the MPEG-DASH standard Sodagar, 2011 that induces an outburst of content dependent packet bursts, despite the stream encryption. The video is represented as information bursts that are sent to the end user from the streaming services. The data traffic features are captured via a script on the client device or intruding detectors in the network. To this end, a CNN composed of 3 convolution layers, max pooling, and 2 dense layers is designed. To train the model, an Adam optimizer Kingma and Jimmy, 2014 was used as well as a categorical cross-entropy error function. The dataset is extracted from 100 Netflix titles, 3,558 YouTube videos, 10 Vimeo and 10 Amazon titles. A different model is trained for each streaming platform. The classifier achieved 92% accuracy. Inspired by these results, the study in Li (2018) further investigates the aspects specifically related to network, by extracting the information from the Wi-Fi traffic, where both transport and MAC (Media Access Control) layers are encrypted via TLS (Transport Layer Security), and WPA-2 (Wi-Fi Protected Access 2), respectively. The Multi-Layer Perceptron (MLP) model achieves 97% accuracy to identify videos from a small 10-video dataset.

Instead of using the output of the CNN as visual features, the study in Kordopatis-Zilos et al. (2017a) advances a method that extracts the image features starting from the activation values in the convolutional layers. The extracted information forms a frame-level histogram. A video-level histogram is then generated by summing all the frame-level histograms. For fast video retrieval, TF-IDF weighing is coupled to an inverted file indexing structure Sivic and Zisserman, 2003. To evaluate the proposed method, the CC_WEB_VIDEO Wu et al., 2009 dataset is used as well as 3 pre-trained CNNs, namely AlexNet Krizhevsky et al., 2012, GoogleNet Szegedy, 2015, and VGGNet Simonyan and Zisserman, 2014. GoogleNet performed the best ($mAP = 0.958$), followed by AlexNet ($mAP = 0.951$) than VGGNet ($mAP = 0.937$).

A deep learning architecture with a focus on DML (Deep Metric Learning) is presented in Kordopatis-Zilos et al. (2017b). For feature extraction, the video is sampled to 1 frame per second then fed to a pre-trained CNN model (AlexNet Krizhevsky et al., 2012 and GoogleNet Szegedy, 2015 are considered). For the DML architecture, a triplet-based network is proposed where an anchor, a positive and a negative video are used to optimize the loss function. The first layer of the DML is composed of 3 parallel Siamese DNN (Deep Neural Network). In their turn, the Siamese DNN are composed of 3 dense fully connected layers followed by a normalization layer where the sizes of the layers and their outputs depend on the input size. The VCDB dataset Jiang and Wang, 2016 is used to train the DML. To evaluate the proposed system, the CC_WEB_VIDEO Wu et al., 2009 dataset is used. The system scores $mAP = 0.969$ using the GoogleNet and $mAP = 0.964$ when using AlexNet, thus increasing the performances presented in Kordopatis-Zilos et al. (2017a).

The study presented in Li and Chen (2017) develops a deep learning model capable of extracting spatio-temporal correlations among video frames based on a CRBM Taylor et al., 2007 that can simultaneously model the spatial and temporal correlations of a video. The spatial correlations are modeled by the connections between the visible and hidden layers at a given moment. The temporal correlations are modeled by the connections among the layers at different timestamps. The CRBM is paired with a denoising auto-encoder Vincent et al., 2010 module that reduces the dimension of the CRBM output by reducing the redundancies and discovering the invariants to distortions. This process can be applied recursively. A so-called post-processing module takes as input 2 video fingerprints and decides whether they are similar or not. The TRECVID 2011 dataset is used for benchmarking. The advanced method reaches *F1* = 0.98, thus outperforming four state-of-the-art techniques: SGM Li and Vishal, 2013 *F1* = 0.91), 3D-DCT Esmaeili et al., 2011 *F1* = 0.89, Lee and Yoo, 2008 *F1* = 0.78, and RASH Roover et al., 2005 *F1* = 0.79.

The study in Wang et al. (2017) investigates the influence of the frame sampling that is usually applied at the beginning of fingerprint extraction and sets its goal on computing a compact fingerprint without decreasing the frame rate (that is, without frame dropping). Three main steps are designed: frame feature extraction, video feature encoding, and video segment matching. The frame feature extraction is realized by means of a VGGNet-16 Simonyan and Zisserman, 2014 composed of 13 convolutional layers, 3 fully connected layers, and 5 max-pooling layers inserted after the convolutional layers. This step follows by PCA whitening on the CNN output to reduce its dimensionality. The feature compression and aggregation are realized via the sparse coding technique and timeline aligning by pooling the frame features into 1sec. interval (max-pooling is chosen). The matching features fast retrieval is ensured by using a KD-tree to store the fingerprints and temporal alignment implemented according to the temporal network described in Jiang et al. (2014). To run the tests, the VCDB dataset Jiang and Wang, 2016 is used. The advanced method performs better than two baseline fingerprinting methods: CNN with AlexNet Jiang and Wang, 2016, and Fusion with SCNN Jiang and Wang 2016. The experiments also studied the impact of frame sampling: *F1* = 0.7 when processing all the frames and it drops to *F1* = 0.66 when processing 1 frame per second.

The study in Hu and Lu (2018) combines CNN and RNN (Recursive Neural Network) architectures for video copy detection purposes. The method is divided into 2 main steps. First, a CNN architecture extracts content features from each frame: by a ResNet model He et al., 2016, each frame is represented by a 2048-component vector. Secondly, spatio-temporal representations are generated on top of frame-level vectors. Thus, a Long-Short Term Memory unit based Siamese Recurrent Neural Networks (SiameseLSTM) is trained. The training is achieved by selecting clips with the same length (20 frames) from CC_WEB_VIDEO Wu et al., 2009. For video searching/matching purposes, the video is cut into 20 frame clips, before their respective spatial-temporal representations are generated. To identify the copied segments a graph based temporal network algorithm is used Tan et al., 2009. This algorithm is tested using the VCDB dataset Jiang and Wang, 2016 and yields *Prec* = 0.9, *Rec* = 0.58, and *F1* = 0.7233.

Liu, 2018 represents an example of spatial fingerprinting relying on CNN. The principle is to represent the video sequence as a collection of conceptual objects (in the computer vision sense) that are subsequently binarized. To compute the fingerprint, the video sequence is first space/time down sampled. For each down sampled frame, visual objects are computed using the RetinaNet structure Lin et al., 2017. The binarization of the detected objects is recursively block-wise achieved: each object is divided into a group of non-overlapping blocks and each block in several non-overlapping subblocks. The fingerprinting bits are assigned according to a thresholding operation: the subblock pixel value average is compared to the average of all the pixels in the corresponding block. The matching technique considers an IIF structure and a weighted Hamming distance. The experimental results concern the values of *Prec*, *Rec* and *F1*, computed on VCDB dataset and show that a 10% higher recall rate can be achieved with a decrease of 1% prediction rate [the comparison is made against an ML method based on SIFT descriptors as well as against the CNN method presented in Wang et al. (2017)].

The method presented by Zhixiang et al. (2018) proposes a nonlinear structural video hashing approach to retrieve videos in large datasets thanks to binary representations. To this purpose, a multi-layer neural network is designed to generate a compact *L*-bit binary representation for each frame of the video. To optimize the matching process, a subspace grouping method is applied to each video, thus decomposing the nonlinear representation to a set of linear subspaces. To compute the distance between 2 video clips, the distances between the underlying subspaces are integrated, where the Hamming distance is used to compute the distance between a pair of subspaces. CCV Jiang et al., 2011, YLI-MED Bend, 2015 and ActivityNet Heilbron et al., 2015 datasets are selected to test the performance of the algorithm that is benchmarked against DeepH (Deep Hashing) Liong et al., 2015, SDH (Supervised Discrete Hashing) Shen et al., 2015, and KSH (Kernel-Based Supervised Hashing) Liu et al., 2012. The experimental results show that the advanced method outperforms state-of-the-art solutions with the increase of code length.

An unsupervised learning video hashing technique is advanced in Ma et al. (2018). The first step is to extract the spatial feature of the video frames using AlexNet Krizhevsky et al., 2012. The output of the CNN is fed to a single-layer LSTM network. Next, a time series pooling is applied. This step combines all the frame level features to form a single video level feature. Finally, an unsupervised hashing network extracts a compact binary representation of the video. To test its effectiveness, UCF-101 Soomro et al., 2012 dataset and 100 h worth of videos are downloaded from YouTube and used as dataset. Few unsupervised hashing networks were evaluated and the ITQ-ST (Iterative Quantizing—Spatio-Temporal) and BA-

ST (Binary Autoencoder—Spatio-Temporal) Carreira-Perpinán and Raziperchikolaei, 2015 methods worked the best to represent the videos, resulting into $mAP \geq 0.65$.

The joint use of CNN (ResNet He et al., 2016) and RNN (SiameseLSTM) is studied in Yaocong and Xiaobo (2018). The selected CNN is ResNet50 that takes 224 × 224 RGB frames as input and outputs a 2048-dimension vector per frame. The RNN achieves the spatio-temporal fusion and sequence matching. To further optimize spatio-temporal feature extraction, positive pairs (similar video content) and negative pairs (dissimilar video content) are fed to the SiameseLSTM. The resulting feature vectors are considered as the video fingerprint. For the matching process, a graph based temporal network Tan et al., 2009 is used. For training, the CC_WEB_VIDEO Wu et al., 2009 dataset is used, and the video clips are normalized to 20 frames. For evaluation, the VCDB dataset Jiang and Wang, 2016 is used. The method yields $Prec$ = 90% and Rec = 58%, which is slightly better than the solution advanced in Wang et al. (2017) and Jiang and Wang (2016).

The challenge of retrieving the top-k video clips from a single frame is taken in Zhang et al. (2019), where visual features are extracted by utilizing CNN and BoVW. The first step is to extract representative frames at fixed-time intervals and to resize them to 256 × 256 pixels. The second step is to feed all those images to a CNN feature extractor, implemented by the AlexNet architecture Krizhevsky et al., 2012. For each frame, a 4096-dimension feature vector is generated. This vector is the input for the BoVW module which aims to create a visual dictionary for the reference video dataset via a feature matrix. The extraction of visual words from visual features is done via the K-means clustering method. To optimize the retrieval time, frame pre-clustering is done, also based on K-means. A VWII (Visual Word Inverted Index) is deployed to improve search efficiency. The performance of the algorithm is benchmarked against SIFT Zhao et al., 2010, and BF-PI de Araújo and Girod, 2018 methods, on two datasets, namely Youtube-8M Abu-El-Haija et al., 2016 and Sports-1M Karpathy et al., 2014. The experimental results consider 4 criteria, namely precision evaluation on the size of dataset, precision evaluation on the number of visual words, efficiency evaluation (execution time) on the number of k results, and the efficiency evaluation (execution time) on the size of dataset.

Duan et al., 2019 presents an overview of the CDVA (Compact Descriptors for Video Analysis standard) promoted by ISO/IEC JTC 1 SC 29, a. k.a MPEG. The DVA framework is specified incrementally with respect to MPEG-CDVS. To extract video features, the key frames and the inter feature prediction are determined before being fed to a deep learning model based on CNN. The proposed CNN model is derived from NIP feature descriptors which adds robustness to the system. To make the system light weight and multiplatform, the NN was compressed by using the Lloyd-Max algorithm. To reduce the time of video retrieval time, the output of the CNN is binarized via a one-bit scalar quantizer. A Hamming distance is used for the fingerprint matching. For testing, a dataset with source and attacked clips is created gathering 4,693 matching pairs as well as 46,930 non-matching pairs. By coupling the deep learning extracted features with the handcrafted features proposed in CDVS, the system gained further precision.

The study in Zhou et al. (2019) presents a video copy detection method establishing synergies among CNN and conventional computer vision tools. The first step consists in dividing the video into equal-length video sequences, from which frames are sampled with a fixed period, thus allowing the computation of the TIRI for each sequence. The second step consists in extracting the spatial features using a pre-trained AlexNet Krizhevsky et al., 2012 model, followed by a sum-pooling layer to reduce the matrix dimension. The model takes as input the TIRI and outputs a 256-dimension vector. The third step extracts the temporal features. In this respect, it starts by feeding all video sequence frames to the AlexNet and follows by averaging all frame matrices and by computing their centroids. Two matrices representing the distance in cylindrical coordinates (distance and angle) between the centroids are subsequently computed. The fourth step first creates a BoVW by clustering the extracted spatial features through a K-means algorithm and then structures the BOVW in an inverted index file. During the copy detection step, for each query-reference pair, three individual distances are computed: between spatial representations, between temporal distance representations and between temporal angle representations. These three distances are fused to compute a decision score that is compared to a pre-defined threshold, thus ascertaining whether the query is a copy version or not. Evaluated under the TRECVID 2008 framework, the method achieves mAP = 0.65.

A supervised stacked HetConv-MK Singh et al., 2019 and BiLSTM hashing model is designed in Anuranji and Srimathi (2020). The model integrates two main blocks devoted to spatial and temporal feature extraction, respectively. First, the convolutional block computes the spatial features via passing the frames through a stacked convolutional filter and a max-pooling layer. Secondly, the BiLSTM model computes the stream forward and backward. Finally, a fully connected layer generates a binary fingerprint that integrates the output of the previous units. The experimental results are obtained out by processing 3 datasets: CCV Jiang et al., 2011, ActivityNet Heilbron et al., 2015, and HMDB (Human Metabolome Database) Kuehne et al., 2011, with a total of almost 30,000 clips. To determine the effectiveness of the algorithm, Hamming ranking, and Hamming lookup are used in conjunction with mAP and $Prec$. The advanced method is compared to existing methods such as SDH (Supervised Discrete Hashing) Shen et al., 2015, supervised deep learning Liong et al., 2015, Deep Hashing Liong et al., 2015, and ITQ (Iterative Quantization) Gong et al., 2013. The results show an improvement in accuracy introduced with large scale dataset.

A video hashing framework, referred to as CEDH (Classification-Enhancement Deep Hashing) is conceived in Nie et al. (2021). CEDH is a deep learning model that is composed of 3 main layers. First, a VGGNet-19 Simonyan and Zisserman, 2014 layer to extract frame-level features. Then, a LSTM Hochreiter and Schmidhuber, 1997 network is adopted to capture temporal features. Finally, a classification module is implemented to enhance the label information. To train the model, the loss term is matched to the peculiarities of the layer: triplet loss, classification loss, and code constraint terms, respectively. To evaluate its performance, 3 video datasets are processed: the FCVID (Fudan-Columbia VIDeo) dataset Jiang et al., 2018, HMDB Kuehne et al., 2011, and UCF-101 Soomro et al., 2012, thus resulting in a total of 7,070 video clips for training and 3,030 clips for testing. The CEDH is benchmarked against 8 state-of-the-art solutions, namely: locality sensitive hashing Datar et al., 2004, PCA hashing Wang et al., 2010, iterative quantization Gong et al., 2013, spectral hashing Weiss et al., 2009, density sensitive hashing Jin et al., 2014, shift-invariant kernel local sensitive hashing Raginsky and Lazebnik, 2009, self-supervised video hashing Song et al., 2018, and deep video hashing Liong et al., 2017. The evaluation criteria are *mAP*, *Prec* and *Rec*.

A hybrid method combining deep learning and hashing techniques to achieve a video fingerprinting technique is presented in Xinwei et al. (2021). The method is based on quadruplet fully connected CNN, centered around 4 *3D ResNet-50* networks that extract spatio-temporal features. The input is composed of 4 videos: the source clip, a copy of the clip (a modified version extracted from the original), and 2 clips that are not related to the original clip. The output consists of 2 elements: a 2048-dimension vector and a 16 bits binary code. For training and testing, three public datasets are considered: UCF-101 Soomro et al., 2012, HMDB Kuehne et al., 2011 and FCVID Jiang et al., 2018. A normalization process of the 4,986 videos takes place before the training, where each video is downsized to *320×240* and only the first 100 frames of each clip are used to identify the video. The proposed method is mainly compared to a similar deep learning method that shares global architectural similarities called NL_Triplet. The two methods have a similar performances and behaviors in the various benchmarking setups.

The study in Li et al. (2021) presents a fingerprinting method that takes advantage of the capabilities of the CapsNet Sabour et al., 2017 to model the relationships among compressed features. The architecture of the convolution layers is composed of two 3D-convolution modules extracting spatio–temporal features, followed by an average pooling module along temporal dimension and finally by a 2D-convolution module. The role of the primary capsule layer is convolution computation and dimension transformation, while the advanced capsule is composed of 32 neurons and is responsible for matrix transformations and dynamic routing Sabour et al., 2017. The output of this architecture is a 32-dimension fingerprint. A triplet network is designed for the matching. During the training, the matching network requires three inputs: an anchor sample (original video), a positive sample (a copy/modified of the original video), and a negative sample (non-related video). The dataset is composed of 4,000 videos randomly sampled from FCVID Jiang et al., 2018, TRECVID, and YouTube. The ROC and F1 scores are considered as evaluation criteria when comparing the advanced method to *DML* Kordopatis-Zilos et al., 2017b, *CNN* + *LSTM* Yaocong and Xiaobo, 2018, and *TIRI* Coskun et al., 2006. The advanced method achieves a *F1* = 0.99 compared to *F1* = 0.97 for DML, *F1* = 0.94 for CNN + LSTM and *F1* = 0.825 for TIRI.

4.2.3 Discussion

A global retrospective view on the investigated NN-based methods is presented in Figure 8 that is paired designed with Figure 6. It originates in 2016 and presents, for each analyzed year, the key conceptual ideas (the dark-blue, left block) as well as the methodological enablers in fingerprinting (the blue, right block). Note that in this case the fingerprint extraction and matching are merged (as they are tightly coupled).

The previous section brings to light that the NN-based fingerprinting is still an emerging research field. It inherits its methodological framework from conventional fingerprinting, while updating both the fingerprint extraction and matching.

Since 2016, fingerprint extraction gradually shifted from considering NN solution at an individual level (e.g., spatial or temporal features) to holistic, 3D Nets able to simultaneously capture integrated spatio-temporal features. Intermediate solutions, combining NN and conventional image processing tools (e.g., SURF, TIRI, or BoVW) are also encountered. The fingerprinting matching generally comes across with the fingerprinting extraction.

The experimental testbed principles are also inherited from the case of conventional methods. Yet, the datasets are different in their size as well as in the fact that experimenter generally creates the attacked versions of the video content (cf. The last 5 lines in Table 1). The evaluation criteria generally cover *Prec*, *Rec*, *F1* and *mAP*. This variety in experimental conditions makes impossible for an objective performance comparison to be stated.

Figure 8 and Section 4.2.2 demonstrate that the exploratory work of using NN in conjunction to conventional tools can be considered as successful and that the way towards effective NN—only solutions is open Li et al., 2021.

However, when comparing current day conventional to NN—based solutions, the quantitative results seem unbalanced in favor of conventional methods. Yet, quick conclusions should be avoided, as the datasets are of significantly different sizes and the task complexity is significantly different. The generic evaluation criteria introduced in Section 3 are seldom jointly evaluated, with each study focusing on a specific metric and/or a pair of metrics. Moreover, note that the computational complexity is

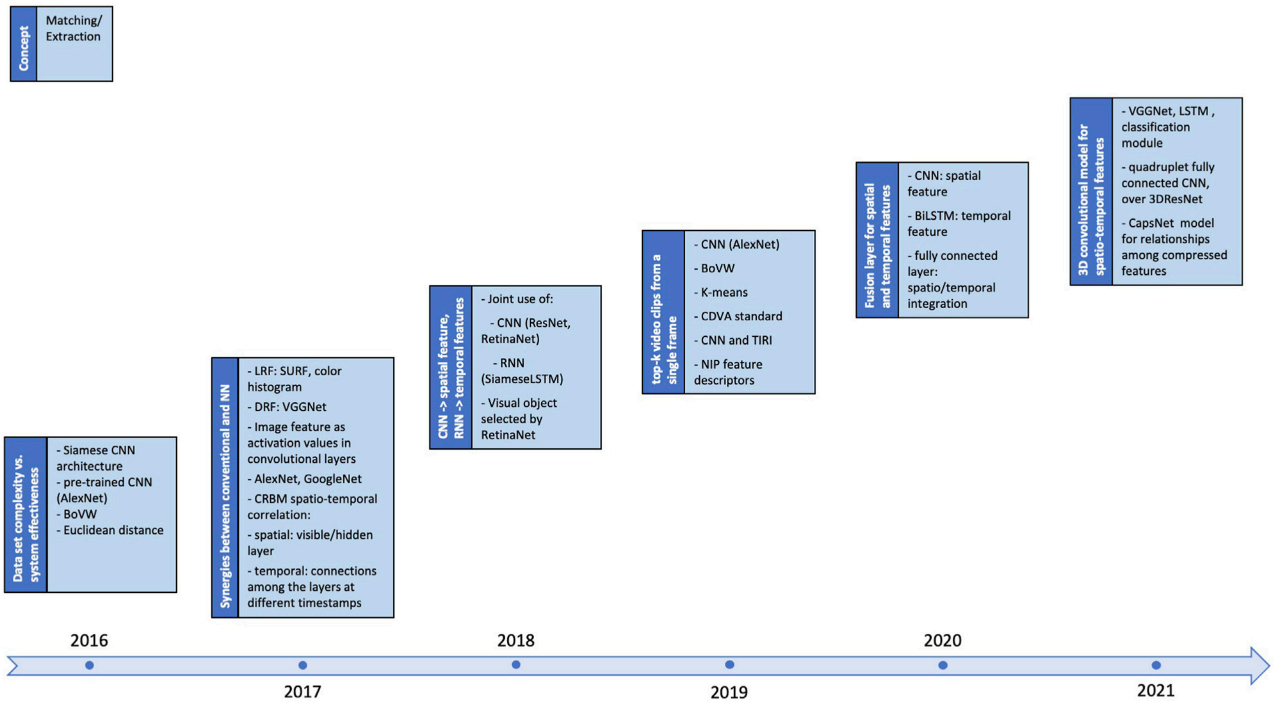
Concept
Matching/ Extraction
Data set complexity vs. system effectiveness
- Siamese CNN architecture
- pre-trained CNN (AlexNet)
- BoVW
- Euclidean distance
Synergies between conventional and NN
- LRF: SURF, color histogram
- DRF: VGGNet
- Image feature as activation values in convolutional layers
- AlexNet, GoogleNet
- CRBM spatio-temporal correlation:
- spatial: visible/hidden layer
- temporal: connections among the layers at different timestamps
CNN -> spatial feature, RNN -> temporal features
- Joint use of:
- CNN (ResNet, RetinaNet)
- RNN (SiameseLSTM)
- Visual object selected by RetinaNet
top-k video clips from a single frame
- CNN (AlexNet)
- BoVW
- K-means
- CDVA standard
- CNN and TIRI
- NIP feature descriptors
Fusion layer for spatial and temporal features
- CNN: spatial feature
- BiLSTM: temporal feature
- fully connected layer: spatio/temporal integration
3D convolutional model for spatio-temporal features
- VGGNet, LSTM , classification module
- quadruplet fully connected CNN, over 3DResNet
- CapsNet model for relationships among compressed features
2016
2017
2018
2019
2020
2021

FIGURE 8
Evolution of the NN methods.

seldom discussed as a true evaluation criterion, thus making a sharp decision even more complicated.

5 Challenges and perspectives

Fingerprint challenges and trends are structured according to the constraints set by current day video production and distribution, and to the new applicative fields in which fingerprinting can help, as discussed in Sections 5.1 and 5.2, respectively.

5.1 Stronger constraints on video fingerprinting properties

Whilst not being either exhaustive or detailed, Section 4 is meant to bring light on the very complex, fragmented yet well-structured landscape of the video fingerprinting methods, as illustrated in Figures 5–8.

Despite clear incremental progress, achieving the ultimate method for generic video content (TV/movies/social media) fingerprinting is still an open research topic that will continuously be faced to new challenges in terms of: 1) video content size and typology, 2) complexity of near-duplicated copies, 3) compressed stream extraction, and 4) energy consumption reduction.

First, the size of video content is expected to continuously increase. Social media, personalized video content, business oriented video content (e.g., videoconferencing) are expected to lead soon to an average of 38 h a week of video consumption per person in US Delloite, 2022. Such quantity of content is expected to be processed, stored and retrieve without impairing the user experience, hence new challenges in reducing the complexity of fingerprinting matching are expected to be set.

Secondly, the image/video software editing solutions as well as professional video transmission technologies (such as broadcasting, encoding, or publishing) will increase the number, the variety and the complexity of the near-duplicated copies to be dealt with. As for time being these near-duplicated copies are rather considered one-by-one and no attempt to exploit would-be statistical models unitary representing them, this trend is expected to increase the constraints on fingerprinting robustness.

Thirdly, although the video content is mainly generated in compressed format, just few partial results related to fingerprinting extraction directly from the stream syntax elements are reported Ngo et al., 2005, Li and Vishal, 2013, Ren et al., 2016, Schuster et al., 2017. This highly contrast with related applicative fields, like indexing and watermarking, where more advanced results are already obtained Manerba et al., 2008; Benois-Pineau, 2010, Hasnaoui and Mitrea, 2014.

Finally, video fingerprinting is also expected to take the challenge of reducing the computational complexity, following a green computing trend in video processing Ejembi and Bhatti, 2015, Fernandes et al., 2015, Katayama et al., 2016. This working direction is expected to be coupled to the previous one, namely designing green compressed video fingerprinting solutions.

With respect to the above-mentioned four items, short term research efforts are expected to address several incremental aspects, from both methodological and applicative standpoints. The former encompasses aspects such as the explicability of the NN-based results, the relationship between semantics, content, and the human visual system, the questionable possibility of modeling the modifications induced in near-duplicated content, … The latter is expected to investigate the very applicative utility of conventional performance criteria, the computational complexity balancing among extraction/detection in context of NN-based methods and massive datasets, the possibility of identifying a unique structure or a set of structures per performance criterion to be optimized, etc.

As a final remark, note that no convergence towards a theoretical model able to accommodate the current-day efforts can be identified and, in this respect, information theory, statistics and/or signal/image processing are expected to still be at stake during long-term research efforts. Such a theoretical model is expected to have different beneficial effects, from allowing a comparison among existing methods to be carried out with rigor to identifying the tools for answering the applicative expectancies and/or the theoretical bounds.

5.2 Emerging applicative domains

Fingerprinting benefits are likely to be become appealing for several new applicative domains, such as fake news identifying and tracking, unmanned vehicles video processing, metaverse content tracking, or medical imaging, to mention but a few. This extension rises new challenges not only in terms of applicative integration between fingerprinting and other technologies but also in terms of content type and composition.

In the sequel, we shall detail the cases of fingerprinting for visual fake news and for the video captured by unmanned vehicles.

5.2.1 Visual fake news

While the concept of *fake news* does not still have a sharp and consensual definition Katarya and Massoudi, 2020, it can be considered that, in the video context, it relates to the malicious creation of a new video content, whose semantic is not genuine and/or whose interpretation yields to false conclusion. The fake news creation starts generally from some original video content that is subsequently edited. Hence, such a problem is multifold and various types of solutions can be envisaged: detecting whether a content is modified or not, detecting the original content that has been manipulated, detecting the last authorized modification of the original content, etc. Consequently, various video processing paradigms can contribute (individually and/or combined) to elucidate some of these aspects, Lago et al., 2018, Zhou et al., 2020, Agrawal and Sharma, 2021, Devi et al., 2021.

For instance, video forensics are generally considered as a tool to identify content modification, solely based on the analyzed content. On its side, watermarking provides effective solutions for identifying video content modifications and/or the last authorized user but requires the possibility of modifying the original content prior to its distribution.

Video fingerprinting affords the detection of the original content that has been manipulated to create the fake news content. Figure 9 illustrates the case[3] where two video contents, from two different repositories, are combined to create a fake content. In this respect, the challenge of designing fingerprinting methods robust to content cropping is expected to be taken soon. This example shows that fingerprint is complementary to forensics. With respect to watermarking, fingerprinting has as main advantage is passive behavior (it does not require the original content to be modified).

Moreover, video fingerprinting is still expected to be complemented with security mechanisms, and blockchain (also referred to as Distributed Ledger Technologies—DLT) seems very promising in this respect.

In a nutshell, a blockchain is a distributed information storage technology, ensuring trust in the tracking and the authentication of the binary data exchanged in a decentralized, peer-to-peer network: even the smallest (1 bit) modification in a message can be identified. As such a bit sensitivity property is incompatible with the digital document tracking (where multiple digital representations can be associated to a same semantic), the blockchain principles should be coupled to the visual fingerprints that ensure robustness to modifications Allouche et al., 2021.

From a methodological standpoint, various potential solutions can be conceived, according to the targeted applicative trade-off. In this respect, the naïve solution would be to replace the DLT native hash function (*e.g.*, SHA256) by the fingerprinting extraction; yet, such a solution is likely to induce some pitfalls in the system security. Alternatively, the specification of management layers over existing DLT solution is also possible: while such an approach would not impact the DLT security, it is likely to drastically increase the system complexity (smart contract definition and deployment,

[3] This example is not based on any real situation.

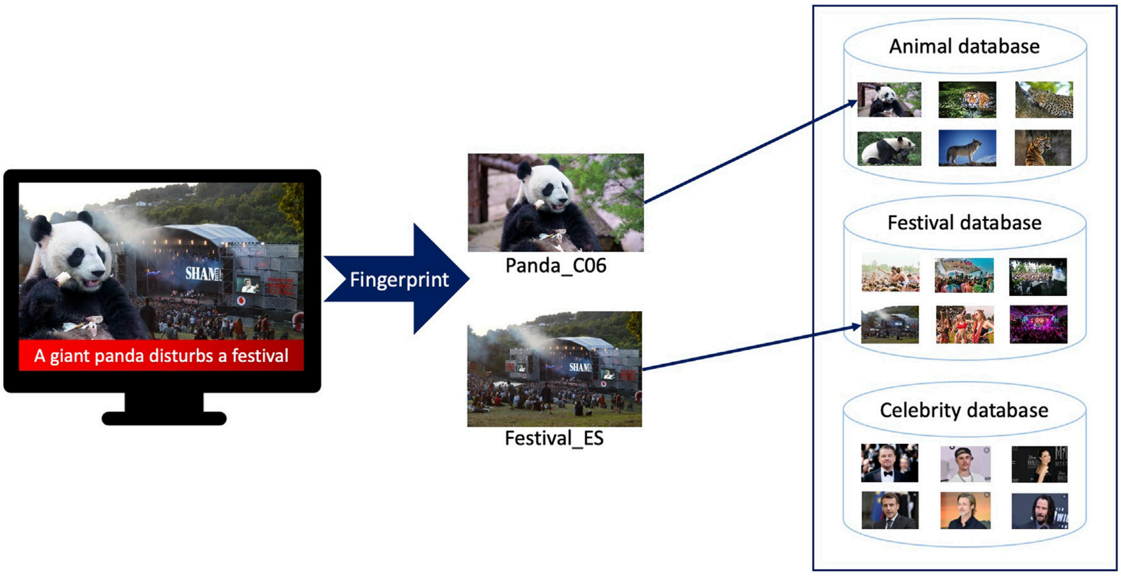

FIGURE 9
Fake video content can be tracked to its original sources thanks to fingerprinting.

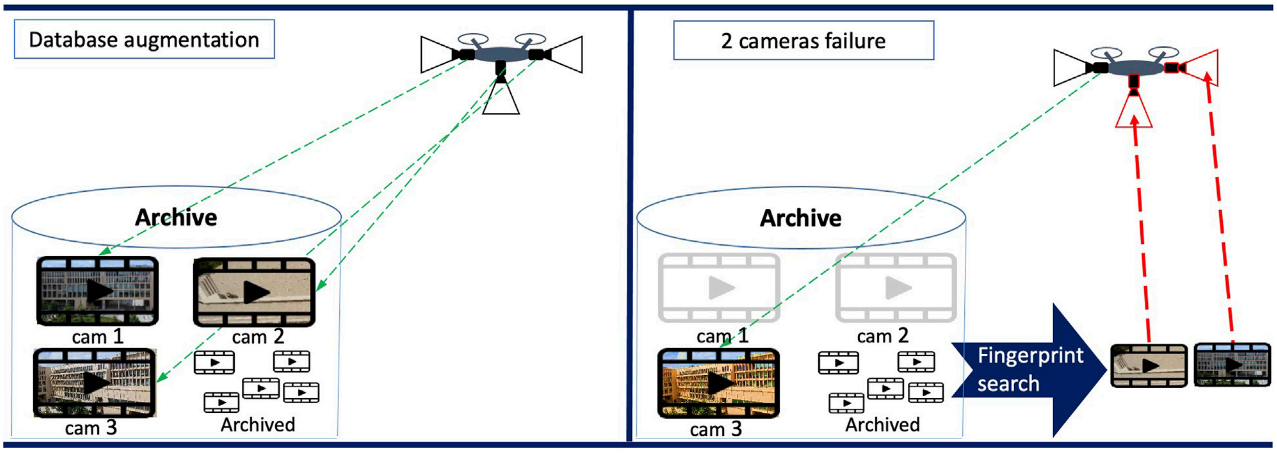

FIGURE 10
Usage of archive video for drone navigation in case of camera failure.

complex operation execution, on-chain/off-chain load balancing). Intermediate solution can also be thought.

5.2.2 Unmanned vehicles

Drones, robots, and autonomous cars are steadily increasing their applicative scope, thus rising new challenges in a large variety of research fields, including video processing. For instance, large video repositories with data produced by unmanned vehicles are expected to be organized soon, for serving different applicative scopes: a posteriori analysis/disambiguation in case accidents occur, real-time assistance in case of partial failures (on-board cameras are partially out of order), distributed cloud-to-edge computing, etc. Figure 10 illustrates the case in which two out of the three cameras available on a delivery drone are out of order. As the delivery trajectory can never be 100% reproducible, video fingerprinting can be an effective tool for searching a near-duplicated video content in the archive, starting from the camera video stream still in use.

In this respect, new challenges related to the very video content type, to its composition as well as to its security (content integrity) are expected to be taken.

First, the video captured by the unmanned vehicles is no longer expected to be optimized for the human visual system peculiarities and should answer a new set of requirements allowing for a better and safer navigation. In a context in which the very concept of just noticeable difference should be extended Jin et al., 2022, new types of local/global features, matched to the navigation task specificities, are expected to be designed and evaluated.

Secondly, multiple cameras are generally positioned on an unmanned vehicle on fixed positions, thus producing a set of video streams (*e.g.*, 6 to 12 streams according to the type of vehicle). As these video streams are spatially corelated and aligned in time, they are expected to permit the development of new fingerprinting approaches based on global features, rather than of features extracted at the level of each stream. The main difficulty related to the fact that the unmanned video streams are neither independent nor complying with the multi-views paradigm.

Finally, the content integrity issue can be dealt with by considering DLT or watermarking solutions.

6 Conclusion

The present paper provides a generic view on video fingerprinting: conceptual basis, evaluation framework, and methodological approaches are first studied.

They show that the fingerprint landscape is complex yet well structured around the main steps in the fingerprinting workflow: pre-processing of the video sequence, extraction of spatio-temporal information, aggregation of basic features into various derived representations, and matching. While this generic framework is set some 20 years ago, the NN advent positioned itself as a precious enabler in applicative-oriented optimizations. Moreover, both conventional and NN solutions can be integrated into global fingerprinting solutions that are able today to process datasets larger than 350,000 h of video while featuring *Prec* and *Rec* values larger than 0.9! This opens the door for effective solutions based on 3D Nets able to simultaneously capture integrated spatio-temporal features.

Moreover, fingerprinting is still an open to research topic. From both methodological and applicative standpoints, it is expected to encompass aspects such as the explicability of the NN-based results, the relationship between semantics, content, and the human visual system, or the questionable possibility of modeling the modifications induced in near-duplicated content. Extracting the fingerprinting directly from the compressed stream syntax elements and synergies with green encoding approaches are also to be dealt with in the near future.

New challenges in terms of applicative integration between fingerprinting and other technologies as well as in terms of content type and composition will be raised by emerging trends in video production and distribution, such as fake news content tracking, unmanned vehicles video processing, or metaverse content tracking.

Author contributions

MA collected the largest majority of references and is the main contributor for Section 4. MM is the main contributor for Section 2, 3, and 5. Both MA and MM contributed to manuscript writing (all sections), revision, read, and approved the submitted version.

Conflict of interest

The MA is financed by Vidmizer.

The remaining 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.

References

Abu-El-Haija, S., Kothari, N., Lee, J., Natsev, A., Toderici, G., Varadarajan, B., et al. (2016). *YouTube-8M: A large-scale video classification benchmark*". ArXiv, abs/1609.08675.

Agrawal, R., and Sharma, D. K. (2021). "A survey on video-based fake news detection techniques," in *8th international conference on computing for sustainable global development (INDIACom)*.

Allouche, M., Frikha, T., Mitrea, M., Memmi, G., and Chaabane, F. (2021). Lightweight blockchain processing. Case study: Scanned document tracking on tezos blockchain. *Appl. Sci. (Basel).* 11, 7169. doi:10.3390/app11157169

Anuranji, R., and Srimathi, H. (2020). A supervised deep convolutional based bidirectional long short term memory video hashing for large scale video retrieval applications. *Digit. Signal Process.* 102, 102729. doi:10.1016/j.dsp.2020.102729

Baris, C., Bulent, S., and Nasir, M. (2006). Spatio-temporal transform based video hashing. *IEEE Trans. Multimed.* 8 (6), 1190–1208. doi:10.1109/tmm.2006.884614

Basharat, A., Zhai, Y., and Shah, M. (2008). Content based video matching using spatiotemporal volumes. *Comput. Vis. Image Underst.* 110 (3), 360–377. doi:10.1016/j.cviu.2007.09.016

Bay, H., Tuytelaars, T., Gool, L. V., and Van Gool, L. (2008). Surf: speeded up robust features. *Comput. Vis. Image Underst.* 110 (3), 346–359. doi:10.1016/j.cviu.2007.09.014

Bend, J. (2015). The YLI-MED corpus: Characteristics, procedures, and plans. *Comput. Res. Repos. ICSI Tech. Rep. TR-15-001*, 1–46.

Benois-Pineau, J. (2010). "Indexing of compressed video: Methods, challenges, applications," in *International conference on image processing theory*, 3–4. Tools and Applications.

Carreira-Perpinán, M. A., and Raziperchikolaei, R. (2015). "Hashing with binary autoencoders," in *Proc. IEEE conf. Comput. Vis. Pattern recog.*, 557–566.

Coskun, B., Sankur, B., and Memon, N. (2006). Spatio-temporal transform based video hashing. *IEEE Trans. Multimed.* 8 (6), 1190–1208. doi:10.1109/tmm.2006.884614

Coudert, F., Benois-Pineau, J., Le Lann, P.-Y., and Barba, D. (1999). "Binkey: a system for video content analysis on the fly," in *Proceedings IEEE international conference on multimedia computing and systems*.

Cox, I., Miller, M., Bloom, J., Fridrich, J., and Kalker, T. (2007). *Digital watermarking and steganography*. Burlington, MA, US: Morgan Kaufmann.

Datar, M., Immorlica, N., Indyk, P., and Mirrokni, V. S. (2004). "Locality-sensitive hashing scheme based on p-stable distributions," in *Proceedings of the twentieth annual symposium on Computational geometry*, 253–262.

de Araújo, A. F., and Girod, B. (2018). Large-scale video retrieval using image queries. *IEEE Trans. Circuits Syst. Video Technol.* 28 (6), 1406–1420. doi:10.1109/tcsvt.2017.2667710

Delloite (2022). *The future of the TV and video landscape by 2030*. Available at: https://www2.deloitte.com/content/dam/Deloitte/be/Documents/technology-media-telecommunications/201809%20Future%20of%20Video_DIGITAL_FINAL.pdf.

Devi, S., Karthik, V., Baga, S., Bavatharani, V., and Indhumadhi, K. (2021). "Fake news and tampered image detection in social networks using machine learning," in *2021 third international conference on inventive research in computing applications (ICIRCA)*.

Do, M. N., and Vetterli, M. (2004). The contourlet transform: an efficient directional multiresolution image representation. *IEEE Trans. Image Process.* 14 (12), 2091–2106. doi:10.1109/tip.2005.859376

Douze, M., Gaidon, A., Jegou, H., Marszałek, M., and Schmid, C. (2008). INRIA-LEAR's video copy detection system. *TRECVID*.

Douze, M., Jégou, H., and Schmid, C. (2010). An image-based approach to video copy detection with spatio-temporal post-filtering. *IEEE Trans. Multimed.* 12, 257–266. doi:10.1109/tmm.2010.2046265

Duan, L. -Y., Lou, Y., Yan, B., Huang, T., Gao, W., Chandrasekhar, V., et al. (2019). Compact descriptors for video analysis: The emerging MPEG standard. *IEEE Multimed.* 26 (2), 44–54. doi:10.1109/mmul.2018.2873844

Ejembi, O., and Bhatti, S. (2015). "Go green with EnVI: the energy-video index," in *2015 IEEE international symposium on multimedia (ISM)*.

Esmaeili, M. M., Fatourechi, M., and Ward, R. K. (2011). A robust and fast video copy detection system using content-based fingerprinting. *IEEE Trans. Inf. Forensic. Secur.* 6 (1), 213–226. doi:10.1109/tifs.2010.2097593

Esmaeili, M. M., and Ward, R. K. (2010). Robust video hashing based on temporally informative representative images. *Proc. IEEE ICCE*, 179–180.

Fernandes, F., Ducloux, X., Ma, Z., Faramarzi, E., Gendron, P., and Wen, J. (2015). The green metadata standard for energy-efficient video consumption. *IEEE Multimed.* 22 (1), 80–87. doi:10.1109/mmul.2015.18

Fridrich, J., and Goljan, M. (2000). Robust hash functions for digital watermarking. *Proc. Int. Conf. Inf. Technol. Coding Comput.*, 178–183.

Garboan, A., and Mitrea, M. (2016). Live camera recording robust video fingerprinting. *Multimed. Syst.* 22, 229–243. doi:10.1007/s00530-014-0447-0

Gong, Y., Lazebnik, S., Gordo, A., and Perronnin, F. (2013). Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (12), 2916–2929. doi:10.1109/tpami.2012.193

Hadsell, R., Chopra, S., and LeCun, Y. (2006). "Dimensionality reduction by learning an invariant mapping," in *Proc. IEEE conf. Comput. Vis. Pattern recog.*, 1735–1742.

Hampapur, A., and Bolle, R. M. (2001). "Comparison of distance measures for video copy detection," in *International conference on multimedia and expo*, 737–740.

Hasnaoui, M., and Mitrea, M. (2014). Multi-symbol QIM video watermarking. *Signal Process. Image Commun.* 29 (1), 107–127. doi:10.1016/j.image.2013.07.007

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *IEEE conference on computer vision and pattern recognition (CVPR)*, 770–778.

Heikkila, M., Pietikainen, M., and Schmid, C. (2009). Description of interest regions with local binary patterns. *Pattern Recognit.* 42 (3), 425–436. doi:10.1016/j.patcog.2008.08.014

Heilbron, F. C., Escorcia, V., Ghanem, B., and Niebles, J. C. (2015). "ActivityNet: A large-scale video benchmark for human activity understanding," in *Proc. IEEE conf. Comput. Vis. Pattern recognit. (CVPR)*, 961–970.

Hochreiter, S., and Schmidhuber, J. (1997). "Long short-term memory," in *Neural comput*, 1735–1780.

Hong, L., Lu, H., and Xiangyang, X. (2010). *SVD-SIFT for web nearduplicate image detection*. IEEE ICIP, 1445–1448.

Hou, Y., Wang, X., and Liu, S. (2015). "Multiple features video fingerprint algorithm based on optical flow feature," in *International conference on computers, communications, and systems (ICCCS)*, 159–162.

Hu, Y., and Lu, X. (2018). Learning spatial-temporal features for video copy detection by the combination of CNN and RNN. *J. Vis. Commun. Image Represent.* 55, 21–29. doi:10.1016/j.jvcir.2018.05.013

Idris, F., and Panchanathan, S. (1997). Review of image and video indexing techniques. *J. Vis. Commun. Image Represent.* 8 (2), 146–166. doi:10.1006/jvci.1997.0355

Jegou, H., Douze, M., and Schmid, C. (2008). "Hamming Embedding and Weak geometry consistency for large scale image search," in *Proceedings of the 10th European conference on Computer vision*.

Jiang, S., Su, L., Huang, Q., Cui, P., and Wu, Z. (2012). "A rotation invariant descriptor for robust video copy detection," in *The era of interactive media*.

Jiang, Y. G., Jiang, Y., and Wang, J. (2014). "VCDB: A large-scale database for partial copy detection in videos," in *European conference on computer vision (ECCV)*.

Jiang, Y. G., and Wang, J. (2016). Partial copy detection in videos: a benchmark and an evaluation of popular methods. *IEEE Trans. Big Data* 2 (1), 32–42. doi:10.1109/tbdata.2016.2530714

Jiang, Y. G., Wu, Z., Wang, J., Xue, X., and Chang, S. F. (2018). Exploiting feature and class relationships in video categorization with regularized deep neural networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 40 (2), 352–364. doi:10.1109/tpami.2017.2670560

Jiang, Y. G., Ye, G., Chang, S.-F., Ellis, D., and Loui, A. C. (2011). Consumer video understanding: A benchmark database and an evaluation of human and machine performance. *Proc. 1st ACM Int. Conf. Multimed. Retr. Art. No.* 29.

Jin, J., Zhang, X., Fu, X., Zhan, H., Lin, W., Lou, J., et al. (2022). Just noticeable difference for deep machine vision. *IEEE Trans. Circuits Syst. Video Technol.* 32 (6), 3452–3461. doi:10.1109/tcsvt.2021.3113572

Jin, Z., Lin, Y., and Cai, D. (2014). Density sensitive hashing. *IEEE Trans. Cybern.* 44 (8), 1362–1371. doi:10.1109/tcyb.2013.2283497

Karen, S., and Andrew, Z. (2014). *Very deep convolutional networks for large-scale image recognition*. arXiv preprint arXiv:1409.1556.

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., and Fei-Fei, L. (2014). Large-scale video classification with convolutional neural networks. *IEEE Conf. Comput. Vis. Pattern Recognit.*, 1725–1732.

Katarya, R., and Massoudi, M. (2020). "Recognizing fake news in social media with deep learning: A systematic review," in *4th international conference on computer, communication and signal processing (ICCCSP)*, 1–4.

Katayama, T., Shi, W., Song, T., Shimamoto, T., and Leu, J.-S. (2016)."NearReference frame selection algorithm of HEVC encoder for low power video device," in *2016 2nd international conference on intelligent green building and smart grid (IGBSG)*.

Kim, C., and Vasudev, B. (2005). Spatiotemporal sequence matching for efficient video copy detection. *IEEE Trans. Circuits Syst. Video Technol.* 15, 127–132. doi:10.1109/tcsvt.2004.836751

Kingma, D., and Jimmy, Ba. (2014). *Adam: A method for stochastic optimization*. arXiv preprint arXiv:1412.6980.

Kordopatis-Zilos, G., Papadopoulos, S., Patras, I., and Kompatsiaris, Y. (2017a). "Near-duplicate video retrieval by aggregating intermediate cnn layers," in *International conference on multimedia modeling*, 251–263.

Kordopatis-Zilos, G., Papadopoulos, S., Patras, I., and Kompatsiaris, Y. (2017b). "Near-duplicate video retrieval with deep metric learning," in *IEEE international conference on computer vision workshops (ICCVW-2017)*, 347–356.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). "ImageNet classification with deep con- volutional neural networks," in *Advances in neural information* *processing sys- tems 25: 26th annual conference on neural information processing systems*, 1106–1114.

Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., and Serre, T. (2011). "HMDB: a large video database for human motion recognition," in *International conference on computer vision* (IEEE), 2556–2563.

Lago, F., Phan, Q.-T., and Boato, G. (2018). "Image forensics in online news," in *2018 IEEE 20th international workshop on multimedia signal processing (MMSP)*.

Law-To, J., Buisson, O., Gouet-Brunet, V., and Boujemaa, N. (2007a). "Video copy detection on the Internet: the challenges of copyright and multiplicity," in *IEEE int'l conf multimed expo*, 2082–2085.

Law-To, J., Joly, A., and Boujemaa, N. (2007b). Muscle-VCD-2007: a live benchmark for video copy detection Available at: http://www-rocq.inria.fr/imedia/civr-bench/.

Lee, S., and Yoo, C. D. (2008). Robust video fingerprinting for content-based video identification. *IEEE Trans. Circuits Syst. Video Technol.* 18 (7), 983–988. doi:10.1109/tcsvt.2008.920739

Lee, S., and Yoo, C. D. (2006). "Video fingerprinting based on centroids of gradient orientations," in *Proc. IEEE int. Conf. Acoust., speech and signal process. (ICASSP)*, 2.II

Lefebvre, F., Chupeau, B., Massoudi, A., and Diehl, E. (2009). "Image and video fingerprinting: Forensic applications," in *Proc. SPIE* (San Jose, CA, US: Media Forensics and Security), 7254.

Li, M., and Vishal, M. (2013). Compact video fingerprinting via structural graphical models. *IEEE Trans. Inf. Forensic. Secur.* 8, 1709–1721. doi:10.1109/tifs.2013.2278100

Li, M., and Vishal, M. (2012). Robust video hashing via multilinear subspace projections. *IEEE Trans. Image Process.* 21 (10), 4397–4409. doi:10.1109/tip.2012.2206036

Li, X., Xu, L., and Yang, Y. (2021). Compact video fingerprinting via an improved capsule net. *Syst. Sci. Control Eng.* 9 (1), 122–130. doi:10.1080/21642583.2020.1833782

Li, Y. (2018). "Deep content: Unveiling video streaming content from encrypted WiFi traffic," in *IEEE 17th international symposium on network computing and application*, 1–8.

Li, Y. N., and Chen, X. P. (2017). "Robust and compact video descriptor learned by deep neural network," in *IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 2162–2166.

Lin, T. Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). *Focal loss for dense object detection"*. arXiv:1708.02002.

Liong, V. E., Lu, J., Tan, Y. P., and Zhou, J. (2017). Deep video hashing. *IEEE Trans. Multimed.* 19 (6), 1209–1219. doi:10.1109/tmm.2016.2645404

Liong, V. E., Lu, J., Wang, G., Moulin, P., and Zhou, J. (2015). "Deep hashing for compact binary codes learning," in *Proc. IEEE conf. Comput. Vis. Pattern recognit. (CVPR)*, 2475–2483.

Liu, J., Huang, Z., Cai, H., Shen, H. T., Ngo, C. W., and Wang, W. (2013). Near-duplicate video retrieval: Current research and future trends. *ACM Comput. Surv.* 45 (4), 1–23. Art. No. 44. doi:10.1145/2501654.2501658

Liu, M. (2018). "Content-based video copy detection using binary object fingerprints," in *IEEE international conference on signal processing, communications and computing (ICSPCC)*.

Liu, M., Po, L. M., Ur Rehman, Y. A., Xu, X., Li, Y., and Feng, L. (2019). Video copy detection by conducting fast searching of inverted files. *Multimed. Tools Appl.* 78, 10601–10624. doi:10.1007/s11042-018-6639-4

Liu, W., Wang, J., Ji, R., Jiang, Y. G., and Chang, S. F. (2012). "Supervised hashing with kernels," in *Proc. IEEE conf. Comput. Vis. Pattern recognit. (CVPR)*, 2074–2081.

Lu, J. (2009). Video fingerprinting for copy identification: from research to industry applications. *Proc. SPIE - Media Forensics Secur. XI* 7254.

Ma, C., Gu, Y., Gong, C., Yang, J., and Feng, D. (2018). Unsupervised video hashing via deep neural network. *Neural process. Lett.* 47 (3), 877–890. doi:10.1007/s11063-018-9812-x

Manerba, F., Benois-Pineau, J., Leonardi, R., and Mansencal, B. (2008). Multiple moving object detection for fast video content description in compressed domain. *EURASIP J. Adv. Signal Process.*, 231930–232015. doi:10.1155/2008/231930

Mansencal, B., Benois-Pineau, J., Bredin, H., and Quenot, G. (2018). *IRIM at TRECVID 2018: Instance search*. TRECVID.

Mao, J., Gang, X., Weiguo, S., Yahong, H., and Zhiguo, Q. (2016). A method for video authenticity based on the fingerprint of scene frame. *Neurocomputing* 173, 2022–2032. doi:10.1016/j.neucom.2015.09.001

Ngo, C. W., Yu-Fei, M., and Hong, J. Z. (2005). Video summarization and scene detection by graph modeling. *IEEE Trans. Circuits Syst. Video Technol.* 15, 296–305. doi:10.1109/tcsvt.2004.841694

Nie, X., Liu, J., Wang, Q., and Zeng, W. K. (2015). Graph-based video fingerprinting using double optimal projection. *J. Vis. Commun. Image Represent.* 32, 120–129. doi:10.1016/j.jvcir.2015.08.001

Nie, X., Jing, W., Ma, L. Y., Cui, C., and Yin, Y. (2017a). "Two-layer video fingerprinting strategy for near- duplicate video detection," in *IEEE international conference on multimedia & expo workshops (ICMEW)*.

Nie, X. S., Yin, Y. L., Sun, D., Liu, Chui C. R., and Cui, C. (2017b). Comprehensive feature-based robust video fingerprinting using tensor model. *IEEE Trans. Multimed.* 19 (4), 785–796. doi:10.1109/tmm.2016.2629758

Nie, X., Zhou, X., Shi, Y., Sun, J., and Yin, Y. (2021). Classification-enhancement deep hashing for large-scale video retrieval. *Appl. Soft Comput.* 109, 107467. doi:10.1016/j.asoc.2021.107467

Oostveen, J., Kalker, T., and Haitsma, J. (2002). "Feature extraction and a database strategy for video fingerprinting," in *Proceedings of the 5th international conference on recent advances in visual information systems*, 2314, 117–128. Lecture Notes In Computer Science.

Open Video (2022). Open Video dataset. Available at: www.open-video.org.

Ouali, C., Dumouchel, P., and Gupta, V. (2017). "Robust video fingerprints using positions of salient regions," in *IEEE international conference on acoustics, speech and signal processing (ICASSP)*.

Ouyang, J., Coatrieux, G., and Shu, H. (2015). Robust hashing for image authentication using quaternion discrete Fourier transform and log-polar transform. *Digit. Signal Process.* 41, 98–109. doi:10.1016/j.dsp.2015.03.006

Raginsky, M., and Lazebnik, S. (2009). "Locality-sensitive binary codes from shift-invariant kernels," in *Advances in neural information processing systems*, 1509–1517.

Ren, D., Zhuo, L., Long, H., Qu, P., and Zhang, J. (2016). "MPEG-2 video copy detection method based on sparse representation of spatial and temporal features," in *IEEE second international conference on multimedia big data*.

Roover, C. D., Vleeschouwer, C. D., Lefebvre, F., and Macq, B. (2005). Robust video hashing based on radial projections of key frames. *IEEE Trans. Signal Process.* 53 (10), 4020–4037. doi:10.1109/tsp.2005.855414

Sabour, S., Frosst, N., and Hinton, G. E. (2017). Dynamic routing between capsules. *Adv. neural Inf. Process. Syst.* 30.

Sarkar, A., Ghosh, P., Moxley, E., and Manjunath, B. S. (2008). "Video fingerprinting: features for duplicate and similar video detection and query-based video retrieval," in *Multimed content access algorithms syst II*, 68200E.

Schuster, R., Shmatikov, V., and Tromer, E. (2017). "Beauty and the Burst: Remote identification of encrypted video streams," in *26th USENIX security symposium*, 1357–1374.

Seidel, C. (2009). "Content fingerprinting from an industry perspective," in *IEEE international conference on multimedia and expo*.

Shen, F., Shen, C., Liu, W., and Shen, H. T. (2015). "Supervised discrete hashing," in *Proc. IEEE conf. Comput. Vis. Pattern recognit. (CVPR)*, 37–45.

Shikui, W., Yao, Z., Ce, Z., Changsheng, X., and Zhenfeng, Z. (2011). Frame fusion for video copy detection. *IEEE Trans. Circuits Syst. Video Technol.* 21 (1), 15–28. doi:10.1109/tcsvt.2011.2105554

Simonyan, K., and Zisserman, A. (2014). "Very deep convolutional networks for large-scale image recognition", arXiv preprint arXiv:1409.1556.

Singh, P., Verma, V. K., Rai, P., and Namboodiri, V. P. (2019). "HetConv: Heterogeneous kernel-based convolutions for deep CNNs," in *IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 4830–4839.

Sivic, J., and Zisserman, A. (2003). "Video google: A text retrieval approach to object matching in videos," in *Computer vision, IEEE international conference*, 1470–1477.

Sodagar, I. (2011). The MPEG-DASH standard for multimedia streaming over the Internet. *IEEE Multimed.* 18 (4), 62–67. doi:10.1109/mmul.2011.71

Song, J., Zhang, V., Li, V., Gao, L., Wang, M., and Hong, M. (2018). Self-supervised video hashing with hierarchical binary auto-encoder. *IEEE Trans. Image Process.* 27 (7), 3210–3221. doi:10.1109/tip.2018.2814344

Soomro, K., Zamir, A. R., and Shah, M. (2012). *UCF101 - a dataset of 101 human actions classes from videos in the wild*. arXiv preprint arXiv:1212.0402.

Statista (2022). Statista. Available at: https://www.statista.com/statistics/267222/global-data-volume-of-internet-video-to-tv-traffic/.

Su, X., Huang, T., and Gao, W. (2009). Robust video fingerprinting based on visual attention regions. *IEEE Int'l Conf. Acoust. Speech Signal Process* 109 (1), 1525–1528.

Sun, R., Xiaoxing, Y., and Jun, G. (2017). Robust video fingerprinting scheme based on contourlet hidden Markov tree model. *Optik* 128, 139–147. doi:10.1016/j.ijleo.2016.09.105

Szegedy, C. (2015). "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1–9.

Tan, H. K., Ngo, C. W., Hong, R., and Chua, T. S. (2009). *Scalable detection of partial near-duplicate videos by visual-temporal consistency*. MM'09.

Taylor, G. W., Hinton, G. E., and Roweis, S. T. (2007). "Modeling human motion using binary latent variables," in *Proc. Advances in neural information processing systems.*

Thomas, R. M., and Sumesh, M. S. (2015). A simple and robust colour based video copy detection on summarized videos. *Procedia Comput. Sci.* 46, 1668–1675. doi:10.1016/j.procs.2015.02.106

Thomee, B., Shamma, D. A., Friedland, G., Elizalde, B., Ni, K., Poland, D., et al. (2016). YFCC100M: The new data in multimedia research. *Commun. ACM* 59 (2), 64–73. doi:10.1145/2812802

Trecvid (2022). trecvid. Available at: https://trecvid.nist.gov.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., and Manzagol, P. A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.* 11, 3371–3408.

Wang, J., Kumar, O., and Chang, S. (2010). "Semi-supervised hashing for scalable image Retrieval," in *Proceedings of the IEEE conf. Comput. Vis. Pattern recognit.*, 3424–3431.

Wang, L., Bao, Y., Li, H., Fan, X., and Luo, Z. (2017). "Compact CNN based video representation for efficient video copy detection," in *International conference on multimedia modeling*, 576–587.

Wang, R. B., Chen, H., Yao, J. L., and Guo, Y. T. (2016). "Video copy detection based on temporal contextual hashing," in *IEEE second international conference on multimedia big data*.

Wary, A., and Neelima, A. (2019). A review on robust video copy detection. *Int. J. Multimed. Inf. Retr.* 8, 61–78. doi:10.1007/s13735-018-0159-x

Weiss, Y., Torralba, A., and Fergus, R. (2009). "Spectral hashing," in *Advances in neural information processing systems*, 1753–1760.

Wu, X., Hauptmann, A. G., and Ngo, C. W. (2007b). "Practical elimination of near-duplicates from web video search," in *Proceedings of the 15th ACM international conference on multimedia*, 218–227. MM '07.

Wu, X., Ngo, C. W., Hauptmann, A. G., and Tan, H. K. (2009). Real-time near-duplicate elimination for web video search with content and context. *IEEE Trans. Multimed.* 11 (2), 196–207. doi:10.1109/tmm.2008.2009673

Wu, X., Zhao, W., and Ngo, C. W. (2007a). "Near-duplicate keyframe retrieval with visual keywords and semantic context," in *Proc. of the 6th ACM international conference on image and video retrieval (CIVR'07)*, 162–169.

Xinwei, L., Chen, G., Yi, Y., and Lianghao, X. (2021). Video fingerprinting based on quadruplet convolutional neural network. *Syst. Sci. Control Eng.* 9 (1), 131–141. doi:10.1080/21642583.2020.1822946

Yang, B., Gu, F., and Niu, X. (2006). "Block mean value based image perceptual hashing," in *IIH-MSP'06 international conference on intelligent information hiding and multimedia signal processing* (IEEE), 167–172.

Yang, G., Chen, N., and Jiang, Q. (2012). A robust hashing algorithm based on SURF for video copy detection. *Comput. Secur.* 31, 33–39. doi:10.1016/j.cose.2011.11.004

Yaocong, H., and Xiaobo, L. (2018). Learning spatial-temporal features for video copy detection by the combination of CNN and RNN. *J. Vis. Commun. Image Represent.* 55, 21–29. doi:10.1016/j.jvcir.2018.05.013

Yuan, F., Po, L. M., Liu, M. Y., Xu, X. Y., Jian, W. H., Wong, K., et al. (2016). Shearlet based video fingerprint for content-based copy detection. *J. Signal Inf. Process.* 7, 84–97. doi:10.4236/jsip.2016.72010

Zhang, C., Lin, Y., Zhu, L., Liu, A., Zhang, Z., Huang, F., et al. (2019). CNN-VWII: An efficient approach for large-scale video retrieval by image queries. *Pattern Recognit. Lett.* 123, 82–88. doi:10.1016/j.patrec.2019.03.015

Zhao, W. L., Wu, X., and Ngo, C. W. (2010). On the annotation of web videos by efficient near duplicate search. *IEEE Trans. Multimed.* 12, 448–461. doi:10.1109/tmm.2010.2050651

Zhao, Y., Liu, G., Dai, Y., and Wang, Z. (2008). Robust hashing based on persistent points for video copy detection. *Proc. Int. Conf. Comput. Intell. Secur. (CIS)* 1.

Zhao, Y., Wang, S., Zhang, X., and Yao, H. (2013). Robust hashing for image authentication using zernike moments and local features. *IEEE Trans. Inf. Forensic. Secur.* 8 (1), 55–63. doi:10.1109/tifs.2012.2223680

Zhixiang, C., Lu, J., Feng, J., and Zhou, J. (2018). Nonlinear structural hashing for scalable video search. *IEEE Trans. Circuits Syst. Video Technol.* 28, 1421–1433. doi:10.1109/tcsvt.2017.2669095

Zhou, J., Pun, C-M., and Tong, Y. (2020). "News image steganography: A novel architecture facilitates the fake news identification," in *IEEE international conference on visual communications and image processing (VCIP).*

Zhou, Z., Chen, J., Yang, C. N., and Sun, X. (2019). Video copy detection using spatio-temporal CNN features. *IEEE Access* 7, 100658–100665. doi:10.1109/access.2019.2930173

Glossary

AUC Area under the curve

BoVW Bag of Visual Words

BRIEF Binary Robust Independent Elementary Features

CCV Columbia Consumer Video

CDVA Compact Descriptors for Video Analysis

CDVS Compact Descriptors for Visual Search

CEDH Classification-Enhancement Deep Hashing

CGO Centroids of Gradient Orientations

CNN Convolutional Neural Network

CPU Computing Processing Unit

CRBM Conditional Restricted Boltzmann Machine

CS-LBP Center-symmetric Local Binary Patterns

DCT Discrete Cosine Transform

DeepH Deep Hashing

DML Deep Metric Learning

DOP Double Optimal Projection

DRF Deep Representation Fingerprint

DWT Discrete Wavelet Transform

FAST Features from Accelerated Segment Test

F_1 F_1 score

FCVID Fudan-Columbia Video Dataset

FPR False Positive Rate

GPU Graphical Processing Unit

HetConv-MK heterogeneous convolutional multi-kernel

HMDB Human Metabolome Database

HOG Histogram of Oriented Gradient

LCS Longest Common Subsequence

LRF Low-level Representation Fingerprint

LSH Locality Sensitive Hashing

LSTM Long Short-Term Memory

mAP mean Average Precision

ML Machine Learning

MLP Multi-Layer Perceptron

NDCR Normalized Detection Cost Rate

NIST National Institute of Standards and Technology

NIP Nested Invariance Pooling

NN Neural Network

ORB descriptor Oriented Fast and Rotated Brief descriptor

P_{fa} Probability of false alarm

P_{md} Probability of missed detection

PCA Principal Component Analysis

Prec Precision

RAQ Randomized Adaptive Quantizer

Rec Recall

RMI Relative Mean Intensity

RNN Recursive Neural Network

ROC Receiver Operating Characteristic

SCNN Siamese Convolutional Neural Network

SDH Supervised Discrete Hashing

SIFT Scale-Invariant Feature Transform

SSCA Sub-Band Coefficient Amplitudes

SURF Speeded Up Robust Features

TF-IDF term frequency–inverse document frequency

TLS Transport Layer Security

TRECVID TREC Video Retrieval Evaluation

VCDB Large-Scale Video Copy Detection Database

VWII Visual Word Inverted Index

WPA-2 Wi-Fi Protected Access 2.