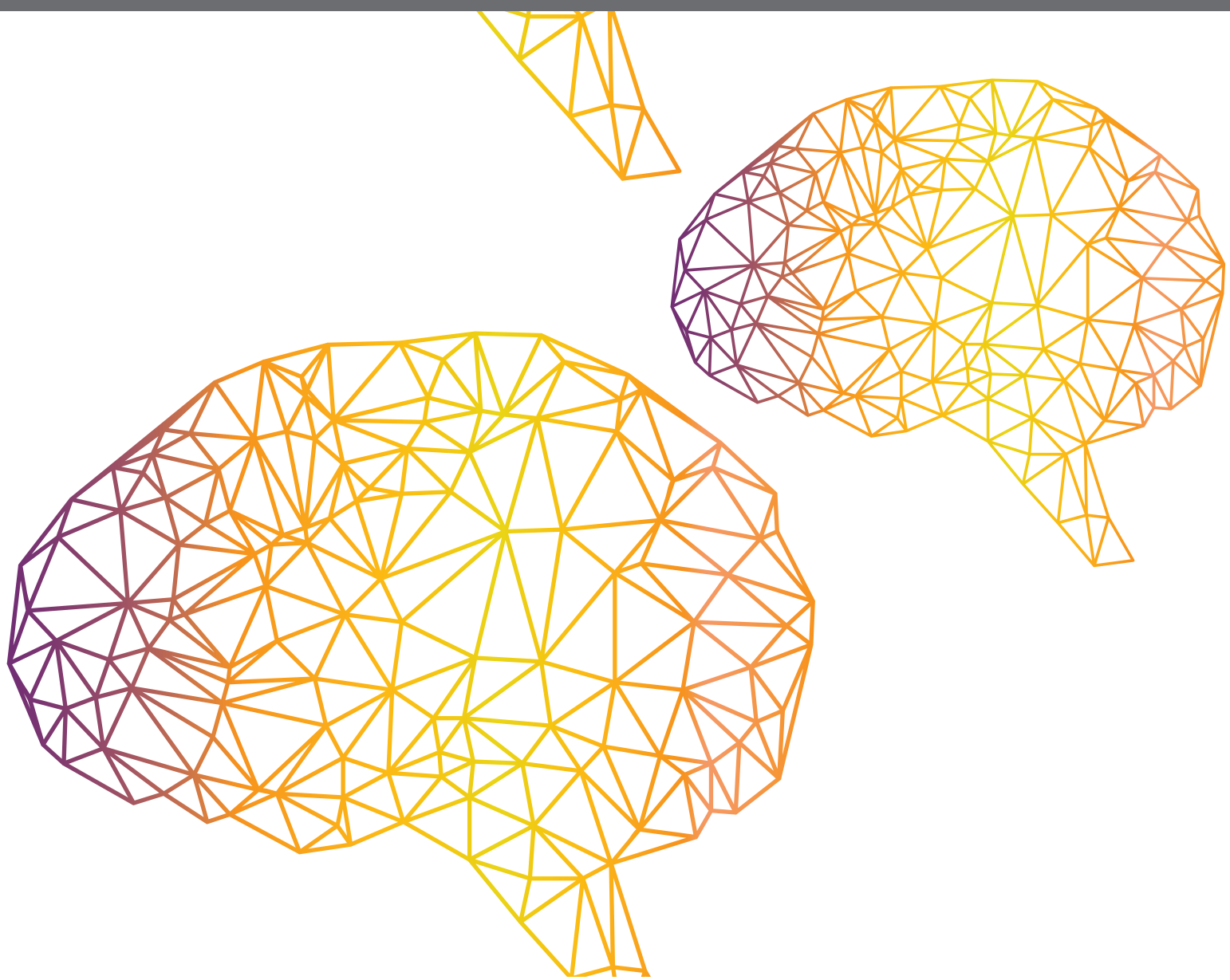
INTELLIGENCE AND SAFETY FOR HUMANOID ROBOTS: DESIGN, CONTROL, AND APPLICATIONS

EDITED BY: Zhihong Tian, Zhaoquan Gu, Yanhua Li, Wei Shi, Keke Tang and Zheng Wang

PUBLISHED IN: Frontiers in Neurorobotics



frontiers Research Topics

frontiers

ISSN 1664-8714
ISBN 978-2-88974-288-2
DOI 10.3389/978-2-88974-288-2

About Frontiers

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

Frontiers Journal Series

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

Dedication to Quality

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

What are Frontiers Research Topics?

Frontiers Research Topics are very popular trademarks of the Frontiers Journals Series: they are collections of at least ten articles, all centered on a particular subject. With their unique mix of varied contributions from Original Research to Review Articles, Frontiers Research Topics unify the most influential researchers, the latest key findings and historical advances in a hot research area! Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers Editorial Office: frontiersin.org/about/contact

INTELLIGENCE AND SAFETY FOR HUMANOID ROBOTS: DESIGN, CONTROL, AND APPLICATIONS

Topic Editors:
Zhihong Tian, Guangzhou University, China
Zhaoquan Gu, Guangzhou University, China
Yanhua Li, Worcester Polytechnic Institute, United States
Wei Shi, Carleton University, Canada
Keke Tang, Guangzhou University, China
Zheng Wang, Southern University of Science and Technology, China

Citation: Tian, Z., Gu, Z., Li, Y., Shi, W., Tang, K., Wang, Z., eds. (2022). Intelligence and Safety for Humanoid Robots: Design, Control, and Applications. Lausanne: Frontiers Media SA. doi: 10.3389/978-2-88974-288-2

Table of Contents

frontiers in Neurorobotics

EDITORIAL
published: 20 December 2021
doi: 10.3389/fnbot.2021.808369

Editorial: Intelligence and Safety for Humanoid Robots: Design, Control, and Applications

Zhaoquan Gu[1], *Keke Tang*[1]*, *Zheng Wang*[2], *Yanhua Li*[3], *Wei Shi*[4] *and Zhihong Tian*[1]

[1] *Cyberspace Institute of Advanced Technology, Guangzhou University, Guangdong, China,* [2] *Department of Mechanical and Energy Engineering, Southern University of Science and Technology, Guangdong, China,* [3] *Computer Science Department, Worcester Polytechnic Institute, Worcester, MA, United States,* [4] *School of Information Technology, Carleton University, Ottawa, ON, Canada*

Keywords: robotics, intelligence, safety, design, control

Editorial on the Research Topic

Intelligence and Safety for Humanoid Robots: Design, Control, and Applications

Humanoid robots attract growing research interests from different communities, both as tools for artificial intelligence research and neurocognitive interaction assessment, and as enabling technology with high societal impacts as personal robots for health, education, and entertainment. These robots, modeled on the basis of the embodiment of neural systems in software and hardware devices, are characterized by a high number of degrees of freedom, complex end effectors, and locomotion mechanisms on the hardware side. On the control side, they are characterized by the intrinsic and complex variety of their behavioral skills that are learned (imitation, reinforcement, statistical), for instance, learning-by-demonstration, data-driven approaches to humanoid arm programming, and the most recent AI-based approaches to manipulation and locomotion control.

Targeting at co-existing or physically interacting with humans, both intelligence and safety are of prominent importance for service-oriented humanoid robots, corresponding to the software and hardware levels, respectively. There has been recent progress in both areas, such as the quick emergence of learning-based artificial intelligence and soft robotics, bringing paradigm changes to brain-inspired humanoid design, control, and applications.

The Research Topic "Intelligence and Safety for Humanoid Robots: Design, Control, and Applications" includes 11 high-quality manuscripts that offer technological and methodological advances, as well as new features and approaches to robots.

Zhu et al. explored the challenging situation where robot grasping easily fails, and proposed a hybrid policy by combining visual cues and proprioception of the gripper for the effective failure detection and recovery in grasping. Particularly, they validated their system using a proprioceptive self-developed soft robotic gripper that is capable of contact sensing.

By investigating the fact that soft actuators usually fall short of motion accuracy and load capacity, or need large-size, bulky compressors, due to the limitations of materials and structures, Lin et al. developed a self-sensing vacuum soft actuation structure that acquires good balance among precision, output force, and actuation pressure.

Considering the huge demands for elderly-serving devices, especially for those with mobility impairment, Zhao et al. designed a novel smart robotic walker that supports multiple modes of interactions through voice, gait, or haptic touch, and allows intelligent control via learning-based methods to achieve mobility safety.

To improve the fast and stable walking ability of a humanoid robot, Tao et al. proposed a gait optimization method based on a parallel comprehensive learning particle swarm optimizer. Experimental results confirmed that the method achieved a quickly optimal solution, and the optimized humanoid robot possessed a fast and steady gait and flexible steering ability.

Edited and reviewed by:
Florian Röhrbein, Technische Universität Chemnitz, Germany

***Correspondence:**
Keke Tang
kktang@gzhu.edu.cn

Received: *03 November 2021*
Accepted: *18 November 2021*
Published: *20 December 2021*

Citation:
Gu Z, Tang K, Wang Z, Li Y, Shi W and Tian Z (2021) Editorial: Intelligence and Safety for Humanoid Robots: Design, Control, and Applications. Front. Neurorobot. 15:808369. doi: 10.3389/fnbot.2021.808369

To enable robots to execute pre-defined tasks based on simple and direct and explicit language instructions, Li et al. developed a framework that includes a language semantics module to extract the keywords, a visual object recognition module to identify the objects, and a similarity computation algorithm to infer the intention based on the given task.

Due to the surge in demand for data, computing resources, and network infrastructure, a scalable architecture was developed by Luo et al. to optimize the image processing efficiency and response rate of the robot's vision capabilities. The implemented Rinegan extension can improve the effectiveness and efficiency of image processing.

With the high demand on robot communication, Ge et al. proposed a traffic classification framework to effectively classify encrypted network traffic using a classification network structure combining a convolution neural network and long short-term memory network capturing traffic time and space characteristics. Experimental results demonstrate that the network can classify encrypted traffic and does not require manual feature extraction.

Peng et al. investigated the challenging situation of the limited number of wild images that are available for robotics to reconstruct 3D faces, and developed an accurate geometrical consistency modeling method based on B-spline parameter domain. Experimental results demonstrate the effectiveness of their method even in a challenging scenario, e.g., limited number of images with different head poses, illuminations, and expressions.

For the agency of a public opinion early warning task, an innovative cascade virus prediction model called CasWarn was proposed by Gao et al., which can be quickly deployed in intelligent agents to effectively predict the virality of public opinion information in different industries.

To exploit the importance of relation representation learning for knowledge graphs, an encoder-decoder model, which achieves better link prediction performance was proposed by Song et al. The core procedure of it is to embed the interaction between entities and relationships, and add a gate mechanism to control the attention mechanism.

Under the trend to integrate the ideas in game theory into the research of multi-robot systems, Jin et al. proposed a team-competition model to solve a dynamic multi-robot task allocation problem. Experimental results under many different cases demonstrate the effectiveness of their method.

AUTHOR CONTRIBUTIONS

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

FUNDING

This work was supported in part by the National Key R&D Program of China (no. 2019YFB1706003), the National Natural Science Foundation of China (U20B2046, 61902082, and 51975268), the Guangdong Basic and Applied Basic Research Foundation (2020A1515110997), the Science and Technology Program of Guangzhou (202002030263, 202102010419, and 202102010507), the Guangdong Higher Education Innovation Group (2020KCXTD007), the Guangzhou Higher Education Innovation Group (202032854), SUSTECH-AISONO Joint Lab Grant, and the SUSTECH Education Endowment.

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

frontiers in Neurorobotics

ORIGINAL RESEARCH
published: 22 October 2020
doi: 10.3389/fnbot.2020.575889

A Smart Robotic Walker With Intelligent Close-Proximity Interaction Capabilities for Elderly Mobility Safety

Xiaoyang Zhao[1†], Zhi Zhu[1†], Mingshan Liu[1†], Chongyu Zhao[1], Yafei Zhao[2], Jia Pan[1], Zheng Wang[2,3] and Chuan Wu[1*]*

[1] Department of Computer Science, Faculty of Engineering, The University of Hong Kong, Hong Kong, Hong Kong,
[2] Department of Mechanical Engineering, Faculty of Engineering, The University of Hong Kong, Hong Kong, Hong Kong,
[3] Department of Mechanical and Energy Engineering, Southern University of Science and Technology, Shenzhen, China

The elderly population has rapidly increased in past years, bringing huge demands for elderly serving devices, especially for those with mobility impairment. Present assistant walkers designed for elderly users are primitive with limited user interactivity and intelligence. We propose a novel smart robotic walker that targets a convenient-to-use indoor walking aid for the elderly. The walker supports multiple modes of interactions through voice, gait or haptic touch, and allows intelligent control via learning-based methods to achieve mobility safety. Our design enables a flexible, initiative and reliable walker due to the following: (1) we take a hybrid approach by combining the conventional mobile robotic platform with the existing rollator design, to achieve a novel robotic system that fulfills expected functionalities; (2) our walker tracks users in front by detecting lower limb gait, while providing close-proximity walking safety support; (3) our walker can detect human intentions and predict emergency events, e.g., falling, by monitoring force pressure on a specially designed soft-robotic interface on the handle; (4) our walker performs reinforcement learning-based sound source localization to locate and navigate to the user based on his/her voice signals. Experiment results demonstrate the sturdy mechanical structure, the reliability of multiple novel interactions, and the efficiency of the intelligent control algorithms implemented. The demonstration video is available at: https://sites.google.com/view/smart-walker-hku.

Keywords: elderly safety, human-robot interaction, intelligent control, falling protection, soft-robotic interface, coaxial front following, sound source localization

OPEN ACCESS

Edited by:
Jian Huang,
Huazhong University of Science and Technology, China

Reviewed by:
Wenxia Xu,
Wuhan Institute of Technology, China
Chi Zhu,
Maebashi Institute of Technology, Japan

***Correspondence:**
Zheng Wang
wangz@sustech.edu.cn
Chuan Wu
cwu@cs.hku.hk

[†] These authors have contributed equally to this work

Received: *24 June 2020*
Accepted: *07 September 2020*
Published: *22 October 2020*

Citation:
Zhao X, Zhu Z, Liu M, Zhao C, Zhao Y, Pan J, Wang Z and Wu C (2020) A Smart Robotic Walker With Intelligent Close-Proximity Interaction Capabilities for Elderly Mobility Safety. Front. Neurorobot. 14:575889. doi: 10.3389/fnbot.2020.575889

1. INTRODUCTION

Over the last few decades, the elderly population has rapidly increased globally and is expected to exceed 2 billion by 2050 (WHO, 2018). While constrained physical and cognitive abilities leave many older adults dependent, most of them prefer to continue living in their homes rather than moving into nursing homes, because the opportunity to stay in a familiar home environment offers them greater privacy and autonomy (Garçon et al., 2016).

For patients with Parkinson's disease, movement disorder can severely disrupt the performance of daily activities and increase the risk of falling. Despite various existing walkers are owned by

seniors, reported statistics show that 33% of people over 60 years fell at least once (Luz et al., 2017). We argue that intelligence is essential for an elderly walker to detect abnormal user behaviors and provide timely safety support, since primitive assistance devices, such as rollators and walkers, are much likely to fail (Bertrand et al., 2017). Exoskeleton (Tucker et al., 2019) is another approach with multiple robotic joints and links worn onto the user body, effective but less practical for daily wearing by older persons. Moreover, rather than merely using remote button (Glover, 2003), voice (Gharieb, 2006), or gesture (Gleeson et al., 2013) to achieve user interaction, older persons need various modes of human-robot interaction for convenience and efficiency. These motivate us to seek a solution of equipping the robotic walker with sufficient intelligence and interaction to guarantee mobility safety of older users.

In this paper, we propose a novel smart robotic walker that targets a convenient-to-use indoor walking safety aid for the elderly. Present-day assistant devices require attentive control of the user while moving (Di et al., 2015; Xu et al., 2018), which could raise safety issues for many elderly people with executive dysfunction or dementia. Although a few studies have investigated the task enabling the walker to follow behind the user (Moustris and Tzafestas, 2016), the problem is simplified since the human intention is known a posteriori by inspection of his/her trajectory. We take the approach of adopting a co-axial differential drive with sufficient braking force and enabling our walker to monitor and predict the movement trend of the user by detecting gait posture, our walker can then automatically move in front of the user, providing mobility support. With the walker moving in the front, we can enforce the elderly walking in a forward-learning position, preventing retropulsion falls, while our walker can support propulsion falls; with auto moving functionality, our walker alleviates the older users from attentive control of the walker.

As a service robot for the older users, the user interface (UI) provides the fundamental information acquisition for any intelligence and human-robot interaction. The vast majority of existing service robots often choose a touch-screen panel for touch input (Hans et al., 2002; Graf, 2009). However, there are severe limitations for touch-screen UI, from not being able to provide user motion data, to only detecting user command with a pre-defined set of items. Given that soft robotics has become a new trend to design and fabricate robots from a very distinctive approach than conventional robotics (Yi et al., 2018), we propose to use a soft robotic layer to be the user interface in constructing the handles due to its inherent safety (lack of rigid components) (Chen et al., 2018) and intelligence add-ons. To measure user intention and detect emergency event (e.g., falling) in a timely manner, we embed a sensor network inside the soft chamber to monitor force pressure on the handles. After conditioning and asynchronous filtering of the pressure data, our walker generates the appropriate output for system execution to meet user demand or provide safety support.

We also consider the very likely scenario that the elderly user and the walker are located in different locations in a household (e.g., the walker being charged and the user in bed). The autonomous mobility of an elderly walker through user voice summoning becomes essential to provide ready assistance to users with mobility impairment, which is often neglected in existing design (Mukai et al., 2010; Xu et al., 2018). To localize the sound source for autonomous mobility, existing methods have used Time Difference of Arrival (TDOA) (Valin et al., 2003) or deep neural network (DNN) (Ma et al., 2018), which are often ineffective in long distance or a multi-room environment. Recently, reinforcement learning (RL) has been widely applied in robotics. The mobility system based on RL (Zhang et al., 2015), for the first time, learns robotic manipulator motion control solely based on visual perception. Tai et al. (2017) learn to navigate by training a policy end-to-end, but the solution is only validated on a robotic platform with low degrees of freedom. A novel DRL approach (Choi et al., 2019) with LSTM embedded is proposed to learn efficient navigation in a complex environment. For multi-robot motion control, a decentralized RL model is presented to learn a sensor-level collision avoidance policy in multi-robot systems (Fan et al., 2020). Domains like UAVs (Hu et al., 2020; Wan et al., 2020) and underwater vehicles (Carlucho et al., 2018; Chu et al., 2020) have also exploited RL for motion control for various purposes, e.g., robust flying, path planning and remote surveillance. In our work, we present a novel approach of exploiting mobility of the walker and RL techniques for efficient sound source localization (SSL).

We conduct extensive experiments to demonstrate the efficiency of our smart walker for elderly mobility safety in the following aspects:

(1) A sturdy mechanical structure that fulfills expected functionalities and supports a user of average weight in home scenarios or outdoor sites with slopes $\leq 16^{\circ}$, which outperforms the safety requirement of the related ISO standard (ISO, 2003).
(2) Ability to track the user in the front to provide close-proximity walking safety support and turn according to user's turning intention with small error, through detecting lower limb gait of the user.
(3) Soft robotic user interface with a finite-state machine (FSM) model to detect user intention and emergency event effectively, ensuring timely safety protection.
(4) Autonomous mobility through RL to locate the user (sound source) and navigate to the user, in a multi-room household with environmental noises, reverberations, and long distance (over 10 m).

2. MATERIALS AND METHODS

2.1. System Overview

An overview of the proposed smart robotic walker with novel functionalities is shown in **Figure 1**. Our walker consists of a sturdy body frame with sensors deployed at appropriate positions, a motion system with differential driver and emergency brake, and a soft robotic interface with haptic monitor. A user staying in a different room from where the walker is can summon it to come close with the help of RL-based SSL technique, and the brake will be activated to prevent slipping once user intention of entering the front-following

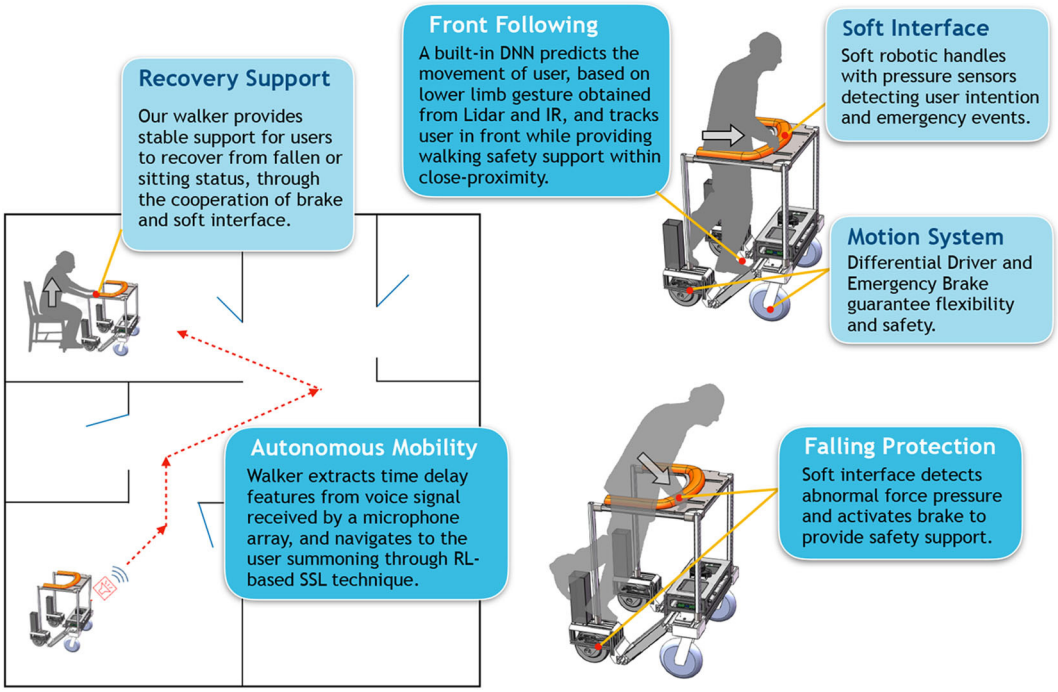


FIGURE 1 | System overview of the smart robotic walker.

status (AKA user walking stage) is detected by the soft interface; then the walker enters front-following walking-assistance status, when a DNN-based method predicts movement of the user to achieve smooth front following with close-proximity safety protection. The force pressure applied on the soft interface is always monitored and analyzed. If an emergency event is detected (e.g., falling), the brake will be activated, and the sturdy body frame and the soft interface will serve as a safety support.

In this section, we will briefly describe our objectives of hardware design and the technique roadmap of software control in order to achieve the proposed functionalities.

2.1.1. Hardware: Smart Walker With Soft Sensing Handle

The hardware design of our smart walker takes cues from both the conventional mobile robotic platform and traditional rollator. Several requirements are met in terms of structural stability, human-robot formation and human-robot interaction.

2.1.1.1. Structural stability

Designing a mechanical structure that is sturdy, strong, and agile enough to provide the safety support for the human user, is a fundamental requirement of the walker. Loading capacity should be sufficient to withstand a human user of $\leq$85 kg leaning against the top handle, with minimum tipping or sliding. To allow the device suitable for home usage, the maximum width of the walker should be no more than 700 mm, to ensure agility when navigating through narrow places.

2.1.1.2. Mobility system

Proposed functionalities require the walker to freely turn into any direction at any time. The walker should achieve zero turning radius or small radius turning in order to navigate within confined spaces. The standard solutions to omni-turning, i.e., omni-wheels or *Mecanum* wheels, have very limited rigidity against tipping disturbances. Besides, holonomic drive is not required as elders rarely walk sideways. In this work, to combine multi-terrain adaptability and standing support, a differential drive is ideal as it is the most widely used mechanism for moving robots and the most effective in terms of control strategy. To further increase safety, a brake mechanism that can respond to emergency in a timely manner is implemented as well.

2.1.1.3. Sensing network

The platform is equipped with a sensing network that enables the walker to perceive and interact with users. Sensing the status of the walker and the user are crucial for intelligent control to ensure user's safety and maximize system performance. This requires equipped sensors to achieve precision, timeliness, and robustness when dealing with various situations. We use optical, thermal, force, and vocal sensors to create a multi-modal sensing network, achieving effective human-robot interaction.

As the most direct way of haptic interaction, we adopt a novel soft-robotic technology (Chowdhary et al., 2019) to construct user interface on the handles for better and safer interacting experience comparing to the existing products.The handles are

designed to be soft with certain elasticity to withstand falling shock and provide comfortable touch. With the physical data (force, pressure, etc.) collected by high-sensitivity sensors inside the soft chamber, the system can acquire some useful information about the user all the time.

2.1.2. Software: Intelligent Control

We adopt learning-based methods to achieve intelligent control of the walker, based on signals obtained during human-robot interactions.

2.1.2.1. Soft haptic monitor

The soft interface on handles can detect user's intention and status. With pressure data collected from embedded sensors, we design a finite-state machine (FSM) model to analyze the temporal and spatial characteristics of the pressure data. Based on these characteristics, the intelligent framework is able to infer current status of the user and produce corresponding system actions to support the user in case of potential emergency (e.g., teetering, falling). The touch history of the handles can also be recorded as less privacy-sensitive health data for healthcare personnel to inspect.

2.1.2.2. Close-proximity coaxial front following

To achieve user tracking from the front, the gait information is collected by an infrared temperature sensor and a lidar sensor. We train a neural network (NN) model to learn the intention of the user from time-serial gait data. After obtaining the intention of the user, we compute a target position of the walker to ensure that one foot of the user is on the rear-wheel axis of the walker and the forward direction of the walker is parallel with the orientation of the foot. Such close-proximity and coaxiality between the walker and the user provide timely protection when the user is walking with the walker.

2.1.2.3. RL-based SSL

To achieve autonomous mobility in case that the walker and the user are located at different places (e.g., two rooms in a household), voice signals are monitored by deployed microphone array in a low-power state. Time-delay features of each microphone pair are extracted to estimate the direction of the sound source, and then the walker can move toward it, when the user summons the walker to come close with certain keywords. Before usage, we first train a NN model using supervised learning, on dataset collected from GSound simulator (Schissler and Manocha, 2011). Trained NN model is then fine-tuned through online RL, in daily usage of the walker. Note that during autonomous movement of the walker, once haptic touch is detected by the soft interface (i.e., when the walker reaches the user), the walker can provide sturdy support for the user to recover from sitting or lying status (if he/she has fallen on the floor).

2.2. Hardware Design

Hardware structure of the proposed smart robotic walker prototype is shown in **Figure 2**, consisting of the chassis and the upper handle, with basic parameters given in **Table 1**.

2.2.1. Body Frame and Actuation

The design of the body frame takes many safety issues into consideration. For static and dynamic stability, weight is concentrated low into the chassis. The resulting center of gravity (CG) height H_{CG} as estimated in Solidworks (Dassault Systems S.A.) is 156 mm, i.e., 18.8% of the walker's total height. The maximum tilt angle Φ_{max} before the CG goes over the supporting point (assuming the CG is located at the center point in the lateral direction) can then be calculated as:

$$\Phi_{max} = arctan\frac{l/2}{H_{CG}}$$

where l = 540 mm is the length of wheel base. Calculated Φ_{max} is approximately 60°, which is significantly larger than the tilt angle that human can incur before losing balance. Thus, before the tilt angle of walker reaches Φ_{max}, walker's weight (37 kg) would help elderly users resume to an upright position.

The walker protects the user from his/her front and both lateral sides to prevent falling, forming a "C" shape from the top view (**Figure 2C**). To maximize the space for the user to walk within the range of support (green area in **Figure 2C**) while constraining the maximum length of the walker, two wheel-hub motors (*Zhongling Technology* Ltd., China) are used as the rear driving wheels due to their lateral compactness. This results in providing a 420 × 436 mm walking space with the maximum walker width of 660 mm.

Both wheels are equipped with individual emergency brakes, modified from bike brakes, actuated by two individual linear actuators with 0.4 s of lead time to maximum break force, ensuring fast emergency response (Ahn et al., 2019).

2.2.2. Sensor Arrangement

To measure the movement of the walker, each wheel is equipped with a high-precision rotary encoder (4,096 ticks per revolution) and a wheel encoder odometer is implemented as well. The odometer yields the position of the walker $(x, y, \delta)_{walker}$, where $(x, y)_{walker}$ indicates the position and δ_{walker} indicates the orientation of the walker in the global frame, and the moving state $(v, \omega)_{walker}$, where v is the linear velocity and ω is the angular velocity of the walker, over time. An inertial measurement unit (IMU) is used to correct the orientation as it uses magnetometer to measure the yaw movement, which is more robust to dynamic disturbance as compared to the odometer.

Multiple sensors are used to acquire user states in order to achieve functions such as hand-free front following and SSL. The lidar used for leg detection is placed lower than the user's knees to ensure a good leg separation result. It is mounted in the front at a height of 410 mm, which is about the height of the upper calf of humans. An IR thermometer which has a 120° field view is placed at an angle that can cover most of the walking area and the user's front foot when walking. Four microphones are installed at the top of the walker with fixed spacing between each other. This will lead to different input signals at the microphones received from the same sound source, helping the controller to locate the source. All the readings from various sensors will be sent to a small form factor PC (*NUC*, *Intel* Co.) for processing.

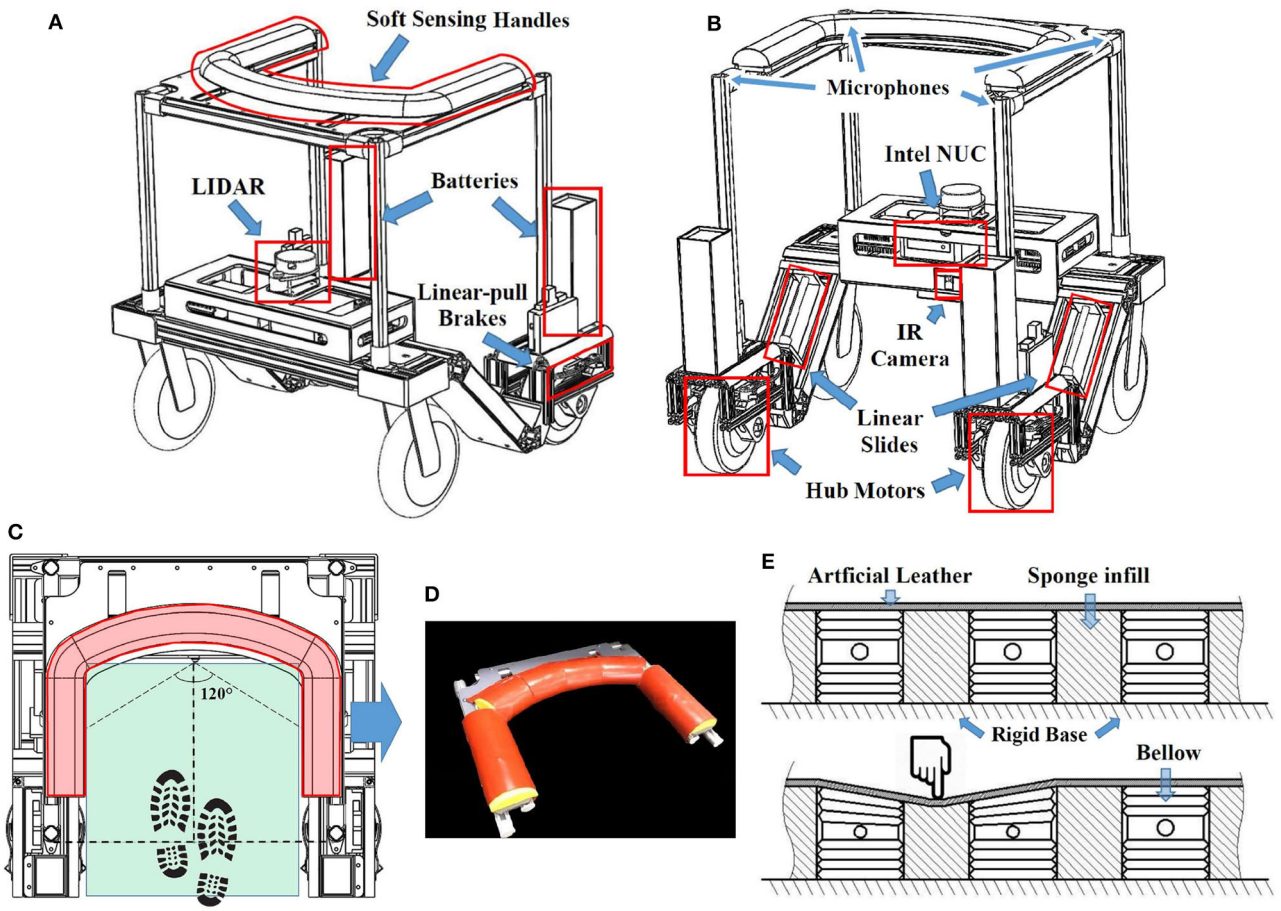


FIGURE 2 | The overview of the smart walker: **(A)** Front view; **(B)** Rear view; **(C)** Top view; **(D)** Soft sensing handle; **(E)** Section view of the handle.

TABLE 1 | Basic parameters of the smart walker.

Parameter	Value
Width * Length * Height	660 mm * 626 mm * 832 mm
Weight	37 kg
Minimum Turning Radius	622 mm
Speed	0–5.34 m/s

2.2.3. Soft Sensing Handle

We use a soft robotic layer to be the user interface when constructing the handles (**Figure 2D**) as a safe and friendly approach of interaction. In the core of a handle, the rigid base made of acrylic board is used to transfer the load to the main frame. The interior of the handle consists of multiple air pressure sensing bellows connected with air pressure sensors (**Figure 2E**), and sponge infills to provide a consistent handle surface. The pressure sensors detect the normal air pressure inside the bellow when the user is not holding the handle. When the user grabs onto the handle, a sudden change in air pressure will be read almost instantly through an MCU (Arduino Mega 2560). By deciphering the air pressure signal, the system calculates whether the user is or is not holding, or how firm the grip is. The handle is covered with a layer of artificial leather providing comfortable texture.

From each pressure bellow we can extract the information of pressure changes and the rate of changes. To enrich the sensing capability of the handle, the slight rigidity of the covering leather acts as a linkage between separated bellows. In this case, even when a force is not directly exerted on a bellow, it will also cause a less significant pressure change in the adjacent bellows (**Figure 2E**). Hence, one more dimension of signal, which is the position of the exerted force, is added to the sensing network. The reliability of this prototype handle will be tested in section 3.

2.3. Software Design

We next present the detailed design of software techniques to achieve intelligent control: (i) soft haptic monitor to recognize pressure pattern, (ii) close-proximity front following, and (iii) sound source localization through a reinforcement learning (RL) model.

2.3.1. Soft Haptic Monitor

Pressure data collected by sensors on the soft haptic handle with temporal and spatial characteristics can be analyzed to monitor the state of the user.

2.3.1.1. Position adjustment

After the walker navigates to proximity of the user (through SSL), the walker needs to know how itself is positioned against the user, and adjusts itself to be well-positioned as the user's walking support, before it enters the walking-support status (i.e., starting close-proximity front following of the user). **Figure 3A** illustrates the relative positioning of the user and the walker when the walker has moved up to the user through SSL. The user is likely to be in the Expected Zone because the walker keeps heading to the sound source. Even if the user is not in the Expected Zone, he/she can call the walker through voice control and the walker will adjust its direction again and eventually the user will be in the Expected Zone. The user can press his/her nearest part on the soft haptic handle to let the walker know where he/she is. Taking the midpoint of the line connecting the two sensors at the two ends of the handle as the origin O and the connection line as the y axis, a rectangular coordinate system is established. The connection line between the origin O and the center of a sensor has an angle α with the y axis. According to α, the walker will rotate at a calculated angle of β so that the walker and the user will be facing the same direction. The rotation angle β and the rotation direction are calculated as follows:

$$\beta = \begin{cases} \frac{\pi}{2} + \alpha & right, \quad 0 < \alpha < \frac{\pi}{2}, \\ \frac{3\pi}{2} - \alpha & left, \quad \frac{\pi}{2} \leqslant \alpha < \pi. \end{cases}$$

2.3.1.2. State monitor and falling protection

When the user is operating the walker, the soft haptic handle will monitor the state of the user. Different states are related to different intentions of the user. As an interface, the soft haptic handle collects the pressure data to infer the states of the user. For example, when there is a fall, there will be an abnormally high pressure or a rapid pressure change. Also, multiple sensors may be pressed since the user tends to lean on the handle when he/she falls. To detect user's intention and falling, the pressure data from different sensors of the soft haptic handle are analyzed independently or collectively: independent analysis concerns the pressure on one specific sensor, while the collective analysis focuses on comparing the pressure and changes of pressure on different sensors.

(1) **Abnormal Maximum Pressure:** One case of independent sensor analysis is that we calculate the maximum pressure value P_{max} of all pressure values of all sensors at the same time. According to P_{max}, the motor brake and the mechanical brake will be activated differently. The former can be activated instantly but is not sturdy enough to support the user, while the latter needs time to be fully activated but can provide more satisfying braking force. Especially, if P_{max} is normal (e.g., P_{max} is below the pressure generated by the user when he/she is grabbing the handle but is not

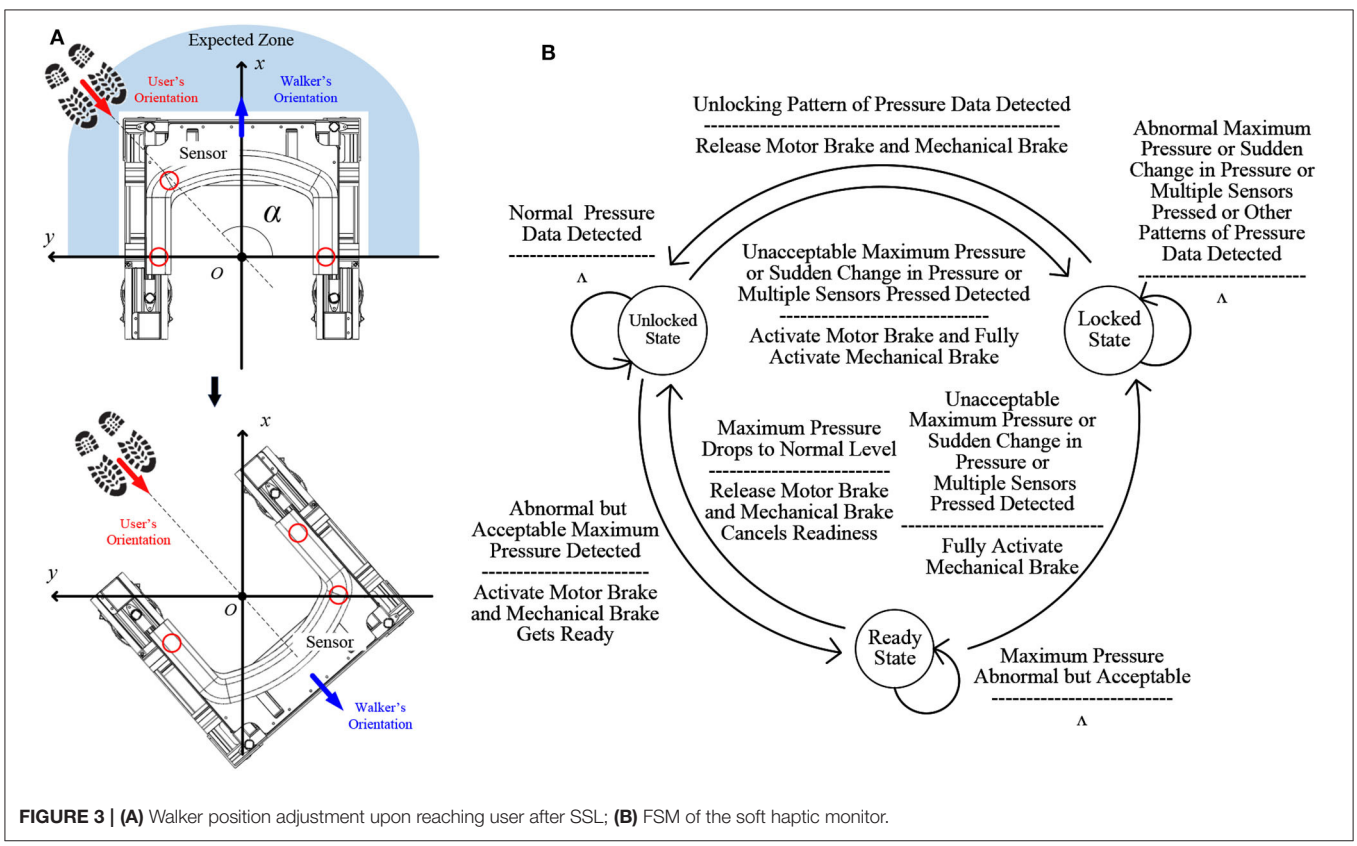


FIGURE 3 | (A) Walker position adjustment upon reaching user after SSL; **(B)** FSM of the soft haptic monitor.

leaning on it), the user is in a safe situation and the walker operates normally. When P_{max} becomes higher but still in an acceptable range, the user is at the boundary state between a normal situation and an accident, e.g., the case when the user stops and leans a little on the walker for a rest. In this range the user can recover to safe situation easily just by straighten his/her body without pushing the walker hard. In such a situation, the motor brake will be activated and the mechanical brake will change to a readiness state. The threshold that differentiates the normal range and the acceptable range is about 5% of user weight. If the value of P_{max} falls and becomes in the normal range again, the motor brake will be released and the mechanical brake will cancel the readiness state; the walker can move again. On the other hand, if P_{max} keeps raising and becomes unacceptable, representing the user leans more heavily on the walker, the mechanical brake will be fully activated and the walker will be locked to offer stable support for the user; the walker will not be unlocked until the soft haptic handle identifies an unlock intention from the user (to be detailed later). The threshold that differentiates the acceptable range and the unacceptable range is about 8.5% of user weight. The two thresholds can be changed by the user within 2% of his/her weight for better user experience. For privacy concern, user can choose one of the preset levels of weight range for threshold calculation. The preset levels consist of low weight (e.g., 40–55 kg), medium weight (e.g., 55–70 kg), and high weight (e.g., 70–85 kg). We use the average weight of each level range to decide the thresholds.

(2) **Abnormal Pressure Change:** Another independent analysis situation focuses on the maximum change rate of R'_{max} of all pressure values of all sensors. For each sensor, the current pressure value is used to minus the previous pressure value. Then the difference is divided by the time in between, about three sampling periods, to get all the change rates of all the sensors. Among all these change rates, the maximum change rate R'_{max} is calculated. With R'_{max}, the walker can detect an accident and offer protection sooner. For example, in cases of a stumble or a fall when the user is walking, the change rate R'_{max} is very large (regardless of P_{max}'s value); this is considered as an accident and both two brakes will be activated immediately, and the walker will be locked. The threshold for detecting the accident is about 15% of user weight per second. The sudden change of pressure can be detected within 0.2 s.

(3) **Multiple Sensors Simultaneously Pressed:** One collective analysis situation is that when too many sensors are being pressed at the same time, a falling tendency can be detected. Among the pressed sensors, if there are only sensors from the left and right sides of the soft handle, the user is inferred to be holding the two sides; if there are sensors from the front part of the soft haptic handle triggered, the user is assumed to lean on the front part of the handle and need support from the walker. In this situation, both brakes of the walker should be fully activated and the walker should be locked.

(4) **Pressure Change Comparison:** Another collective analysis situation focuses on temporal characteristics of pressed sensors. By comparing different pressure changes on different sensors, we can detect the strength and direction of the force applied by the user to the soft haptic handle. There are different cases: (a) When the two brakes are fully activated, if the user grabs the handle for recovery from fallen or sitting status, the direction and the strength of the force applied on the handle changes over time. If the change rates and the time for pressure value reaching the peak of different sensors are different or P_{max} is very high, we infer that the user needs support, and the two brakes will not be released. (b) If the user gently puts his/her hands on the left and right sides of the handle, the pressure data collected from different sensors on the two sides vary a bit over time, while the change rates and the time for reaching the peak will be similar among the sensors because the direction of the force remains unchanged and P_{max} is also at an acceptable level. We regard these characteristics as a signal of unlocking, and the walker will be unlocked and the two brakes will be released. (c) When the handle detects that the pressure of the left side is a bit higher than that on the right side, the user may want the walker to turn to the left. We will use such a pattern to decide turning radius of the walker.

By analyzing these pressure data patterns, falling and other user intentions can be recognized. Therefore, the walker can monitor the state of the user and provide falling protection, or respond to other user intentions.

2.3.1.3. The FSM of the soft haptic monitor

An FSM is embedded to control the working of the walker based on user states, as shown in **Figure 3B**. There are three states: *unlocked state*, *locked state*, and *ready state*. When the walker is in the *unlocked state*, the motor brake and the mechanical brake are released, and the walker is movable. When the Soft Haptic Monitor detects an accident, such as in cases of an unacceptable large maximum pressure P_{max}, a sudden change in the pressure, or multiple sensors are being pressed at the same time, the motor brake and the mechanical brake will be fully activated immediately and the walker will be in the *locked state*. At this state, the walker will be stable enough to offer support for the user, and will not respond to other signals except the unlocking pressure pattern. The other signals include not only those that activate the conversion from *unlocked state* to *locked state*, but also the patterns that cannot be analyzed as the unlocking pressure pattern such as the pressure pattern of recovery from falling. When the unlocking pattern is detected, the walker's state changes from the locked state to the unlocked state, when the two brakes will be released. When the walker is in the unlocked state, if an abnormal but acceptable P_{max} is detected, the walker will be in the *ready state*: at this state, the motor brake will be activated so the walker can not move; the mechanical brake will be ready for further protection. If P_{max} drops back to a normal value, the walker will go back to the unlocked state; if P_{max} keeps rising and finally becomes unacceptable, the walker

will enter the locked state. Since the mechanical brake is ready, it will take less time for the mechanical brake to be fully activated. Accidents like a sudden change in pressure and multiple sensors pressed will also activate this conversion.

2.3.1.4. Speed control

As the interface of the walker, the soft haptic handle allows the user to control the speed of the walker for effective walking assistance. Five speed-levels are preset and the user can select their preferred one by pressing two sensors on the handle. The two sensors are the sensors at the end of the left and right sides of the handle. One is used for acceleration and the other is used for deceleration. For the safety of the user, if one button is pressed, the walker will not respond to the other sensor, i.e., the walker will only respond to one button at a time. If one button is pressed and not released, the speed level of the walker will not keep changing. When the user presses the speed control button, there will be a unique peak of pressure value on that sensor while the pressure on other sensors will be weak. Therefore, this pattern of pressure data is different from other patterns and can be used while the FSM is monitoring the state of the user.

2.3.2. Close-Proximity Front Following

The walker tracks the user in the front through an NN-based intention detecting approach and generates movement through building a virtual target position. See **Figure 4A** for an illustration.

2.3.2.1. Sensor data processing

The IR sensor returns a 32 × 24 thermal image (see **Figure 4B** as an example), which can be flattened to a temperature vector $\vec{u}$ with dimension of 768. Compared with an RGB camera, a low-resolution IR sensor as a visual sensor is less costly and more privacy preserving. We normalize data in the temperature vector $\vec{u}$ into image data in a vector $\vec{g}$, where $u_{\min}$ and $u_{\max}$ are the maximum value and the minimum value in this vector, respectively:

$$g[i] = \frac{u[i] - u_{\min}}{u_{\max} - u_{\min}}.$$

Meanwhile, we identify the user leg positions in relation to the walker using data from the lidar sensor. We set a *baseline* to be the straight line connecting the two rear wheels of the walker, and the origin as the midpoint of this baseline. The forward direction is the positive x-axis, and the left direction is the positive y-axis. In this way, we define a coordinate system relative to the origin with the right-hand system, and calculate the coordinates of user leg (x, y) relative to the walker as follows:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos\theta_{walker} & -\sin\theta_{walker} \\ \sin\theta_{walker} & \cos\theta_{walker} \end{bmatrix} \begin{bmatrix} x_{leg} - x_{walker} \\ y_{leg} - y_{walker} \end{bmatrix}$$

where (x_{obj}, y_{obj}) and θ_{obj} describe the coordinates and orientation of user leg or the walker relative to the initial starting position of the walker, respectively.

We distinguish the computed leg positions into two classes using a k-means algorithm (Krishna and Murty, 1999), and use the prior condition that the y value of the left foot is more than that of the right foot to tell which class represents the left or right foot. The image of the two feet and the leg positions relative to the walker are output of sensor data processing.

2.3.2.2. Movement intention detection

Front following is essentially replacing the need of user pushing the walker, such that the walker can move automatically according to user's movement intention. We design an NN model to learn the relationship between user gait and user intention using time-serial data.

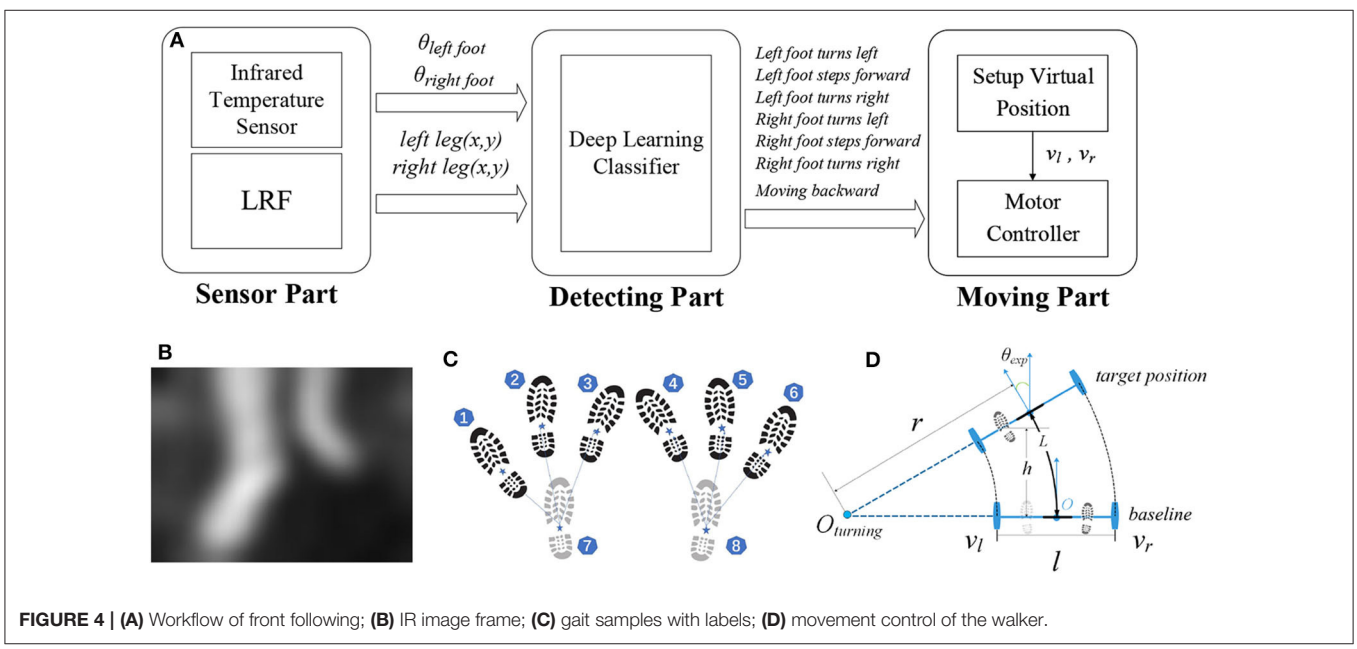


FIGURE 4 | (A) Workflow of front following; **(B)** IR image frame; **(C)** gait samples with labels; **(D)** movement control of the walker.

Each input sample to the NN is a sequence of 8 data points, where each data point contains vector $\vec{g}$ and leg positions (x_{leg}, y_{leg}) computed from the corresponding IR image. Each sample is labeled with the respective movement according to the user's gait, out of 6 cases (as illustrated in **Figure 4C**):

(1) Left foot turns left, with left foot in the front ① and right foot at the back ⑧;
(2) Left foot steps forward, with left foot in the front ② and right foot at the back ⑧;
(3) Left foot turns right, with left foot in the front ③ and right foot at the back ⑧;
(4) Right foot turns left, with left foot at the back ⑦ and right foot in the front ④;
(5) Right foot steps forward, with left foot at the back ⑦ and right foot in the front ⑤;
(6) Right foot turns right, with left foot at the back ⑦ and right foot in the front ⑥.

In addition, for the straight-backward case, we can easily tell whether the user is moving backward based on the lidar data, and hence it is not included as one output class from the NN. We use an NN consisting of two 512-unit hidden layers with *ReLU* as the activation function. The output is the probability distribution over the above six cases from a *Softmax* function.

2.3.2.3. Walker movement

Based on the leg positions (x_{leg}, y_{leg}) and inferred movement intention from the NN, we then compute a virtual position that the walker should move to, to achieve front following.

We use the turning radius r and a forward or backward distance h to decide the moving trajectory (arc length L) and the target position of the walker, as illustrated in **Figure 4D**. **Figure 4D** demonstrates the rear wheels in the origin and in the target position, respectively. The intersection of the extension of the rear wheels is the turning center $O_{turning}$. The distance from the center of rotation $O_{turning}$ to the center of the walker O is the radius of rotation r.

$$L = \frac{\theta_{exp} \cdot \pi \cdot r}{180} \qquad \theta_{exp} = \arcsin(\frac{h}{r})$$

l is the length of the driving wheel base; we ensure that there is one foot on the baseline and the orientation of the foot is parallel with the forward direction of the walker. There are three cases:

(a) When the NN output is case (1) or (6), the user is making a left or right turn. The range of this turning radius r is l to $2 \cdot l$, as determined by the probability p corresponding to this case: when the probability is large, the turning intention is obvious, and the walker is given a relatively small turning radius r; when the probability is relatively small but is still higher than that of the forward case, the user's intention is to move forward with a turn, and the walker will be given a relatively large turning radius r. When the turning center corresponding to the turning radius r is on the left side of the walk O, the turning radius r is positive; for symmetry, when the turning center $O_{turning}$ is on the right side of the center of the walker O, turning radius r is negative. Therefore, there are two cases of the turning radius:

$$r = \pm(l \cdot p + l)$$

(b) When the inferred movement is (2), (5), or moving backward, the turning radius r is positive infinity, i.e., $r = +\infty$.
(c) When the NN output is case (3) or (4), the user is marking a sharp right or left turn. Generally, these two situations occur after case (a) and are to further complete a sharp turning process. Therefore, we set the turning radius to one half of the walker width to provide a maximum rotation space. We have two cases of the turning radius: $r = \pm\frac{l}{2}$.

Due to differential drive control, two velocities v_l and v_r are calculated to control the walker to move to the target position. In practice, a walker is typically assigned with a linear velocity v. The two velocities v_l and v_r of each of the rear wheels can be calculated as follows:

$$v_l = v - \frac{v \cdot l}{2r} \quad v_r = v + \frac{v \cdot l}{2r}$$

2.3.3. RL-Based SSL

Our walker monitors audio signal received by a 4-channel microphone array, and can be waked up by customized keywords through a simple keyword spotting system. We choose 1 s of raw audio as input signal. In particular, 40 MFCC features are extracted from a frame of length 40 ms with a stride of 20 ms, which gives 1960 (40×49) features for each 1-s audio clip. We use Google speech commands dataset to train an NN model with three hidden layers, each with 128 neurons, to classify the incoming audio clips into one of the predefined words in the dataset, along with the default class "silence" (i.e., no word spoken) and "unknown" (i.e., word not in the dataset). Once waked up, our walker performs SSL following an RL model as follows (see **Figure 5** for an overview of the workflow).

2.3.3.1. State space

We define the input state s of the RL model to be an $m \times c$ matrix, where m is the total number of microphone pairs [e.g., $m = \binom{4}{2}$ if the walker is installed with four microphones], and c is the length of the feature vector of one pair. The input state indicates the time difference of arrival (TDOA) of sound signals received at each pair of microphones. The generalized cross correlation (GCC) of two sound signals is a measure of similarity. To accurately calculate the TDOA from the received signals, we firstly perform spectral subtraction (Martin, 1994) to raw audios for the purpose of de-noise and then calculate the GCC-Phase Transform (GCC-PHAT) (Knapp and Carter, 1976) as follows:

$$G_{PHAT}(f) = \frac{X_i(f)[X_j(f)]^*}{\left|X_i(f)[X_j(f)]^*\right|},$$

where f is a series of sound data after denoising, $X_i(f)$ denotes the Fourier transformation of the signal of the ith microphone,

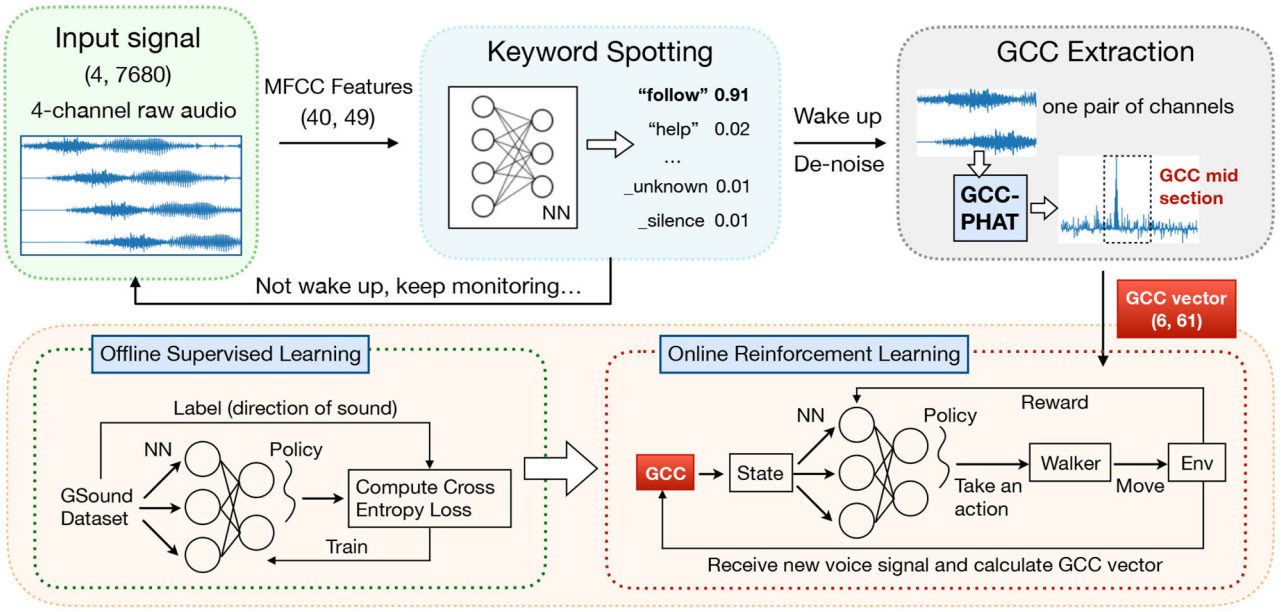


FIGURE 5 | The workflow of sound source localization (SSL) with a series of pre-processing and a reinforcement learning (RL) model.

and []* represents the complex conjugate. Then we compute the c-dimensional vector $\vec{s}_n$ in s as the c-dimensional subset of $G_{PHAT}(f)$, which indicates near-central part of a pair's GCC vector (Knapp and Carter, 1976). We empirically set $c = 61$ to make $\vec{s}_n$ contain the most useful information of TDOA.

The objective of our SSL module is to output the Direction of Arrival (DOA) of the sound source. Traditional DOA estimation approaches such as the Azimuth Method (Wikipedia contributors, 2020) are often unreliable under high reverberation conditions (as in our scenario) and with complex structures between microphones (as on our walker) (Xiao et al., 2015). We enable the walker to learn the nonlinear mapping from the input GCC features to the DOA output through RL.

2.3.3.2. Action space

After collecting state s, the controller selects a horizontal angle (i.e., the DOA) as action a based on policy $\pi_\gamma(a|s)$, which is a probability distribution over action space. The policy is produced by a neural network with γ as the set of parameters. We use a discrete action space including eight angles which are 45° apart: 0°, 45°, 90°, ..., 315°.

In output layer of the policy NN, we mask invalid actions, which points to a direction of obstacles within one meter from the walker, by setting their probability to 0 in the probability distribution. Then we re-scale the probabilities of all actions such that the sum still equals 1 (Bao et al., 2019). The walker will then move one meter toward the chosen direction. Note that, while approaching the user, the infrared distance sensors deployed in the front part of the walker will keep feeding distance data (to objects ahead) to the control module. If the walker detects that the user is 5–10 cm in front of it, the walker stops immediately to avoid collision with the user.

2.3.3.3. Reward

We carefully design a reward to use in RL, addressing variability of sound intensity and unknown location of the sound source. Consider a home with one hall and K rooms. At the beginning, the walker estimates the user to be in each room or hall with an equal probability, which is the confidence on which room (or hall) the sound source is located in. The walker updates its confidence on each room k in every time step:

$$Bel_{t+1}(k) = \rho(z_{t+1}, z'_{t+1}) * Bel_t(k),$$

where z_{t+1} is the vector of relative intensity collected at the microphones during real usage, z'_{t+1} is a relative intensity vector of simulated signals when putting sound source at the center of room k in our GSound simulator (which emulates the impact of reflection, diffraction and reverberation on sound propagation.), and ρ denotes the Pearson Coefficient. By comparing the similarity of received signals and simulated signals, the walker accumulates probability on each room. Note that after every update, we re-scale $Bel(k)$'s to make their sum remains to be 1.

Our reward functions are defined in four cases:

(1) When the walker is located in a different room from the sound source, the reward should encourage the walker to step out of the current room k:

$$r_t = 1 - \frac{d_k \sum_{i \in K \setminus k} Bel_t(i)}{Max}$$

where d_j is the shortest distance from the walker to the door of room k and Max is a constant value for normalization.

(2) When the walker is located in the hall, we encourage the walker to explore rooms with higher Bayesian Confidence:

$$r_t = 1 - \frac{\sum_{i \in K} d_i Bel_t(i)}{Max}$$

(3) When the walker is located in the same room with the sound source, the difference between direction estimations of consecutive inferences is used as the reward:

$$r_t = 1 - \frac{|a_t - a_{t-1}|}{315^\circ}$$

(4) When the walker has reached the sound source, it receives a relatively large reward $r_t = 5$.

2.3.3.4. Offline training and online tuning

The policy NN used by the walker is trained with SGD method (Sutton et al., 2000) by updating the NN parameters γ using policy gradients computed with samples $\langle s, a, r, s' \rangle$: a is the chosen direction for the walker to step forward, s and s' are the input state before and after action a is taken, and r is the reward computed in the current inference.

We collect samples using the GSound simulator (Schissler and Manocha, 2011) for offline training of the NN model, by specifying the locations of a sound source and recording received signals in arbitrary other locations in the multi-room setting. Then we use the trained model in the online setting: during real-world usage of the walker, the NN model is further fine-tuned with collected realistic samples.

3. EXPERIMENTS

In this section, we first conduct experiments to evaluate the mechanical structure of our walker and test sensors deployed on the soft handle. Usability test is also done to prove that our walker can achieve expected functionalities though intelligent control. Specifically, we evaluate walker's ability and efficiency to monitor user intention through soft interface, track user in front within close proximity, navigate to the user based on voice signals. All these demonstrate that our walker is sturdy and agile, with learning-based algorithms implemented to provide elderly users with sufficient mobility safety and effective human-robot interactions.

3.1. Mechanical Structure Test

The structural stability is validated according to requirements in ISO (ISO, 2003). For static stability, the ISO standard states that the walker should be placed on a slope in certain ways and it should remain stable without tipping. The slope angle requirement and the corresponding results under different test situations (see **Figure 6A**) are listed in **Table 2**. For dynamic stability (see **Figure 6B**), the self-modified brake mechanism is

FIGURE 6 | Stability tests; **(A)** Static stability test, from left to right: forward, backward, sideways; **(B)** Dynamic stability; **(C)** Tipping resistance test.

TABLE 2 | ISO.11199-2 stability test.

Test	ISO requirement	Result
Forward stability	≥15°	≥16°±1°
Backward stability	≥7°	≥14°±1°
Sideways stability	≥3.5°	≥12°±1°
Brake test	No sliding or 100 mm in not <1 min	No sliding

also reliable as the walker stays stationary when it is placed on a 6° slope with a subject weighing 63 kg leaning on it.

Moreover, a series of static load tests from the front and lateral directions are conducted (see **Figure 6C**) to find out how resistant the walker is to the external tipping force. The maximum force exerted on the handle, measured by a spring scale, when the walker starts tipping are 180 N forwardly and 182.5 N laterally. The actual maximum resistance should be larger since the fall rarely happens horizontally.

Overall, our walker passed all the tests with equal or better performance than that required in the ISO standard. It also has good tipping resistance against external push or pull force which plays an important roll in the fall prevention function. Therefore, our structure has strong advantages over those traditional mobility-assistant products, by implementing intelligent autonomous control based on better structural stability.

3.2. Soft Handle Evaluation

3.2.1. Sensitivity

To measure the modulus of each pressure sensing bellow and the relation between the load and the pressure detected on it, we implement a dedicated testing platform as shown in **Figure 7A**. Both pressure change ΔP and compression force F are measured together and the results are shown in **Figure 7B**. The maximum load force when the bellow reaches maximum compression (25 mm) is 140 N, giving a modulus of approximately 5.6 N/mm. The normal interaction force in the non-emergency scenarios should be <5 kg. The plot shows that ΔP corresponds well with the force exerted on it, laying a good foundation for monitoring user interaction for intelligent control.

3.2.2. Repeatability

The soft handle is expected to have long-term reliable performance and consistency. In the repeatability test, the bellow is gradually pressed until reaching maximum load F_{max}, then it was released to the normal state, and the same process was repeated. The results are passed through a 2nd Order Low-pass Filter (LPF) with a cutoff frequency of 1.50 Hz to remove high frequency noise. The repeatability test plot in **Figure 7C** shows that ΔP and F have an almost linear correlation with little deviation throughout the repeated compression. This also indicates that the bellow has good sealing as no negative effect due to air leakage showed up in the test.

3.3. Usability Test

We next perform usability tests to evaluate efficiency and practicality of our design in achieving the expected functionalities.

3.3.1. Soft Haptic Monitor

Experiments are conducted to collect the pressure data of different patterns of touching and grabbing the soft haptic handle. These patterns represent different interactions when the user is operating the walker. The pressure value is transformed from a physical quantity to a raw programmable digital quantity. We record the raw digital quantities of the pressure values from all sensors and their changes over time. A sliding mean filter is applied to pre-process the raw data. The window size of the filter is 5. **Figure 8** illustrates the data of different patterns. **Figures 8A–C** show three possible pressure data patterns when there is a fall or a potential fall. **Figures 8D,E** illustrate how to detect the unlock intention of the user. **Figure 8F** shows the position of each sensor on the handle.

Figure 8A shows that when the user is falling, he/she leans more and more onto the left front part of the handle. The maximum pressure P_{max} of all pressure values at the same time is detected on sensor 4, which rises slowly and eventually exceeds the unacceptable anomaly threshold of the pressure value 120 (e.g., about 8.5% of the user's weight). At the beginning of this process, the walker is in the *unlocked state*. P_{max} first exceeds the acceptable abnormal threshold of the pressure value 70 (e.g., about 5% of the user's weight), and then the walker goes into the *ready state* for further protection. There is a pause after that, corresponding to the situation where the user is not leaning more onto the handle. Then the user continues leaning more on the walker by putting most of his weight on the handle; P_{max} keeps increasing and finally reaches the unacceptable abnormal threshold, which brings the walker to the *locked state*.

Figure 8B corresponds to the case of a fall or stumble, with a sudden change in the pressure values. The fastest pressure change rate is about eight times of the change rate of P_{max} as in **Figure 8A**, while the peak pressure value in **Figure 8B** is only half of that in **Figure 8A**. Such a characteristic can be easily detected and the walker can react promptly to the locked state without waiting for the pressure to rise beyond the anomaly threshold.

Figure 8C shows that when the user falls onto the front part of the handle, multiple sensors (sensors 3–11) are being pressed simultaneously. When the user lies on the front part of the handle for rest and contacts multiple parts of the handle, the pattern of pressure data will be similar. Under these situations, the walker should offer falling protection.

Figures 8D,E show two patterns of user grabbing the handle when the walker is locked and stable. **Figure 8D** is the case that the user uses the walker for recovery support (from sitting or lying status), when the pressure changes of different sensors and the moments when their pressures reach the peaks are different. The reason is that during recovery, the gesture of the user keeps changing, resulting in the changes of the pressure values and the direction of the force applied to the handle. **Figure 8E** is the case that the user is ready to walk and gently puts his/her hands on the handle and grabs it. The pressure changes on different sensors

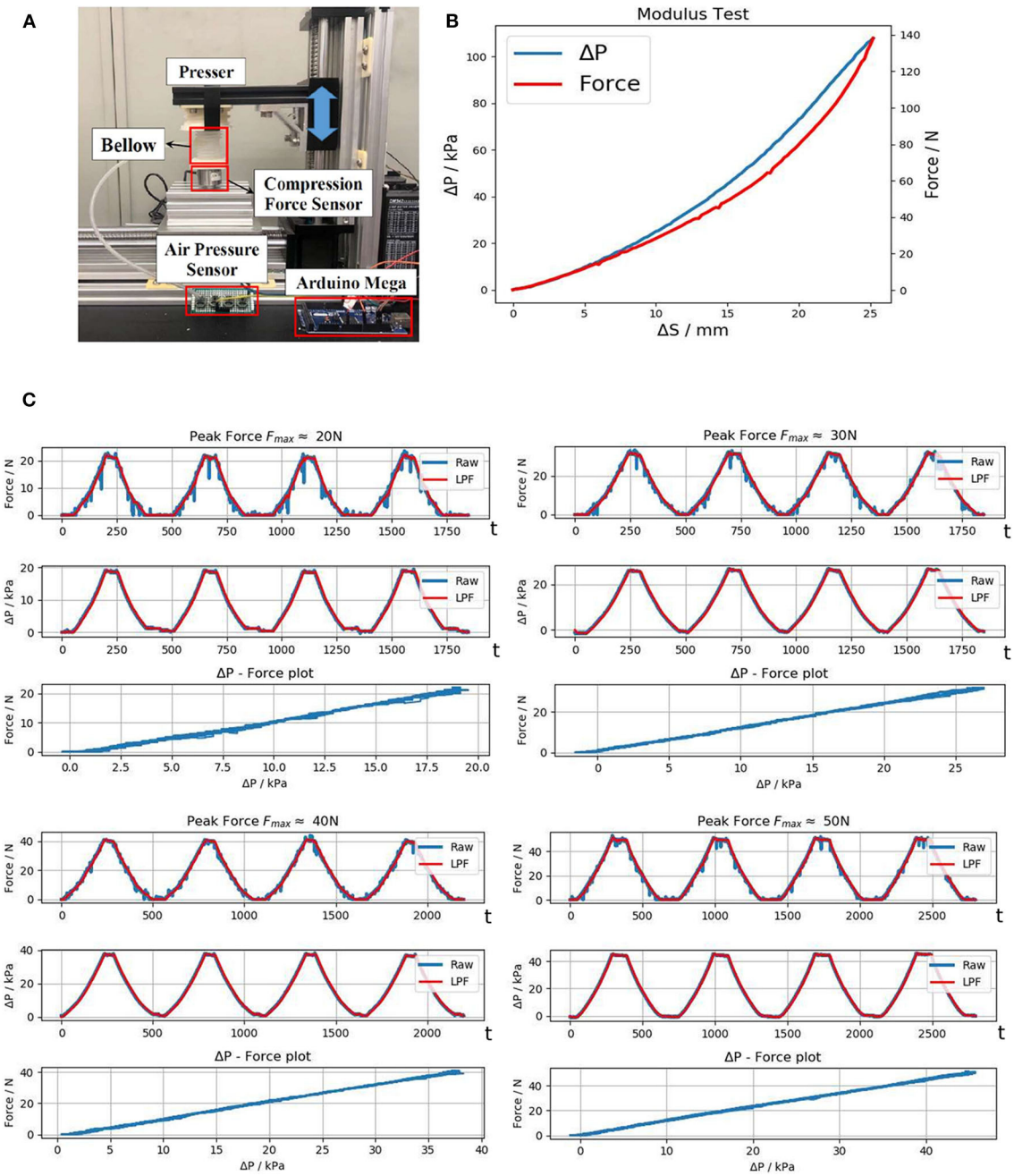


FIGURE 7 | Evaluation of the pressure sensing bellow; **(A)** Testing platform; **(B)** Modulus test; **(C)** Repeatability tests under different external force, from left to right: F_{max} ≈ 20N, 30N, 40N, 50N.

and the time of their pressures reaching the peaks are similar. Such a pattern can be used by the user to unlock the walker.

All these results show that the pressure data can be used to detect different states of the user, including falling and other user intention. By analyzing the pressure data, the walker can monitor the user's state to offer falling protection or respond to other user intention. In further development, more applications can be designed to make the walker more intelligent and safer comparing to the current version. For example, by comparing the changes in the pressure over a long period, the walker can detect whether the user is getting tired. Also, more advanced models such as NNs can be applied to learn from the pressure data and extract more information for medical observation.

3.3.2. Front Following

We experiment with user moving forward, turning and moving through narrow space without pushing the handle, as the **Supplementary Video 1** demonstrated. For

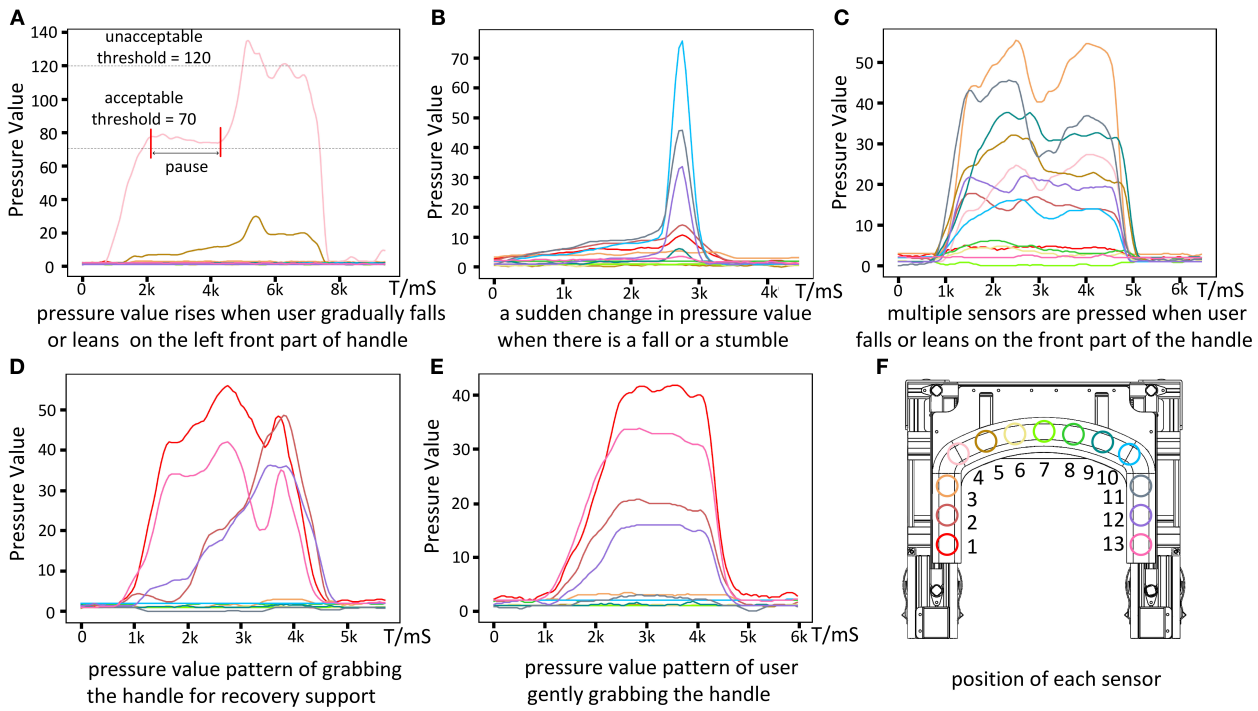


FIGURE 8 | Different patterns of pressure data corresponding to different user interactions and position of each sensor. **(A–C)** Are three possible pressure data patterns when a fall occurs. **(D,E)** Show two types of data patterns when the walker is detecting the user's unlock intention. **(F)** Shows the position of each sensor on the handle.

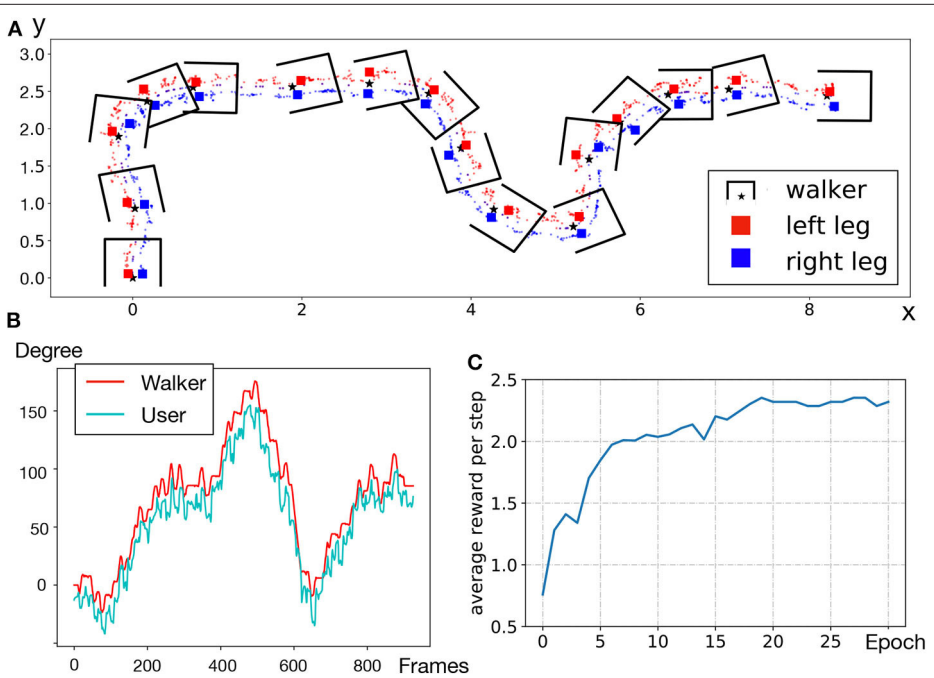


FIGURE 9 | (A) Leg position trajectory; **(B)** Orientation of the walker as compared to the user; **(C)** RL training curve of SSL in simulator.

evaluation purpose, we also record actual orientation of the user during front following by having the user wearing an IMU.

Figure 9A shows user leg positions collected. The x and y axes represent the spatial position, and the unit is meter. Scatter points represent leg locations; we draw a box to represent the walker for

every 60 leg points; the center of the walker and corresponding leg positions are highlighted. From the highlighted points, we observe that the walker area always covers both legs and there is always one leg on the baseline. This shows that our front following achieves coaxial following, which is a novel practical function for older persons who cannot consistently push the walker well, but rely on the walker's fall support functionality while walking.

Figure 9B compares the orientation of the user with that of the walker. The range of orientation θ is -180 to 180° and the starting position where the walker enables front following is 0°. We observe that the two lines follow a similar trend, and overlap at some peak points. The average error is 5.5° approximately. It demonstrates that the walker can change direction promptly according to the user's expected angle.

Most existing human-following studies consider robot following the human from the back; in case of robot front-following a human, existing systems assume that the robot is a distance away from the person, and the robot can easily amend its route on the go. In our scenario, the robotic walker and the user are within close proximity (user walking with feet along the rear axis of the walker). While this functionality enables elderly users to achieve hands-free walking, we also consider the situation that elderly users still need physical support with hands touching on soft handles. The intention of turning will then be detected and analyzed through haptic monitor, to cooperate with movement prediction of hands-free front-following to generate better tracking strategy.

The robustness of the proposed NN to predict lower leg gesture can be enhanced in the product usage phase, through collecting diverse and long-term training samples: walkers used by elderly users are allowed to compute and push their model gradients to the cloud periodically, and pull updated model parameters after aggregation is done in the backend cloud.

3.3.3. Autonomous Mobility Through SSL

We experimented in a real-world home-like setting with one hall (maximum length over 10 m causing strong reverberation) and four separate regions (similar as rooms). The signal-to-noise ratio (SNR) can reach as low as 7 dB. Environment layout is given in **Figure 10**.

We first build the same environment in the GSound simulator. A microphone array with four microphones, whose maximum distance is 0.75 m, records sound data generated from one sound

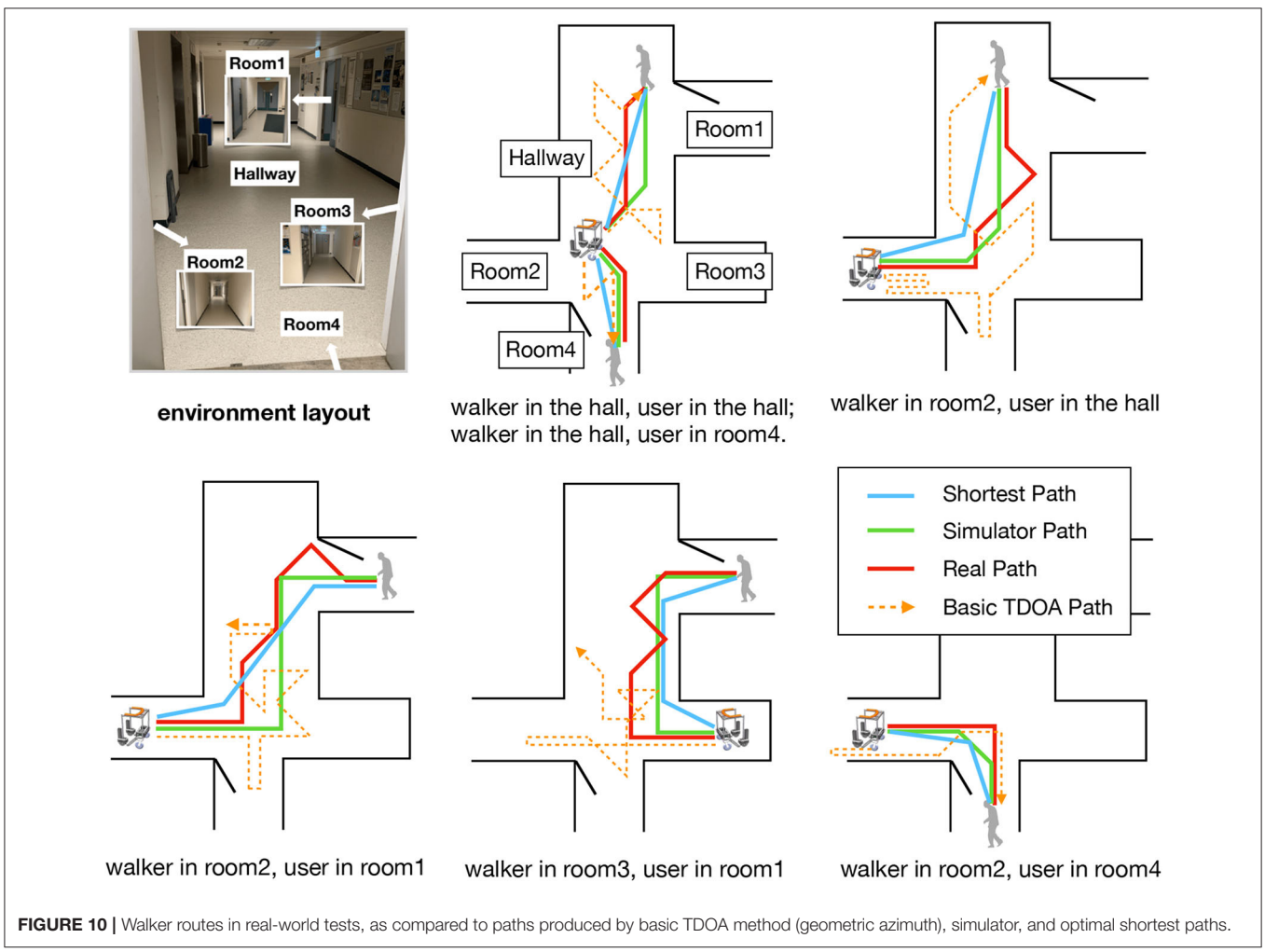


FIGURE 10 | Walker routes in real-world tests, as compared to paths produced by basic TDOA method (geometric azimuth), simulator, and optimal shortest paths.

source. The length of each step of the walker is 1 m. By changing the positions of the walker and the sound source, we simulate in total 4,000 raw data samples (90% used as training dataset, 10% as test dataset) for training of the RL model. **Figure 9C** shows the training convergence curve, where a higher average reward per step indicates that the walker produces a better route toward the sound source.

The offline trained model is then deployed to the real-world scenario. In **Figure 10**, we compare the routes performed in reality to the routes derived by the simulator, as well as the optimal shortest routes computed. The difference between simulator paths and the shortest paths is mainly due to our defining eight discrete directions for the walker to move on. Our RL-based approach outperforms the basic geometric azimuth method (which directly uses calculated time-delay of each microphone pair to generate the basic TDOA path): the latter often makes the walker lost in the hall (due to high reverberation and the lack of reward mechanism), especially when the user summons the walker in different rooms. The "reality gap" (Tan et al., 2018) between real paths and simulator paths stems from the difference between the simulator and real-world environments: especially, for sound propagation in a multiple-room home with high reverberation, physical parameters of the real world are hard to be simulated exactly. Even so, our smart walker with RL model is able to automatically approach the user located in another room only based on voice signals. Our RL model is robust as it only needs offline training using data from the simulator and slight online tuning to achieve autonomous mobility in new environments.

4. CONCLUSION

This paper proposes a novel smart robotic walker platform to assist the elders with mild mobility impairment. We design a unique and sturdy mechanical structure that cooperates with sensors, and apply soft-robotic technology on the walker's handle to achieve better protection and richer sensing capabilities. A series of stability tests show that the walker has good resistance against external disturbance. The soft handle prototype meets our expectation and can provide useful information about the user with a low-cost solution.

We design a comprehensive finite-state machine model to detect user intention and emergency events in a timely manner through analyzing spatiotemporal pressure data collected from soft handle. We also develop a hands-free close-proximity front-following function through intelligent control using an IR sensor, a lidar, and NN-based gait classifier. A reinforcement learning-based sound source localization approach is implemented for summoning the walker to the user through voice signals. Field tests show that our walker can actively approach the user in a complex indoor environment through an acceptable path. All these intelligent functionalities achieved enables our walker to provide an elderly user with sufficient mobility safety and rich modes of human-robot interaction.

As future work, we will further investigate learning-based algorithms to learn more of user behavior through the soft handle interface. To make our front following more generic, we seek to collect data on more walking styles. For SSL, we plan to further reduce the "reality gap" when applying the RL model in the real world.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

ETHICS STATEMENT

Ethical review and approval was not required for the study on human participants in accordance with the local legislation and institutional requirements. The patients/participants provided their written informed consent to participate in this study. Written informed consent was obtained from the individual(s) for the publication of any potentially identifiable images or data included in this article.

AUTHOR CONTRIBUTIONS

XZ and ML designed software algorithms, performed the experiments, and wrote the corresponding part of the manuscript. ZZ designed hardware structure, performed the evaluations, and wrote the corresponding part of the manuscript. CZ analyzed user intention from data collected and wrote the corresponding part of the manuscript. YZ help design soft robotic sensors. JP, ZW, and CW conceived, supervised the project, and revised the manuscript. All authors contributed to the article and approved the submitted version.

FUNDING

This work was supported by Hong Kong Innovation and Technology Commission's Innovation and Technology Fund (Award No. ITS/305/19FP).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fnbot.2020.575889/full#supplementary-material

REFERENCES

Ahn, S., Koo, B., Jeong, H., Nam, Y., Kim, J., Kim, T., et al. (2019). Evaluation of imu-based pre-impact fall detection algorithms using public dataset. *Assist. Technol.* 231–232. doi: 10.3390/s19040774

Bao, Y., Peng, Y., and Wu, C. (2019). "Deep learning-based job placement in distributed machine learning clusters," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications* (Paris: IEEE), 505–513.

Bertrand, K., Raymond, M.-H., Miller, W. C., Ginis, K. A. M., and Demers, L. (2017). Walking aids for enabling activity and participation:

a systematic review. *Am. J. Phys. Med. Rehabil.* 96, 894–903. doi: 10.1097/PHM.0000000000000836

Carlucho, I., De Paula, M., Wang, S., Petillot, Y., and Acosta, G. G. (2018). Adaptive low-level control of autonomous underwater vehicles using deep reinforcement learning. *Robot. Auton. Syst.* 107, 71–86. doi: 10.1016/j.robot.2018.05.016

Chen, X., Yi, J., Li, J., Zhou, J., and Wang, Z. (2018). Soft-actuator-based robotic joint for safe and forceful interaction with controllable impact response. *IEEE Robot. Autom. Lett.* 3, 3505–3512. doi: 10.1109/LRA.2018.2854409

Choi, J., Park, K., Kim, M., and Seok, S. (2019). "Deep reinforcement learning of navigation in a complex and crowded environment with a limited field of view," in *2019 International Conference on Robotics and Automation (ICRA)* (Montreal, QC: IEEE), 5993–6000.

Chowdhary, G., Gazzola, M., Krishnan, G., Soman, C., and Lovell, S. (2019). Soft robotics as an enabling technology for agroforestry practice and research. *Sustainability* 11:6751. doi: 10.3390/su11236751

Chu, Z., Sun, B., Zhu, D., Zhang, M., and Luo, C. (2020). Motion control of unmanned underwater vehicles via deep imitation reinforcement learning algorithm. *IET Intell. Trans. Syst.* 14, 764–774. doi: 10.1049/iet-its.2019.0273

Di, P., Hasegawa, Y., Nakagawa, S., Sekiyama, K., Fukuda, T., Huang, J., et al. (2015). Fall detection and prevention control using walking-aid cane robot. *IEEE/ASME Trans. Mechatron.* 21, 625–637. doi: 10.1109/TMECH.2015.2477996

Fan, T., Long, P., Liu, W., and Pan, J. (2020). Distributed multi-robot collision avoidance via deep reinforcement learning for navigation in complex scenarios. *Int. J. Robot. Res.* doi: 10.1177/0278364920916531. [Epub ahead of print].

Garçon, L., Khasnabis, C., Walker, L., Nakatani, Y., Lapitan, J., Borg, J., et al. (2016). Medical and assistive health technology: meeting the needs of aging populations. *Gerontologist* 56(Suppl. 2), S293–S302. doi: 10.1093/geront/gnw005

Gharieb, W. (2006). "Intelligent robotic walker design," in *International Conference on Automation, Robotics and Autonomous Systems*.

Gleeson, B., MacLean, K., Haddadi, A., Croft, E., and Alcazar, J. (2013). "Gestures for industry intuitive human-robot communication from human observation," in *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* (Tokyo: IEEE), 349–356.

Glover, J. (2003). A robotically-augmented walker for older adults.

Graf, B. (2009). An adaptive guidance system for robotic walking aids. *J. Comput. Inform. Technol.* 17, 109–120. doi: 10.2498/cit.1001159

Hans, M., Graf, B., and Schraft, R. (2002). "Robotic home assistant care-o-bot: Past-present-future," in *Proceedings of 11th IEEE International Workshop on Robot and Human Interactive Communication* (Berlin: IEEE), 380–385.

Hu, Z., Wan, K., Gao, X., Zhai, Y., and Wang, Q. (2020). Deep reinforcement learning approach with multiple experience pools for UAV's autonomous motion planning in complex unknown environments. *Sensors* 20:1890. doi: 10.3390/s20071890

ISO (2003). Waking aids manipulated by both arms-requirements and test methods–part 2: Rollators.

Knapp, C., and Carter, G. (1976). The generalized correlation method for estimation of time delay. *IEEE Trans. Acoust. Speech Signal Process.* 24, 320–327. doi: 10.1109/TASSP.1976.1162830

Krishna, K., and Murty, M. N. (1999). Genetic k-means algorithm. *IEEE Trans. Syst. Man Cybern. Part B* 29, 433–439. doi: 10.1109/3477.764879

Luz, C., Bush, T., and Shen, X. (2017). Do canes or walkers make any difference? Nonuse and fall injuries. *Gerontologist* 57, 211–218. doi: 10.1093/geront/gnv096

Ma, N., Gonzalez, J. A., and Brown, G. J. (2018). Robust binaural localization of a target sound source by combining spectral source models and deep neural networks. *IEEE/ACM Trans. Audio Speech Lang. Process.* 26, 2122–2131. doi: 10.1109/TASLP.2018.2855960

Martin, R. (1994). Spectral subtraction based on minimum statistics. *Power* 6.

Moustris, G. P., and Tzafestas, C. S. (2016). "Intention-based front-following control for an intelligent robotic rollator in indoor environments," in *2016 IEEE Symposium Series on Computational Intelligence (SSCI)* (Athens: IEEE), 1–7.

Mukai, T., Hirano, S., Nakashima, H., Kato, Y., Sakaida, Y., Guo, S., et al. (2010). "Development of a nursing-care assistant robot riba that can lift a human in its arms," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Taipei: IEEE), 5996–6001.

Schissler, C., and Manocha, D. (2011). "Gsound: Interactive sound propagation for games," in *Audio Engineering Society Conference: 41st International Conference: Audio for Games* (Chapel Hill: Audio Engineering Society).

Sutton, R. S., McAllester, D. A., Singh, S. P., and Mansour, Y. (2000). "Policy gradient methods for reinforcement learning with function approximation," in *Advances in Neural Information Processing Systems*, 1057–1063.

Tai, L., Paolo, G., and Liu, M. (2017). "Virtual-to-real deep reinforcement learning: continuous control of mobile robots for mapless navigation," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Vancouver: IEEE), 31–36.

Tan, J., Zhang, T., Coumans, E., Iscen, A., Bai, Y., Hafner, D., et al. (2018). Sim-to-real: Learning agile locomotion for quadruped robots. *arXiv[Preprint].arXiv:1804.10332*. doi: 10.15607/RSS.2018.XIV.010

Tucker, M., Novoseller, E., Kann, C., Sui, Y., Yue, Y., Burdick, J., et al. (2019). Preference-based learning for exoskeleton gait optimization. *arXiv[Preprint].arXiv:1909.12316*. doi: 10.1109/ICRA40945.2020.9196661

Valin, J.-M., Michaud, F., Rouat, J., and Létourneau, D. (2003). "Robust sound source localization using a microphone array on a mobile robot," in *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)(Cat. No. 03CH37453), Vol. 2* (Las Vegas, NV: IEEE), 1228–1233.

Wan, K., Gao, X., Hu, Z., and Wu, G. (2020). Robust motion control for UAV in dynamic uncertain environments using deep reinforcement learning. *Remote Sens.* 12:640. doi: 10.3390/rs12040640

WHO (2018). *World Health Statistics 2018: Monitoring Health for the SDGs, Sustainable Development Goals*.

Wikipedia contributors (2020). *Azimuth — Wikipedia, the Free Encyclopedia*. (accessed February 8, 2020).

Xiao, X., Zhao, S., Zhong, X., Jones, D. L., Chng, E. S., and Li, H. (2015). "A learning-based approach to direction of arrival estimation in noisy and reverberant environments," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (Brisbane, QLD: IEEE), 2814–2818.

Xu, W., Huang, J., and Cheng, L. (2018). A novel coordinated motion fusion-based walking-aid robot system. *Sensors* 18:2761. doi: 10.3390/s18092761

Yi, J., Chen, X., Song, C., Zhou, J., Liu, Y., Liu, S., et al. (2018). Customizable three-dimensional-printed origami soft robotic joint with effective behavior shaping for safe interactions. *IEEE Trans. Robot.* 35, 114–123. doi: 10.1109/TRO.2018.2871440

Zhang, F., Leitner, J., Milford, M., Upcroft, B., and Corke, P. (2015). Towards vision-based deep reinforcement learning for robotic motion control. *arXiv[Preprint].arXiv:1511.03791*.

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

frontiers in Neurorobotics

ORIGINAL RESEARCH
published: 11 December 2020
doi: 10.3389/fnbot.2020.572856

Self-Sensing Pneumatic Compressing Actuator

Nan Lin[1†], *Hui Zheng*[2†], *Yuxuan Li*[2], *Ruolin Wang*[3], *Xiaoping Chen*[2] *and Xinming Zhang*[2*]

[1] School of Data Science, University of Science and Technology of China, Hefei, China, [2] School of Computer Science and Technology, University of Science and Technology of China, Hefei, China, [3] School of Information Science and Technology, University of Science and Technology of China, Hefei, China

Using soft pneumatic actuator is a feasible solution in the complex unstructured environment, owing to their inherent compliance, light weight, and safety. However, due to the limitations of soft actuators' materials and structures, they fall short of motion accuracy and load capacity, or need large-size, bulky compressors. Meanwhile, in order to gain better control, it is essential for them to sense the environments as well. This leads to high-price sensors or a complicated manufacture technique. Here, a self-sensing vacuum soft actuation structure is proposed, aiming at acquiring good balance among precision, output force, and actuation pressure. The actuator mainly comprises a flexible membrane and a compression spring. When actuated, the flexible membrane outside the actuator compresses the internal spring skeleton, realizing large contractile motion in axial direction. Its built-in force sensor can indirectly measure the absolute displacement of the actuator with certain accuracy (about 5% F.S.). Besides, it does not require high actuation pressure to generate enough output force. The actuator is quite easy to manufacture with low cost, and there are a variety of materials to choose from. We established quasi-static models for actuators built of two different kinds of membrane materials, and tested their accuracy and output force. In addition, to break through the limits of vacuum actuation, a method of positive-negative pressure combined actuation has been proposed, which lowers the requirements for air source equipments, increases actuation pressure, and reduces potential safety threats at the same time. This kind of soft actuators can also effectively resist and detect impacts. The design of a two-finger dexterous robot hand and robot joint based on this soft actuator illustrates its broad application prospects in the fields of mobile robots, wearable devices, and human–robot interaction.

Keywords: soft robotics, design, safety, pneumatic actuator, self-sensing

OPEN ACCESS

Edited by:
Zheng Wang,
Southern University of Science and Technology, China

Reviewed by:
Feifei Chen,
Shanghai Jiao Tong University, China
Surya Girinatha Nurzaman,
Monash University Malaysia, Malaysia

***Correspondence:**
Xinming Zhang
xinming@ustc.edu.cn

†These authors have contributed equally to this work and share first authorship

Received: *15 June 2020*
Accepted: *31 August 2020*
Published: *11 December 2020*

Citation:
Lin N, Zheng H, Li Y, Wang R, Chen X and Zhang X (2020) Self-Sensing Pneumatic Compressing Actuator. Front. Neurorobot. 14:572856. doi: 10.3389/fnbot.2020.572856

1. INTRODUCTION

Soft actuators, relative to rigid mechanical structures, have been widely used in rescue, medical care, wearable devices (Ilievski et al., 2011; Kim et al., 2013; Cianchetti et al., 2014; Park et al., 2014; Rus and Tolley, 2015), etc, owing to their inherent compliance and safety. Traditional actuators like electric motors can reach high precision and speed, which makes it excel at repetitive tasks in industry. But they are often bulky and stiff, and a structured environment is needed for operation, otherwise they may do damage to the environment or break themselves. Soft actuators, by contrast, can perfectly adapt to the complex or dynamic situation, which reduces the threat to users.

Recently, research on soft actuator are promoted with the rapid development of flexible materials, structures, and sensors. According to the actuation method, there are mainly several kinds, that is, electromagnetic, thermal, chemical, fluid actuation, and hydraulic actuation.

Shape memory alloys (SMAs) (Jani et al., 2014), actuated by electric heating, have large contraction force output and the strain is also significant, yet their high non-linearity and hysteries are barriers to application. Similar to SMAs, shape memory polymers (SMPs) (Ahn et al., 2008; Hu et al., 2012) used more kinds of stimuli like chemical or light, while the response time is limited accordingly. Dielectric elastomer actuators (DEAs) (O'Halloran et al., 2008; Anderson et al., 2012), powered by high electric field, can meet the demands of high-frequency actuation. DEAs use electrostatic force to attract two different potential electrodes on either side of a compressible membrane, thus get large strain. Furthermore, Keplinger et al. proposed a novel hydraulical-electrostatic hybrid actuator, Peano-HASEL (Acome et al., 2018; Kellaris et al., 2018), based on Peano fluidic muscle (Sanan et al., 2014). It provided direct coupling of electrostatic and hydraulic forces for high-power and precise operation. The main remaining hurdle in using electrostatic actuation is the need of driving voltages up to the order of kilovolts, which is difficult to achieve and might be a potential safety issue, limits the usefulness of this technique.

Soft pneumatic actuators (SPAs) are the most popular actuators in soft robotics. SPAs make use of compressed air (or vacuum) as power source, so it will not cause any pollution to environment. Pneumatical driving offers other advantages such as lightweight, compliance, and inherent safety. The well-known McKibben artificial muscle, invented and developed in the 1950s (Gavrilović and Marić, 1969; Chou and Hannaford, 1996), is a landmark pneumatic actuator. The McKibben actuator comprises a rubber inner tube covered with a shell of braided, inextensible fibers. When the inner tube is inflated by positive pressure, the muscle swells radially and contracts axially to shorten its overall length. By using these flexible materials, the McKibben actuator considerably is more compliant and lightweight than common pneumatic cylinder. However, the contraction ratio is not very satisfactory ($\leq$40%), and usually high pressure is required for operation. Several improvements have been preformed, such as choosing superior shell materials, structures, or implementing different membrane composition (Daerden and Lefeber, 2001; Villegas et al., 2012; Belding et al., 2018; Terryn et al., 2018), but high driving pressure is still essential. Contrary to McKibben muscle, fiber-reinforced actuators lengthen when pressurized (Galloway et al., 2013; Connolly et al., 2015, 2017). The actuators consist of a core bladder reinforced with inextensible fibers, which wrap around to limit the radial expansion. They are able to realize a wide range of motions (bending, twisting, and extension), and have larger strain (300% in Hawkes et al., 2016). However, the fabrication process for fiber-reinforced actuators is complicated.

Another famous SPA is pneumatic network (PneuNet) originally developed by Harvard University (Sun et al., 2013; Mosadegh et al., 2014). It is made almost entirely out of soft materials such as the silicone rubber, with a series of channels and chambers inside an elastomer. When pressurized, its channels are inflated and create assigned motion like bending or twisting. The PneuNets actuators are entirely soft, and can be inflated with low pressure, which guarantees the safety for human interaction and environmental adaptability. However, the inherent compliance also severely limits the stiffness and output forces, and makes it unstable.

In recent years, a novel design pattern combining origami with other actuation method has been attracting wide attention, and many outstanding achievements have emerged (Onal et al., 2013; Mu et al., 2015; Paez et al., 2016; Miyashita et al., 2017; Kim et al., 2018). Martinez et al. presented composite structures comprising elastomers and paper (or other flexible sheets) (Martinez et al., 2012). They fabricated the 3D paper structure using origami or laser cutting, and embedded it into silicone elastomers. Due to the self-folding character, this structure was able to reach high stretching ratio, and the sheet inside could reinforce the elastomeric matrix to withstand external disturbance. Yi et al. proposed a fiber-reinforced origamic robotic actuator (FORA) to improve the performance of McKibben-type artificial muscles by replacing the rubber inner tube with specially designed origamic chamber (Yi et al., 2018). Li et al. presented an architecture for fluidic artificial muscles, which could be programmed to produce complex multiaxial motion (Li et al., 2017). The fluid-driven origami-inspired artificial muscles (FOAMs) consisted of a compressible skeleton and a non-strechable membrane. When driven by negative pressure, the membrane deformed inwards to push the skeletal structure contract. This innovative structure made FOAMs extremely lightweight, low cost, and provided large contracting ratio.

In order for precise control, it is necessary to pair the actuator with sensors to create a feedback control system. But the deformable characteristic of soft robots prevents the use of many conventional sensors, and the accurate sensor models are often unavailable for calculation and analysis (Polygerinos et al., 2017). In soft robotics, alternative sensing methods with low-modulus sensors are preferred. Ionic liquid–based resistive sensors (Chossat et al., 2013; Yeo et al., 2016) are perceived by monitoring the resistance variation of the driven fluid. The elastomer layers of these sensors are often patterned with microfluidic channels, which are filled with liquid conductors (Majidi et al., 2011; Wong et al., 2012). liquid–based resistive sensors can be tuned by modulation of channel geometries, and are able to measure various types of strains (Vogt et al., 2013), thus have acquired wide attention in applications like soft robotic hand (Wall et al., 2017), wearable devices (Kramer et al., 2011b), and human fingers (Kramer et al., 2011a). But they suffer the large temperature drift due to the correlation between temperature and ion concentration. Besides, poor long-term stability and risk of leakage are also tricky. In other related works, conductive thermoplastic material is adopted to avoid these problems (Culha et al., 2014).

Capacitive sensors measure the capacitance variations caused by geometry changes when the elastic body is deformed. In these systems, dielectric layer is sandwiched between conductive soft plates, and conductive elements are employed to create conformable electrodes in the sensing system. Some electrostatic actuators like DEAs, EAPs, or Peano-HASEL naturally sense the deformation through capacitance monitoring (Jung et al., 2008; Kruusamäe et al., 2015; Acome et al., 2018) through their actuation mechanism. For other soft robots, nanowires (Lipomi et al., 2011), nanotubes (Amjadi et al., 2014), carbon black (Tsouti

et al., 2016), and conductive fabrics (Atalay et al., 2017) are also used as dielectric layers. capacitive-based sensors offer some advantages over other systems, such as high linearity and fast response time, which are important parameters when the sensors are intended to be used in real-life scenarios (Lipomi et al., 2011; Hu et al., 2013). But they are sensitive to environmental contaminants, like proximity effect to conductive objects, and are mostly prone to cracking and delamination over extended usage of the sensor.

More recently, optical-based sensing has emerged as another soft sensor category where motion is detected through changes in the light that is emitted and received in a light guide (Zhao et al., 2016a; Harnett et al., 2017; Teeple et al., 2018). The probed optical signal properties can be intensity (Polygerinos et al., 2011), phase (Pang et al., 2007), frequency (Zook et al., 2000), or polarization (Saad et al., 1995). Optical-based sensors are insensitive to any environmental interference, thus already been used for tactile sensing in prosthetic fingers (Du et al., 2017), soft surgical manipulator (Sareh et al., 2014), and other clinical practices (Liu et al., 2011). Fiber optic intensity modulation is a common method that refers to a class of sensing techniques, and has been applied in a soft bending actuator (Zhao et al., 2016a) to detect motion and infer the actuator shape. Furthermore, the integration of stretchable optical waveguides makes the sensor seamlessly deform with the actuator (Zhao et al., 2016b), but these methods are limited by the assumption that the sensor curvature is uniform. Recently, other researchers have also explored optoelectronic shape detection with fiber Bragg gratings (FBGs), which reflect light with a peak wavelength that shifts in proportion to variations in strain and temperature (Hill and Meltz, 1997). Multiple FBGs can be fabricated on different longitudinal positions of one fiber to monitor the distributed strain and pressure (Zhuang et al., 2018). FBGs show great potential to develop completely soft strain sensors for soft continuum robots (Wang et al., 2016b), but expensive and complex fabrication process is the main barrier. Instead, TacTip sensors (Cramphorn et al., 2016; Wardcherrier et al., 2017, 2018) directly use embedded camera to monitor the deformation of a soft structure's skin, but rigid camera system is needed to integrate in soft robots.

Besides the aforementioned works, other soft sensors such as inductive (Rahimi et al., 2014; Felt et al., 2016), magnetic (Ozel et al., 2015; Wang et al., 2016a; Luoming et al., 2017), and piezoresistive sensors (Yamamoto et al., 2007; Shapiro et al., 2014) are also able to be put in practice. More details about soft sensing can be found in literatures (Wang et al., 2018). But until now, the production process of most soft sensors is intricate, especially hard to be integrated with the manufacture of actuators, or they may affect the movement of actuators.

In this paper, we present a novel linear actuator Self-sensing Pneumatic Compressing Artificial Muscle (SPCAM) based on the work of Li et al. (2017), but there are several key improvements. Driven by vacuum pressure, the actuator can realize the axial contraction motion, similar to muscle tissue. Besides, the actuator can be stretched passively, and the elastic structure ensures its safety and its impact resisting property. A simple but effective sensor, which has certain accuracy, is integrated into SPCAM without disturbing its motion, so SPCAM has the ability of self-sensing. The actuator is lightweight with large strain and output force. It is easy to manufacture and be put into mass production. Furthermore, a positive–negative pressure combined mechanism is proposed. That is, the SPCAM actuator can be embedded in other positive pressure actuation structures to lower the requirements of actuation pressure and enhance its accuracy, which makes it more practical in mobile devices.

The article is organized as follows: the design and actuating principles are presented in section 2. The membrane materials, SPCAM's static model among traction force, actuating pressure, and structure parameters, are analyzed in section 3. Its performance like accuracy and actuation force is tested in section 4. We combine the SPCAM with air cylinder and McKibben artificial muscle, which is presented in section 5. Finally, in sections 6 and 7, a two-finer dexterous gripper and a revolute joint for exoskeleton are designed to demonstrate SPCAM's great potential for applications; meanwhile, conclusions and future work are summarized.

2. ACTUATOR CONCEPT AND DESIGN

2.1. Schematics and Operation Principle of SPCAM

Figure 1 shows the schematic of SPCAM. The actuator consists of several major parts as follows: a spring skeleton, a flexible membrane outside, 3D-printed connectors, sealing rings, and a pull-pressure microsensor. The spring is sealed in the membrane, forming a compressible internal cavity. When the actuation pressure is zero, the self-locking performance is guaranteed by the elasticity of latex membrane and spring to some extent. Once we pump air out of the cavity, the pressure difference between internal and external part makes the membrane deform inward, similar to the work described in Li et al. (2017). The deformation of the membrane will exert pressure on the spring and makes it compress axially, generating a compressing force. The pulling/pressing sensor fixed on the end of the actuator is connected to the end of the spring, being able to sense the compressing force F_{sensor} (**Figure 1B**). Since both the spring and the pulling/pressing microsensor are inside of the cylindrical hyperelastic membrane, the outer pressure force and tension can be considered as external forces. According to Hooke's law, the compressing force scales linearly with respect to the displacement of the spring, so the displacement of the spring, that is, the displacement of the actuator, can be approximately calculated as the compressing force F_{sensor} divided by Hooke's coefficient.

2.2. Materials and Fabrication

The low cost of SPCAM is ensured by the standardized parts and simple assembly technology without any complicated procedure or advanced equipment (displayed in **Supplementary Video**). The core component is a flexible membrane and there are a variety of membrane materials to choose from. Here, we choose stretchable latex membranes and non-stretchable low-density polyethylene (LDPE) films for test. Other components, such as springs (304 stainless steel) and the pull-pressure microsensors (freud, DHMH-106, 1% F.S, 3 kg max.), are widely used in robots

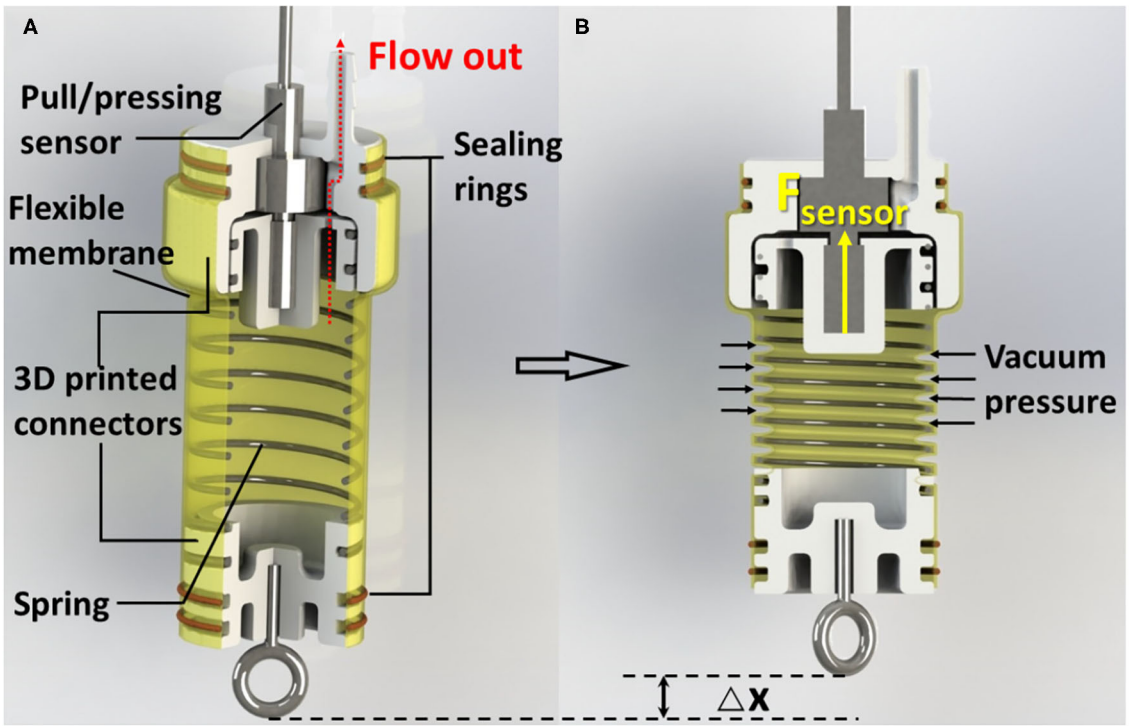


FIGURE 1 | (A) Cutaway view of the Self-sensing Pneumatic Compressing Artificial Muscle (SPCAM). A typical SPCAM mainly consists of a compressing spring, 3D-printed parts, a flexible membrane, seals, and a pull/pressing sensor. **(B)** SPCAM under working state. The vacuum pressure pushes the membrane deform inward to make the actuator contract.

and automation equipment. There is an axial torque generated during contraction process, so we attach a microrotating ring to compensate the twist. The only non-standard part is 3D-printed connectors on the two ends, which can be replaced by injection molding parts if put into quantity production. The weight of the whole actuator is <40 g, 60% of which is occupied by the sensors, and it can be further reduced if the Micro Electromechanical System (MEMS) technology is adopted.

We have compared SPCAM with other four kinds of flexible actuators, taking their maximum strain and output force into consideration. Peano-HASEL (Kellaris et al., 2018) uses both electrostatic and hydraulic principles to linearly contract on application of voltage in a muscle-like fashion. PneuNet (Sun et al., 2013) is made with soft material and inner chambers. When pressurized, the inflated chambers will create assigned motion. FORA (Yi et al., 2018) is a fiber-reinforced origamic actuator, which improves the performance of McKibben-type artificial muscles. PPAM (Terryn et al., 2018) is another pneumatic artificial muscle whose membrane is constructed out of self-healing polymer.

Table 1 shows the comparison of five actuators. Here, the data of Peano-HASEL is acquired under the condition of 10 kV voltage. For other four actuators, the actuation pressure is limited with 40 kPa. The data show that SPCAM outperforms others in maximum output force and maximum strain. In the same time, our SPCAM design does not require customized parts and advanced fabrication techniques; while Peano-HASEL requires specialized high-voltage electrodes, others require complicated production procedure like pouring forming, which leads to higher cost.

TABLE 1 | Comparison of five actuators.

Actuator	SPCAM	Peano-HASEL	PneuNet	FORA	PPAM
Maximum strain(%)	71	18	50	50	12
Maximum force(N)	40	10	2.2	50	18
Production complexity	Easy	Hard	Middle	Hard	Easy
Production cost	Low	Middle	Middle	High	Middle

3. MODELING AND ANALYSIS

3.1. Outer Membrane Material

We have chosen two different kinds of membrane materials to build the actuator. The first one is polyethylene membrane, as used in Li et al. (2017). It is a kind of non-stretchable material, which can be made into certain shapes through hot-pressing technique. One of the main problems of polyethylene membranes is there will be indentations during manufacture process. This will cause axial non-uniformity of the material and affects actuator's motion. The other one is cylindrical latex membrane, being widely used in soil sample analysis. We prepare different

specifications of membranes (different diameters, thickness, etc.) for actuator in order to fulfill design and test requirements.

Latex is a kind of hyperelastic, incompressible material. In small-strain areas, classic elasticity theory can well explain the strain–stress relation within the elastic material. However, latex membranes will have large strain under external force. To solve the large deformation (finite length deformation) issues of this kind of polymer, the strain energy function should be introduced. There are three commonly used forms of function as follows: the Ogden function, the Mooney–Rivlin function, and the neo-Hookean function (Mooney, 1940; Gent, 1996; Ogden, 1997). Here, we choose the Ogden strain energy theory. The three-form of the Ogden function is adopted as:

$$W(\lambda_1, \lambda_2) = \mu \sum_{i=1}^{3} \frac{\beta_i}{\alpha_i} (\lambda_1^{\alpha_i} + \lambda_2^{\alpha_i} + \lambda_1^{-\alpha_i} \lambda_2^{-\alpha_i} - 3) \quad (1)$$

where μ, α_i, and β_i are the inherent material constants. λ_i $(i = 1, 2, 3)$ are strains in three orthotropic directions. We divide the latex cylindrical membrane model into three directions: the meridional direction, circumferential direction, and vertical direction, as shown in **Figure 3A**.

3.2. Analysis of Static Equilibrium

In this section, the static model will be established to analyze the relationship between the characteristic parameters of SPCAM, that is, the initial screw pitch L_0, the screw pitch L_1 after deformation, actuator's diameter D, active coil number of spring N_a, thickness of the membrane H, pressure difference ΔP, and the output force F_{output}. Latex is a kind of non-linear, material, so we adopt a finite element analysis model based on the Ogden strain energy theory. Since polyethylene film is non-stretchable, its approximate solution can be obtained by using simple geometrical methods combined with classical mechanics. The analysis is based on the following assumptions:

1. The membrane has axial symmetry and density uniformity.
2. Thickness of the membrane is much thinner than the size of the actuator.
3. The influence of lateral force is ignored.

When SPCAM is under the effect of pressure difference, the spring will be compressed by the flexible membrane. However, the spring has a helix angle, so the compressing force does not apply entirely in the axial direction. Considering the screw pitch L_1 is small compared to the spring's diameter D, we assume that the force interference caused by helix angle has little influence on the Hooke's coefficient of spring, yet it still affects the force equilibrium of actuator, manifested as torque τ.

The equivalence process of model analysis is displayed in **Figure 2**. On the spring helix, the effect of pressure on a membrane material particle generates a force T_1, which has an included angle ϕ with the axis, where $\tan\phi = L_1/D$. Therefore, a single helix in the actuator after deformation can be equivalent to a slant cylinder model shown in **Figure 2B**, where its height $L_2 = L_1/\cos\phi$, and diameter $D_2 = D/\cos\phi$. Furthermore, considering the spring's compressive force, the model is eventually equivalent to a spring-cylinder model fixed at one end, as shown in **Figure 2C**. We can obtain the axial force T_1 by analyzing this model. See model **a** in **Figure 2**. T_1 can be divided into a SPCAM axial force $T_1' = T_1\cos\phi$, and a radial force $T_2' = T_1\sin\phi$. T_2' of all material particles has the same torsion direction, thus it creates a torque τ on the spring, making the spring twist for a certain angle. The final output force of the actuator, F_{output}, can be expressed as follows:

$$F_{output} = F_p + T_1'\pi D - K\Delta x \quad (2)$$

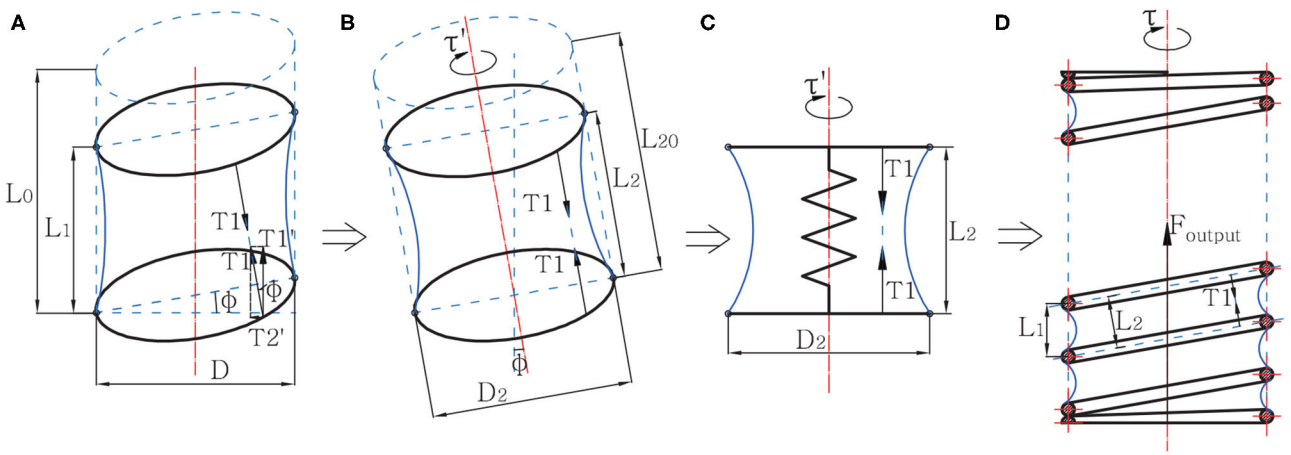


FIGURE 2 | The equivalence process of the Self-sensing Pneumatic Compressing Artificial Muscle (SPCAM) simplified mechanical model. The dashed lines stand for a single helix part of SPCAM before deformation, while the solid lines stand for its state after deformation. **(A)** The original compression model of a single helix section under negative pressure. **(B)** The equivalent slant cylinder model of the single section. **(C)** The spring-cylinder model. Consider the helix as two parallel annuluses connected by an ideal spring, with the same Hooke's coefficient as the SPCAM's spring. **(D)** The mechanical model of the whole SPCAM, showing the output force F_{output} and torque τ.

where K and Δx are stiffness coefficient and displacement (stretching or contracting) of the spring, respectively. F_p is the pushing force applied on the end of SPCAM by vacuum pressure,

$$F_p = \frac{\pi D^2}{4}\Delta P \tag{3}$$

The interference torque τ is calculated as:

$$\tau = T_2' \cdot \pi D \cdot \frac{D}{2} \cdot N_a \tag{4}$$

Note that there might be relative sliding between the spring and the membrane, caused by τ. For different kinds of materials, the friction may not be the same, which is beyond the scope of this article.

Contraction rate is an important parameter of linear actuators. The contraction rate of SPCAM is related to the initial screw pitch L_0 and membrane thickness H. If the change of membrane thickness is ignored during deformation process, the largest contraction strain of SPCAM can be approximately written as:

$$\delta_{contraction} = \frac{N_a(L_0 - 2H)}{L_{SPCAM}} \tag{5}$$

where L_{SPCAM} is the total length of SPCAM.

Polyethylene film is non-stretchable, so only latex-membrane SPCAMs have the issue of stretching rate. The stretching strain is mainly limited by the fixing and clamping force on both ends of the latex membrane, and the breaking strength of latex material itself. The maximum stretching length $\Delta X_{stretching}$ satisfies the following constraints:

$$\begin{cases} T_1'(\Delta X_{stretching}, \Delta P)\pi D < F_{fixed} \\ \frac{T_1'(\Delta X_{stretching}, \Delta P)}{H} < \sigma_{latex} \end{cases} \tag{6}$$

3.2.1. Latex SPCAM Model

From the equivalent model mentioned above, we can conclude that we only have to analyze the spring-cylinder model in **Figure 2C** to get the solution of T_1, and eventually obtain the output force F_{output} and interference torque τ expressed in Equations (2–4). The static equilibrium of cylindrical hyperelastic materials can be seen in Guo (2001) and Soleimani and Funnell (2016). Different from those literature, our actuator only has one fixed end, and the other end will have axial compressing displacement, which makes the modeling process pretty troublesome. To simplify the model, we assume that the actuator will first come to an intellectual state. In this process, the actuator is not under any external force (or is under an infinitesimal force), but has the meridional uniform deformation, reaching a contracting (or stretching) state. That is, $\lambda_1(X) \equiv L_{20}/L_2 = \lambda_0$, where L_{20} is the height of slant cylinder model before deformation, and L_2 is the distance between the two annuluses after the deformation. Then from this intellectual state, we fix the other end and analyze the compression deformation. The cylindrical membrane has the initial radius of midsurface R_m, mounting length L_2, and thickness H_m. The undeformed membrane is referred to a cylindrical polar coordinate system (X, ϕ, R), while the deformed membrane is referred to a different cylindrical polar coordinate system (x, ϕ, r). The material particle moves from its position in the undeformed profile C(X, ϕ, R) to a new position in the deformed profile c(x, ϕ, r). For each particle, we have defined the principal stretches in the meridional λ_1, the circumferential directions λ_2, and the direction λ_3 normal to the deformed membrane surface as:

$$\lambda_1 = \frac{\mathrm{d}s}{\mathrm{d}S}, \lambda_2 = \frac{r}{R_m}, \lambda_3 = \frac{h}{H_m} = \frac{1}{\lambda_1\lambda_2} \tag{7}$$

where s is the arc length measured from the pole ($x = 0$) to the particle c(x, ϕ, r) along the meridian of the deformed profile; S is the length measured from the pole ($X = 0$) to the particle C(x, ϕ, R) in the undeformed profile, where $S \equiv X$. h is the thickness of the membrane in the deformed situation, and λ_3 is determined by assuming that the membrane is incompressible. From the Ogden function, intellectual state transformation and the analysis in Guo (2001), λ_1, λ_2, and θ can be expressed as:

$$\frac{\mathrm{d}\lambda_1}{\mathrm{d}x} = \lambda_0 \cdot \frac{\sin\theta}{R}\left(\frac{\partial^2 W}{\partial\lambda_1^2}\right)^{-1}\left(\frac{\partial^2 W}{\partial\lambda_1\partial\lambda_2} \cdot \lambda_1 - \frac{\partial W}{\partial\lambda_2}\right) \tag{8}$$

$$\frac{\mathrm{d}\lambda_2}{\mathrm{d}x} = \lambda_1 R^{-1}\sin\theta \tag{9}$$

$$\frac{\mathrm{d}\theta}{\mathrm{d}x} = \left(\frac{\partial W}{\partial\lambda_1}\right)^{-1}\left(\frac{\lambda_1\lambda_2}{H} \cdot \Delta P - \frac{\cos\theta}{R} \cdot \frac{\partial W}{\partial\lambda_2}\right) \tag{10}$$

The geometric constraints at the boundaries are specified as follows:

$$x\,|_{X=0} = 0, \theta\,|_{X=0} = 0, x\,|_{X=L_2/2} = \frac{L_2}{2}, \lambda_2\,|_{X=L_2/2} = 1 \tag{11}$$

To get the numerical solution at the boundaries, we use the fourth-order Runge–Kutta method to run iterations on λ_1, λ_2, and θ. The visual results are also displayed in **Supplementary Video** using ABAQUS.

In this analysis, we mainly focus on the traction transmitted in the meridional direction of the material particle, noted as F_1. And F_1 can be expressed as:

$$F_1 = h \cdot \lambda_1 \cdot \frac{\partial W}{\partial\lambda_1}\,|_{X=L_2/2} \tag{12}$$

After getting the exact λ_1, λ_2,and θ at the boundaries from the iteration, we can plug them into Equation (12) to get F_1. However, this F_1 has a included angle θ with the vertical direction axis of slant cylinder model. We need to transform F_1 into T_1, which can be calculated as:

$$T1 = F1 \cdot \cos(\theta)\,|_{X=L_2/2} \tag{13}$$

According to Equations (12)–(13), we can eventually get the output force and interference torque of the actuator.

3.2.2. Polyethylene Film SPCAM Model

Same as latex membrane, a quasi-static cylindrical polyethylene membrane model is established to get T_1. Here, the principle of virtual work and geometric approximation conditions are used to obtain the approximate solutions. Output forces have direct relations with vacuum pressure ΔP and model length $X = L_2$, which together determine the curve of membrane. According to the principle of virtual work, resultant force F can be written as:

$$F(X) = T_1(X) \cdot \pi \cdot D_2 \tag{14}$$

$$F(X) \cdot \delta X = -\Delta P \cdot \delta V(X) \tag{15}$$

where V represents the internal air volume. We built the coordinate system XOY, where O is on the center profile of the membrane (i.e., the point with maximum cavity), whose height is marked as h_O. Two endpoints are marked as P_1,P_2, as shown in **Figure 3B**. The function of the parabola can be expressed as:

$$y = \frac{4h_O}{X^2}x^2 \tag{16}$$

Here, we approximately consider that the arc length of the parabola $P_1\overset{\frown}{O}P_2$ equals to twice the length of segment $\overline{P_1O}$. While the polyethylene material is non-stretchable, the length of membrane is constant, yielding:

$$\sqrt{X^2 + h_O^2} = \frac{L_{20}}{2} \tag{17}$$

The total volume V can be written as:

$$V(X) = 2 \cdot \int_0^{\frac{X}{2}} \pi \left(\frac{4h_O}{X^2} l^2 + \frac{D_2}{2} - h_O \right)^2 dl \tag{18}$$

Combining the Equations (14, 15, 17, and 18), one achieves:

$$T_1(X) = \frac{-\Delta P}{D_2} \left(\frac{D_2^2}{4} - \frac{D_2\sqrt{L_{20}^2 - X^2}}{3} - \frac{2X^2}{5} + \frac{2L_{20}^2}{15} + \frac{D_2X^2}{3\sqrt{L_{20}^2 - X^2}} \right) \tag{19}$$

Note that when $X = L_{20}$, T_1 tends to be infinity. Theoretically, because of the membrane's non-stretchability, the actuator cannot be in the non-compression state under vacuum pressure. Meanwhile, the model is approximate, so there will be rather big errors under small deformation.

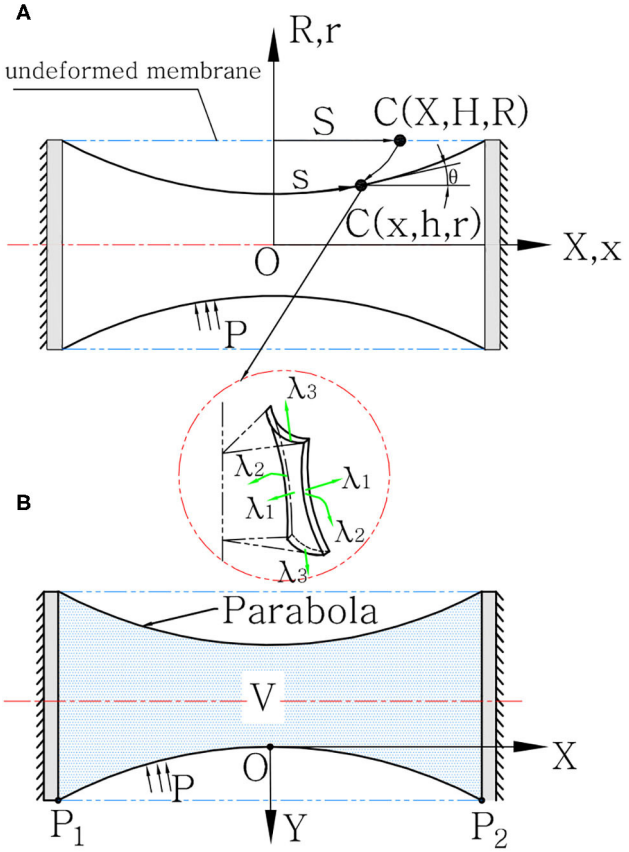


FIGURE 3 | SPCAM's quasi-static model. **(A)** The profiles of undeformed and deformed hyperelastic latex membrane under coordinate systems (X, H, R) and (x, h, r), respectively. The detail view illustrates the principle stretches λ_1, λ_2, and λ_3 of a material particle. **(B)** The profile of deformed polyethylene membrane under the coordinate system (x, y).

4. EVALUATION EXPERIMENTS

4.1. Experimental Setup

A platform was built to test the static characteristics of SPCAM and evaluate the results of finite element analysis, as shown in **Figure 4**. The platform mainly comprises a lifting table, a capacitive displacement transducer, and an external tension sensor. One end of the actuator is fixed to the base, and the other end is connected to the external tension sensor via a pulley, thus we can detect the output force. We used the lifting table to adjust the displacement of the actuator, which would be recorded by the displacement sensor. A vacuum pneumatic proportional valve (SMC, ITV-2090-042BS5, −80 kPa max.) is connected to the miniature vacuum pump (kamoer, KVP08, −82 kPa max.) to control the internal pressure of actuator. Before the experiment, the Hooke's coefficient of spring was measured first. For the membrane material, we chose three kinds of latex membrane with the diameter of 25 mm and the thickness of 0.3, 0.5, 0.8 mm, respectively, and a kind of LDPE membrane with the thickness of 0.2 mm. Five actuators were made in total for test, whose parameters are listed in **Table 2**. We tested the displacement detection accuracy of the built-in sensor and the output force of SPCAM under static equilibrium (considering no effects of dynamic force) and compared it with simulation results. Meanwhile, we displayed the actuator's ability of interference detection and shock resistance.

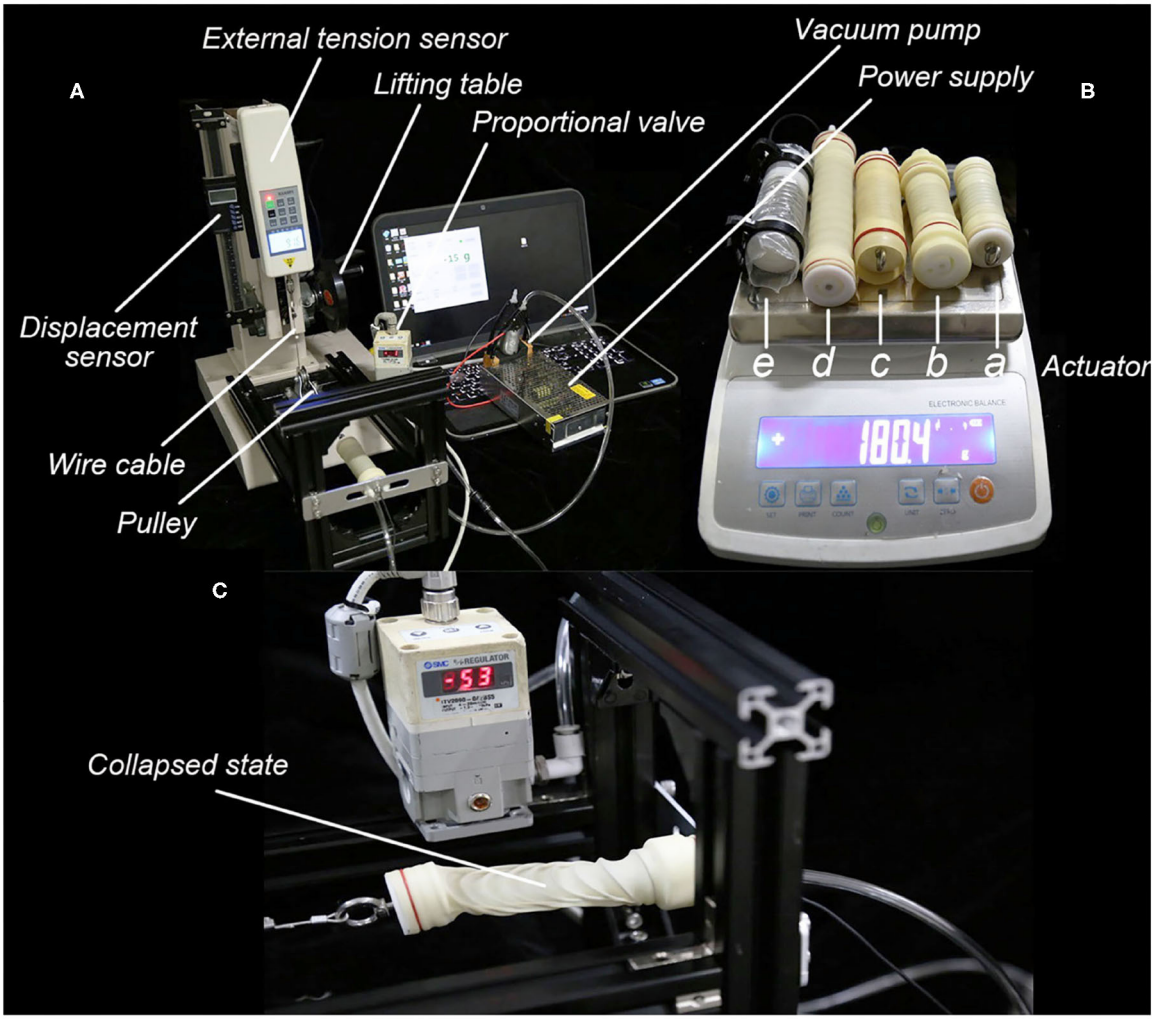


FIGURE 4 | The experimental setup and actuators for test. **(A)** The platform built to test the characteristics of Self-sensing Pneumatic Compressing Artificial Muscle (SPCAM), with the displacement sensor and the external tension sensor. **(B)** Five different actuators to be tested, with the average weight <40 g. Actuator a, b, c, and d are made up of latex membrane with different thickness, while actuator e is made up of polyethylene film. Specific parameters are listed in **Table 2**. **(C)** The collapse phenomenon of latex membrane actuator. When overstretched and under high negative pressure, the actuator collapses sidewise due to the unbalanced force, and is unable to work normally.

TABLE 2 | Parameters of five actuators.

Actuator	Overall length (mm)	Active coil number	Hooke's coefficient (N/m)	Membrane thickness (mm)	Membrane material
a	85	8.1	135.77	0.3	Latex
b	85	8.1	135.77	0.5	Latex
c	85	8.1	135.77	0.8	Latex
d	140	9.5	158.80	0.5	Latex
e	85	8.1	135.77	0.2	LDPE

4.2. Sensor Accuracy Test

Theoretically, the output force of the built-in microsensor should depend linearly on the actuator's displacement, with the scale factor of K (i.e., the Hooke's coefficient of the spring). During the experiment, we applied different pressure to the actuator at each displacement point in the static equilibrium state, and sensor's measured value is expected to be constant regardless of the change of vacuum pressure. Thus, the fluctuation of the measured value caused by pressure change is the main motion error of SPCAM. **Figure 5** shows the relations between detected and actual displacements of five actuators, respectively. Different colors of solid lines represent different pressures, varying from 10 to 60 kPa. The red dashed line with the slope of 45° presents the reference value. The abscissa presents the actual displacement of the steel wire, which is connected to the end of the actuator, so it is also the absolute displacement of the actuator when the wire is tensioned. Its value is measured by a high-precision capacitive sensor. Negative value of the abscissa indicates that the actuator is in contraction state, while positive value means

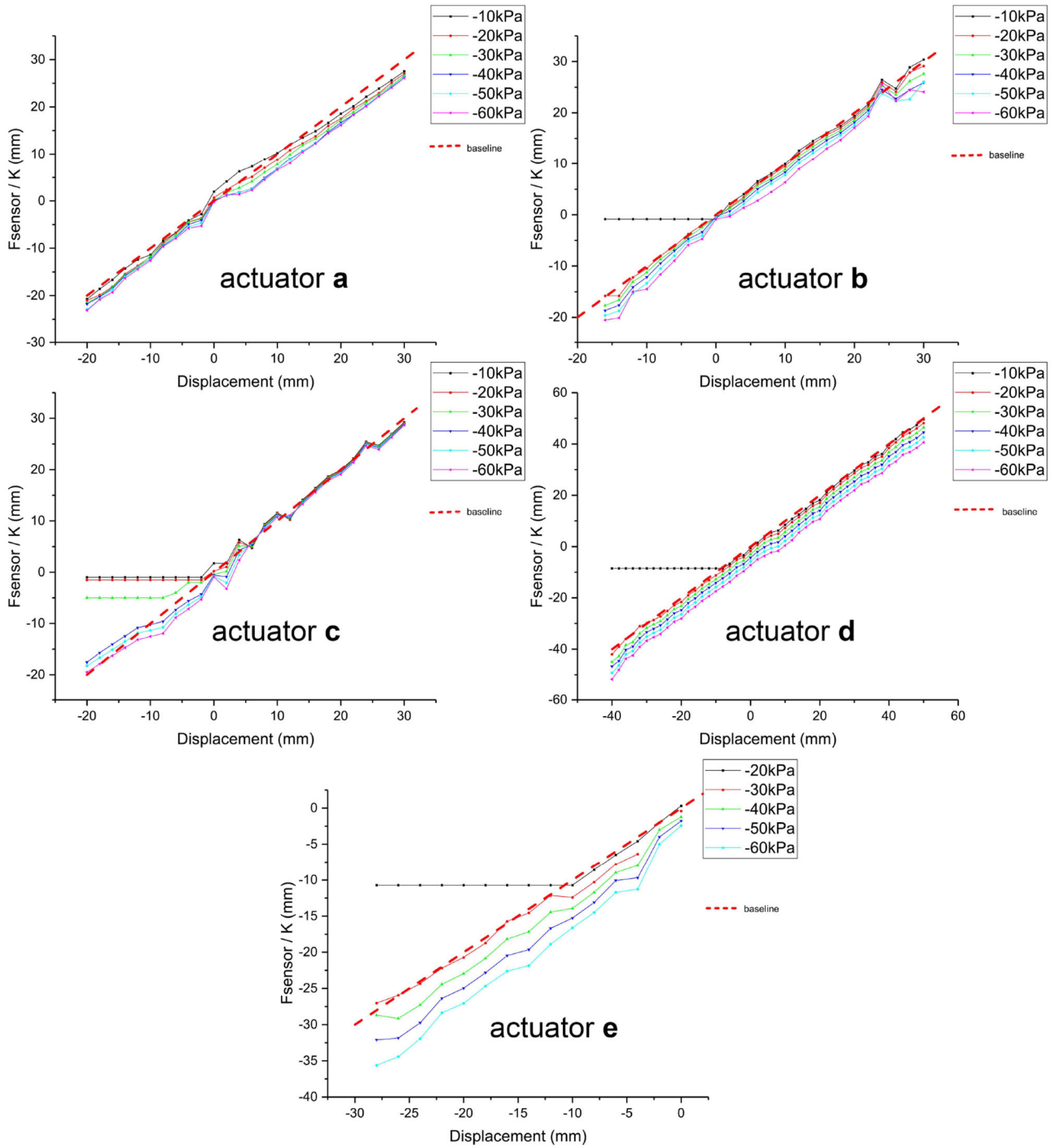


FIGURE 5 | Accuracy of five actuators under quasi-static equilibrium. The displacement detection error of the sensor is represented by the distance between experimental data points and the reference line in the vertical direction. Negative value of the abscissa indicates that the actuator is in contraction state, while positive value means stretching state. Horizontal experimental data line indicates that the actuation force is too weak to actuate the SPCAM, leading to the relaxed state.

stretching state. The ordinates is the measured value F_{sensor} of built-in sensor divided by K. The position error is displayed as the height difference to the base line vertically. For actuators a–c, the maximum position errors are 4.0, 6.1, and 5.2 mm, and root-mean-square errors (RMSEs) are 2.2, 1.8, and 1.3 mm, respectively (erroneous data are excluded). For actuator d with the total length of 140 mm and maximum displacement about 100 mm, the RMSE is 4.5 mm, therefore the relative displacement error is <5%. The error tends to increase with the increase of pressure. This is mainly because the spring twists along

with the contracting (or stretching) process, which changes the spring's structure, thus affects the Hooke's coefficient. Besides, the contraction force T_1 of the latex membrane does not uniformly distribute on the spring due to installation error, which results in an additional lateral force and further increases the error.

Meanwhile, when the negative actuating pressure is too low and the contraction displacement exceeds a certain limit, the actuator will come to a relaxed state and is unable to work normally, like the black curves representing −10 kPa pressure of actuators b–e in **Figure 5**. And this is also directly related to the thickness of the membranes. Negative pressures must apply work on the membrane, so the thicker the membrane, the more energy is needed to deform it, and the more obvious the relaxed phenomenon.

As for different membrane materials, the sensor error of actuator e (maximum position error 8.4 mm, and RMSE 3.4 mm) using polyethylene membrane is obviously larger than that of latex membrane actuators. This is because polyethylene membrane's manufacture process creates hot pressing indentations, which causes its inherent axial non-uniformity, and its inextensibility with incompressibility worsens the situation.

In addition, we observed that if SPCAM with latex membrane was in the stretching state, beyond certain displacement, it would collapse sidewise (**Figure 4C**) due to unbalanced forces (disturbing force and lateral force) when vacuum pressure reached a critical point, similar to buckling deformation of springs. To avoid this situation, the maximum tensile strain of SPCAM should be carefully limited.

4.3. Output Force Test

The actuator can produce output force more than 50 times larger than its weight. The relationship among pressure, displacement, and output force was investigated in output force test, and the result is shown in **Figure 6**. The black surface represents the results of simulation as described above, and the colored surface represents the results of experiments. We can see that the output force obviously increases when vacuum pressure increases. When the spring is in compressing state, output force gradually decreases if displacement increases. On the contrary, when the spring (the latex membrane actuator) is in stretching state, output force increases with the increase of displacement, which resembles animal muscles' characteristics.

Comparing experiment results with finite element analysis, overall trend is the same, but there are still certain differences. Almost all numerals of simulation results are larger than experiment results. This may because some simplifying assumptions used in finite element analysis do not completely conform to actual conditions. For example, the torque τ has severe influence on long actuator d, making the spring twist too much, enlarging the lateral friction between spring and membrane, and largely decreasing the output force. Further analysis will be made to model the friction accurately. At present, under the condition of low output force, the simulation results do not deviate much from the experimental ones.

Besides, the resultant force of the membrane is not completely in the axial direction, which also makes most experiment results generally smaller. As shown in **Figure 6B**, as the thickness of the membrane becomes larger, the output force of the actuator also increases substantially. Because the surface tension of the latex membrane is positively correlated with the membrane thickness, if the membrane is too thin, it cannot resist vacuum force, causing a severe inward deformation and a large angle θ (shown in **Figure 3A**) between the force and the axial, thus reduces the output force (actuator a). However, if the membrane is too thick, it is disadvantageous for low-pressure actuation. As the aforementioned relaxated state phenomenon, when the latex membrane is too thick, the vacuum force is insufficient to deform it. Therefore, actuator c with the membrane thickness of 0.8 mm has almost zero output force under negative pressure of 0–20 kPa when it is in the contraction state. The length of the actuator and the initial screw pitch of the spring also affect the actuator's performance (**Figure 6C**). Long actuators have more effective coils, thus have larger strain. Under the same pressure, with the same contraction displacement or low stretching displacement, output force of actuator d is obviously larger than that of actuator b due to relaxed state phenomenon. But when the stretching displacement is beyond 25 mm, latex membrane of actuator b has larger axial deformation rate. In this case, the tension in the membrane becomes the dominant factor, which makes the output force of actuator b slightly larger than that of actuator d. As a result, long actuator has better performance than short one concerning total output force and maximum displacement. However, actuator d has larger error under the same displacement due to the influence of the torque τ. At the same time, its stability becomes markedly lower and is more likely to collapse.

Although the non-stretchability of polyethylene membrane on actuator e can help enlarge output force, the improvement is not so significant compared to latex membrane, shown in **Figure 6D**. This is because the extensibility of latex makes it possible for the membrane to totally fit on the spring under initial configuration. polyethylene membrane, however, is non-stretchable, so its inner diameter should be a little larger than spring's diameter when installed, which affects output force to some extent. Meanwhile, polyethylene membrane actuators cannot work in the stretching state, which severely impacts actuator's stretching rate. So we prefer to choose latex membrane rather than polyethylene film used in FOAMs (Li et al., 2017) under light load.

4.4. Dynamic Response and Impact Test

We roughly tested the actuator's dynamic performances. A motion capture system (OptiTrack, Prime 17Wx12, 0.5 mm location accuracy) was used to detect the actuator's displacement. Actuator *d* was chosen for the test, with a 1 kg weight on the end. A feedback control algorithm was adopted to control the actuators' position. Results are shown in **Figure 7A**. The actuator is able to lift the 1 kg weight for 75 mm within 6 seconds, only with feedback of the built-in sensor. The motion error is about 2.5 mm, measured by motion capture system. Furthermore, the actuator's dynamic response speed depends on the flow rate of the air pump and proportional valve. If we use air pump with larger flow, the speed of movement can be further increased.

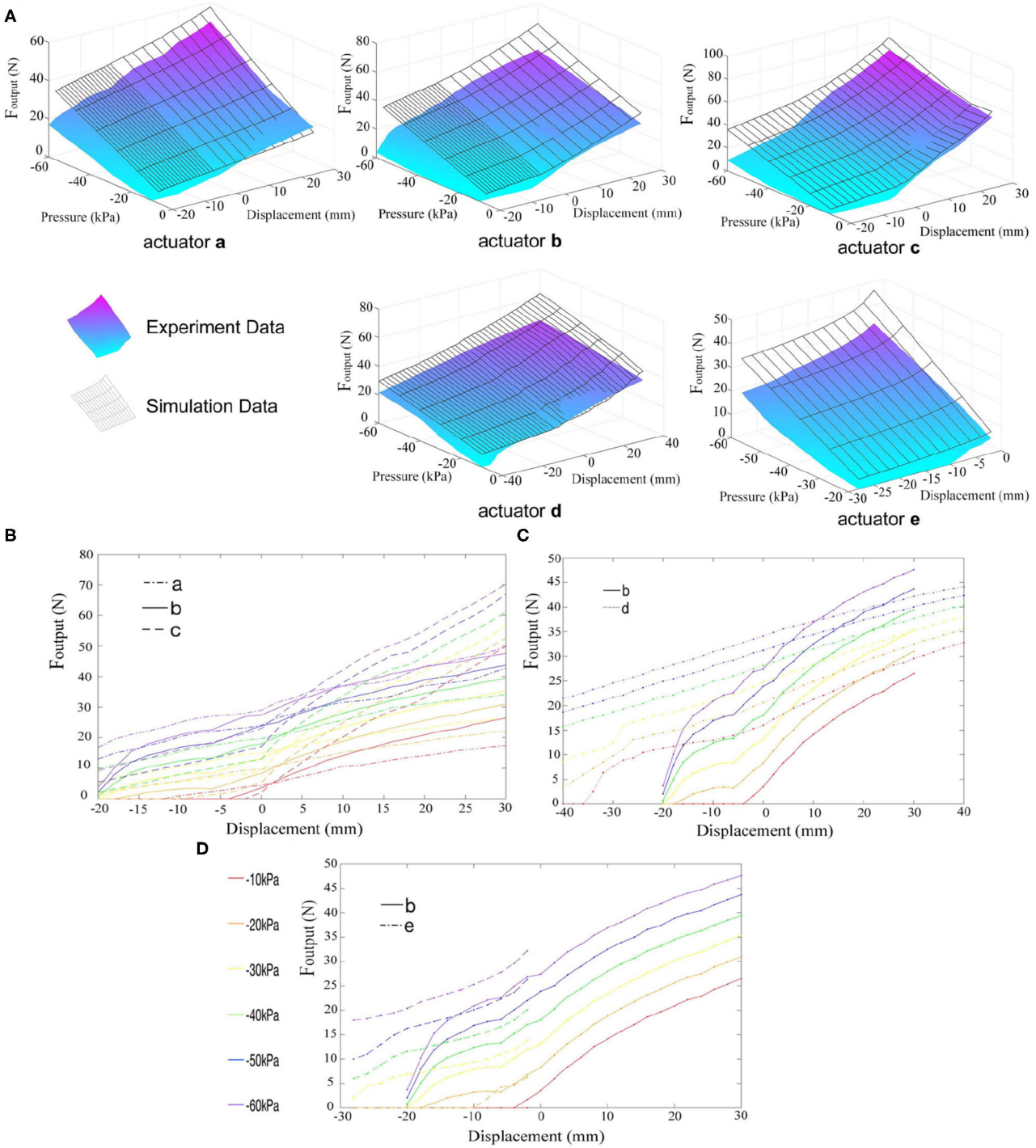


FIGURE 6 | The relations between output force and actuation pressure, displacement of the actuators. **(A)** The simulation results and experiment data of five different actuators. The black surface represents the results of simulation as described above, while the colored surface represents the experimental results. **(B–D)** The comparison of output forces of actuators with different parameters. Different colors represent different actuation pressure, varying from –10 to –60 kPa. Actuators a–c in **(B)** have different membrane thickness (0.3 mm for actuator a, 0.5 mm for actuator b, and 0.8 mm for actuator c). Actuators b and d in **(C)** have different length, and the longer actuator d has larger strain. In **(D)**, actuator b is made up of latex membrane, while actuator e is made up of polyethylene film.

One of the major advantages of the soft actuator is its inherent flexibility. Since the actuator comprises elastic parts like latex membrane and spring, similar to series elastic actuators (Pratt and Williamson, 1995), it can resist external impacts and guarantee safety. Not only can the actuator withstand shocks, it is also able to roughly detect the intensity of the impact, and

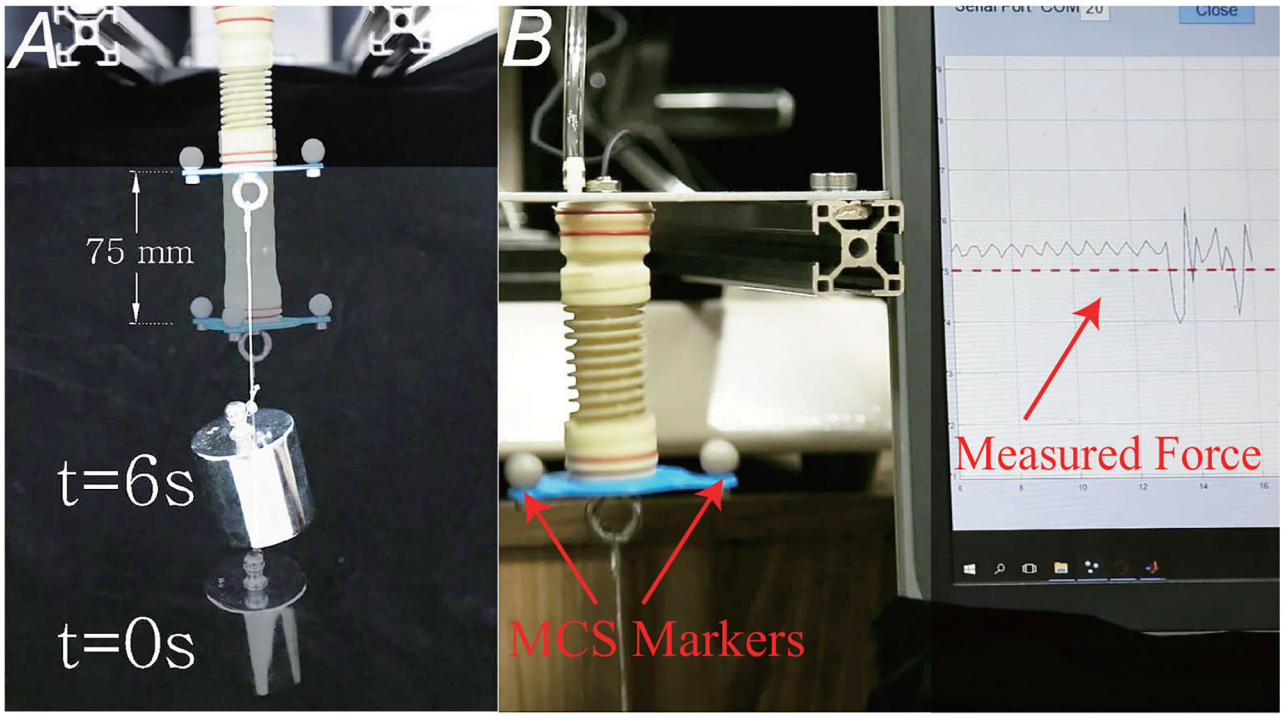


FIGURE 7 | Demonstration of the dynamic performance and impact tests. **(A)** The position before and after actuation. The actuator is able to lift the 1 kg weight for 75 mm within 6 s. **(B)** The actuator is able to withstand shocks, as well as roughly detect the intensity of the impact, and whether it touches the target object or interferes with obstacles. When the external impact happens, the internal sensor's measured value will have a large fluctuation.

whether it touches the target object or interferes with obstacles, as illustrated in **Figure 7B**. When the external impact happens, the internal sensor's measured value will have a large fluctuation, then the detector is triggered. Besides, the impact strength can be considered to be proportional to the fluctuating value.

5. POSITIVE-NEGATIVE PRESSURE COMBINED ACTUATION

Another core advantage of SPCAM is that it can combine negative and positive pressure actuation. Output force of previous SPCAM or FOAMs is limited by maximum negative pressure (–100 kPa max.), while positive pressure can reach a large value. Another issue is when actuating pressure rises, requirements for pressure sources become stricter, which leads to clumsy air pumps or compressors, accompanied by huge noises. Meanwhile, high pressure actuation also brings in potential safety threats. This is undesirable for most mobile robots and wearable devices.

To solve these problems, the method of combined actuation for soft actuator has been first proposed. When there is already negative pressure within SPCAM, we can provide positive pressure outside the membrane for actuation. Obviously, under low-pressure actuating circumstances, positive-negative pressure combined actuation has much lower requirements on devices than negative or positive pressure actuation alone. For example, using an equipment combining 50 kPa positive pressure with –50 kPa negative pressure will have much lower weight, noise, and cost than an air pump or vacuum pump that generates 100 kPa pressure difference. Based on such principles, we designed two different kinds of positive-negative pressure combined actuation structures. This can improve the actuator's loading ability and safety, which greatly broadens the application prospect of the actuator in low-pressure actuating field.

5.1. Lightweight Air Cylinder

First, we designed an actuator combining a traditional air cylinder, whose structure is shown in **Figure 8A**. The wall of the air cylinder is a 180 mm long and 5 mm thick acrylic tube with SPCAM actuator **b** inside of it. A precision-machined shaft with the diameter of 3 mm is connected to the end of SPCAM and a sliding seal structure is adopted at the end face of the acrylic tube. Two sliding bearings are used to constrain shaft's degree of freedom in the radial direction, and a micro variseal is used to seal high-pressure air. By applying positive pressure to the acrylic tube and negative pressure to SPCAM, we can produce larger pressure difference or lower the requirements for air source equipments. **Figure 8B** shows that the actuator can still generate satisfactory output force even both positive and negative pressure difference is under 30 kPa. Meanwhile, due to the effects of sliding bearings, the influence of lateral force reduces sharply and position accuracy of SPCAM is further improved, as can be seen in **Figure 8E**. The RMSE of combined actuator is about 1.0

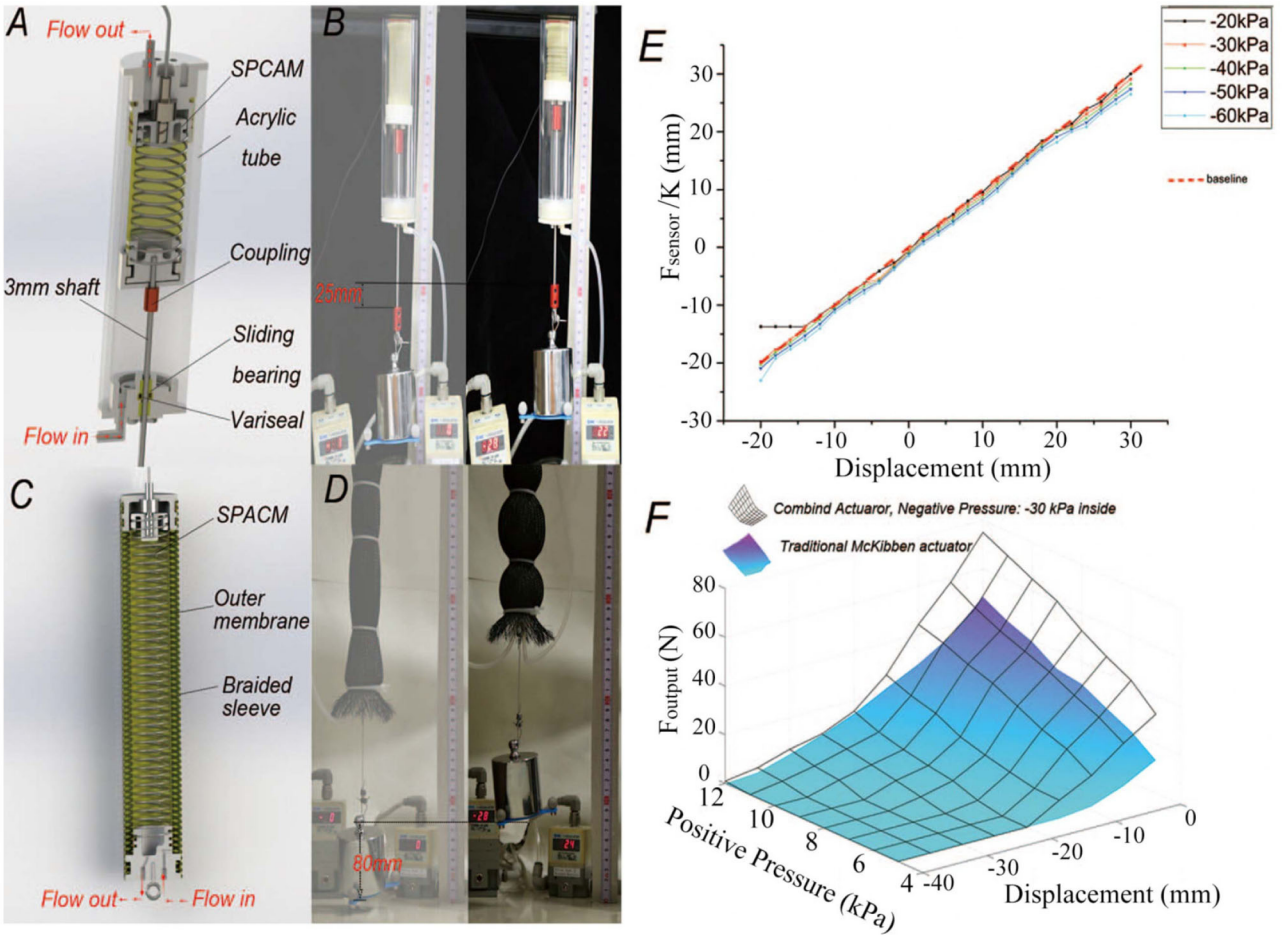


FIGURE 8 | The structures and performance of two different types of combined actuator. **(A)** The schematic of negative–positive pressure combined air cylinder. **(B)** Demonstration of the combined air cylinder actuator. The actuator can still get satisfactory output force and displacement even both positive and negative pressure difference is under 30 kPa. **(C)** The structural drawing of the second type of actuator which combined the SPCAM with McKibben artificial muscle. **(D)** The actual performance of the second combined actuator. It is able to lift the 1 kg weight for about 80 mm. **(E)** The accuracy of air cylinder combined actuator. The RMSE of the combined actuator is about 1.0 mm and the maximum error is 3.5 mm, much smaller than that of original actuator b alone. **(F)** The relation among displacement, actuation pressure, and output force of McKibben-combined actuator and traditional one. Under low-pressure actuation, the output force of the combined actuator (represented by the black surface, -30 kPa vacuum pressure inside) is larger than that of the traditional artificial muscle (colored surface) and maximum contracting displacement also increases.

mm and the maximum error is 3.5 mm, much smaller than that of actuator b.

5.2. McKibben Artificial Muscle

McKibben artificial muscle is a widely used pneumatic actuator. It is usually considered to be biological muscle like for its similarity in real muscle contraction and relaxation. Besides, the basic working mechanism endows itself with great variable compliance dependent on applied pressure. But one of its problems is the need for higher air pressure. A lot of literatures have researched the property of McKibben artificial muscle and tried to improve its performance. Here, we designed a combined actuator having SPCAM put inside the McKibben artificial muscle, as shown in **Figures 8C,D**. The outer part of the combined actuator consists of latex membrane and nylon shell with the diameter of 38 mm, similar to the McKibben artificial muscle in Chou and Hannaford (1996). The internal part is our SPCAM. When applying positive pressure ΔP_1 to the McKibben muscle and negative pressure $-\Delta P_2$ to the SPCAM, the McKibben actuator shortens under positive pressure ΔP_1, while SPCAM contracts under pressure difference $\Delta P_1 + \Delta P_2$, achieving larger output force and contraction.

The Model can be resolved into a SPCAM model and McKibben artificial muscle. The simplified artificial muscle model proposed in Chou and Hannaford (1996) is used to analyze the output force. For the McKibben artificial muscle, we can know from the virtual work principle:

$$\mathrm{d}W_{in} = \Delta P_1 \cdot dV_{McKibben} = dW_{out} = -F_{McKibben} \cdot dX \quad (20)$$

where ΔP_1 is the provided positive pressure, $dV_{McKibben}$ is the volume change of the McKibben Artificial Muscle, and dX is its effective length.

$$F_{McKibben} = -\Delta P_1 \frac{dV_{McKibben}}{dX} \tag{21}$$

The model of braided shell has following geometric relationships:

$$X = b \sin\theta \tag{22}$$

$$D_{McKibben} = \frac{b \sin\theta}{n\pi} \tag{23}$$

where θ is the angle between a braided thread and the cylinder long axis, $D_{McKibben}$ is the diameter of the McKibben actuator cylinder, n is number of turns of a thread, and b is the thread length, same as what has been discussed in Chou and Hannaford (1996).

Approximately consider the artificial muscle as a cylinder, and assume that the deformation of SPCAM does not affect the volume change of the air inside the McKibben artificial muscle, the volume of McKibben actuator can be expressed as:

$$\begin{aligned} V'_{McKibben} &= \frac{1}{4}\pi(D^2_{McKibben} - D^2_{SPCAM})X \\ &= \frac{b^3}{4\pi n^2}\sin^2\theta\cos\theta - \frac{\pi b D^2_{SPCAM}}{4}\cos\theta \end{aligned} \tag{24}$$

Combine Equations (21)–(23):

$$\begin{aligned} F'_{McKibben} &= -\Delta P_1 \frac{dV'_{McKibben}}{dX} = -\Delta P_1 \frac{dV'_{McKibben}/d\theta}{dX/d\theta} \\ &= \frac{\Delta P_1 b^2(2\cos^2\theta - \sin^2\theta)}{4\pi n^2} + \frac{\Delta P_1 \pi b D^2_{SPCAM}}{4}\sin\theta \\ &= \frac{2\Delta P_1 b^2 - 3\Delta P X^2}{4\pi n^2} + \frac{\Delta P_1 \pi X D^2_{SPCAM}}{4} \end{aligned} \tag{25}$$

Meanwhile, the internal SPCAM is under the pressure difference of $\Delta P = \Delta P_1 + \Delta P_2$, and generates output force $F_{SPCAM}(\Delta p, b\sin\theta)$, which can be obtained in modeling analysis section previously. The final output force F_{output} is:

$$F_{output} = F'_{McKibben} + F_{SPCAM} \tag{26}$$

While the output force of the original McKibben artificial muscle is:

$$F_{McKibben} = \frac{2\Delta P_1 b^2 - 3\Delta P X^2}{4\pi n^2} < F_{output} \tag{27}$$

After adopting the combined actuating method, the actuator has not only greater output force but also larger displacement than the traditional McKibben artificial muscle under the same maximum pressure. In the experiment, we tested the performance of combined actuator and traditional McKibben muscle. For the combined SPCAM, the negative pressure $-\Delta P_2$ here is simply set identically as –30 kPa. The relation among the actuator's displacement, actuating pressure, and output force is displayed in **Figure 8F**. It can be seen that under low-pressure actuation, the output force of the combined actuator is larger than that of the traditional artificial muscle and maximum contracting displacement also increases.

6. APPLICATION

In this section, we designed two prototypes to demonstrate the great advantages of SPCAM and positive–negative pressure combined actuation. The first one is a soft robot gripper, which can resist impacts and ensure safety. It can also detect impacts and grasping state, that is, whether it has grasped the object firmly. The second one is a robot joint using Mckibben-SPCAM actuator. Also it has the potential to be used on wearable devices or exoskeletons.

6.1. Dexterous Robot Hand

We designed a two-finger soft gripper with two degrees of freedom (DoFs), weighing about 240 g. The structure of the gripper is shown in **Figure 9A**. The gripper consists of two flexible fingers, tendon-driven mechanism, two SPCAM actuators, and base. On each finger, five 3D-printed hard connectors are used as phalanges. Suction cups are fixed on the phalanges, which share the same vacuum circuit with SPCAMs, which can generate suction force to help better grasp target objects. In order to increase friction force on contact surface, fingers are covered with silica gel skin. When the SPCAM contracts, the effective length of steel wire in the fingers shortens, and the elastic deformation of the PVC board happens between two neighboring 3D-printed connectors, causing the fingers bent and the gripper closed. Since the actuator is able to detect impacts and collisions, the gripper can sense its grasping state, so in an unstructured environment, it can be well adapted and ensure safety.

The robotic gripper is mounted on the end of a six DoFs manipulator (Kinova Company, *MICO*^{2}). We first did experiments on impact resistance during grasping. Based on the detecting method mentioned before, we can find whether the gripper is influenced by impact loads readily. When fierce impacts occur, the actuator's structure still will not be damaged and is able to recover. Furthermore, using the information of force sensor to limit the output force, the gripper is capable of automatically grasping fragile or soft target objects like inflated packaging bags and raw eggs, without crushing them, which shows its dexterity and adaptivity (illustrated in **Figure 9C**).

6.2. Flexible Joint for Exoskeleton

Wearable devices and exoskeletons have been research hotspots in recent years. Here the SPCAM, combined with McKibben artificial muscle, is used on a robotic joint as exoskeleton equipment. The robotic joint has one rotation DoF, which connects the forearm with upper arm (both made up of acrylic board). The two ends of the combined actuator are fixed on the forearm and upper arm, respectively. When the actuator contracts, the forearm is pulled to rotate about the axis of

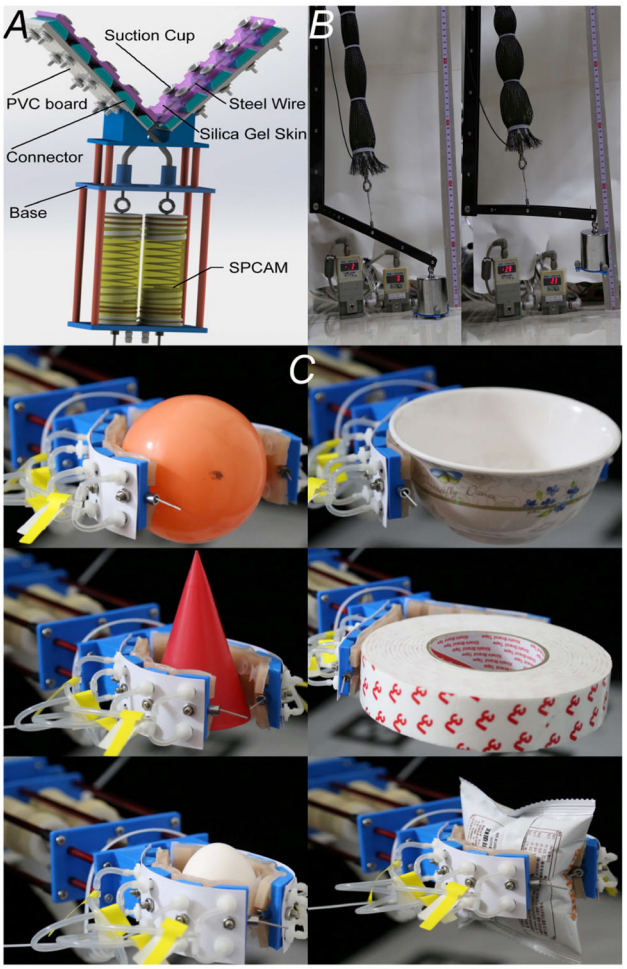


FIGURE 9 | The applications of Self-sensing Pneumatic Compressing Artificial Muscle (SPCAM). **(A)** The structural schematic diagram of the two-finger dexterous robot hand. **(B)** The application of positive–negative pressure combined SPCAM on a robotic joint as exoskeleton equipment, resembling human's muscles. **(C)** Demonstration of the dexterous gripper when grasping different target objects, including fragile and soft ones. From left to right, top to bottom, are rubber ball, bowl, cone, tape, raw egg, and puffed food bag, respectively.

joint, resembling human's muscles (**Figure 9B**). Besides, since the joint is flexible, it also has the ability to sense impacts. In the demonstration, the endpoint of forearm is loaded with a 1 kg weight, and it is able to lift the weight up to 70 mm.

This combined mechanism is suitable for exoskeleton actuator. The wearable devices and exoskeleton need to be lightweight, adaptive, quiet, and energy-saving, which can be well fulfilled by our positive–negative pressure combined actuators. In contrast, other actuators like traditional electric motor cannot guarantee the compliance to users, or they are heavy and costly.

A simple prototype is designed for demonstration. We choose two micro air pumps (ZQ370-03PM, positive pressure 80 kPa max., negative pressure –45 kPa max.). The actuator is able to generate maximum traction about 70 N, which can offer assistance to some extent. The weight of the whole equipments is <500 g, and the working noise is only 52 dB, which accords with the requirements of living environment.

7. CONCLUSION AND FUTURE WORK

Soft actuator is the core technique in soft robot field. In this paper, we proposed a pneumatic actuator SPCAM, which realizes axial contracting movement. Our design uniquely uses stretchable latex as the membrane material for the actuator, bringing it higher strain rate, flexibility, and ability to resist impacts. The actuator has built-in microtension sensors, making it able to detect its absolute displacement and roughly sense the surrounding environment, like collisions and impacts. We established simplified mechanical models under quasi-static equilibrium for latex membrane and polyethylene film, respectively. Finite element analysis has been chosen to build the model for latex membrane due to the material's hyperelasticity, and for polyethylene film, we combined the virtue work principle with geometry approximation. Finally, the output force and interference torque are related with the actuation pressure and other inherent parameters of the actuator. This can help analyze the actuator's operating principle and system error, thus help optimize the design. Five actuators with different parameters are used for experiments, whose result are roughly identical with the simulation results. Latex membrane actuators are obviously better than polyethylene film actuators concerning overall performance, and the performance of actuators with different membrane thickness varies distinctly under different pressures. For example, the actuator with thicker latex membranes has larger output force under high pressure, but works unsatisfyingly under low pressure, while thinner membrane actuator has just the opposite performances. Meanwhile, the length of actuator has significant influence on the maximum displacement and torque. These theoretical and experimental results can guide us to choose proper actuators for different target applications in order to gain the best control effect.

Another essential highlight is the proposal of positive–negative pressure combined actuation, which can effectively reduce the demand of maximum actuation pressure. In the experiments, positive and negative pressure difference both no larger than 30 kPa can reach the same control effect as the normal SPCAM under larger pressure. This also guarantees safety during application like human–robot interaction, and lowers the requirements for air source equipments.

The SPCAM actuator still has some shortcomings. There are certain differences between simulation results and experiment data, mainly because too many simplification hypotheses were used in mechanical analysis, which does not completely accord to reality. A more precise model has to be built in the future in order to further optimize actuator's parameters. Data from pressure sensors and tension sensors also can be combined more effectively. For instance, we can compensate the force interference caused by vacuum pressure by detecting the

actuator's internal pressure, thus obtaining higher detecting accuracy of displacement. Besides, if air source equipment with large flow is used, the actuator's response speed will be largely increased. In this case, the actuator's dynamic performance should be taken into account for it has considerable influence on precision, output force, and stability. Furthermore, in order to verify the actuator's reliability, further fatigue-limit tests should be done.

DATA AVAILABILITY STATEMENT

All datasets presented in this study are included in the article/**Supplementary Material**.

AUTHOR CONTRIBUTIONS

NL and HZ has contributed equally to the core idea as well as the experiment design and results analysis. YL and RW has provided assistance in experiments and analysis, under XZ's supervision. Besides, XZ provided the research group with financial support and experimental equipments, as well as being a supportive corresponding author.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fnbot.2020.572856/full#supplementary-material

REFERENCES

Acome, E., Mitchell, S., Morrissey, T., Emmett, M., Benjamin, C., King, M., et al. (2018). Hydraulically amplified self-healing electrostatic actuators with muscle-like performance. *Science* 359, 61–65. doi: 10.1126/science.aao6139

Ahn, S.-K., Kasi, R. M., Kim, S.-C., Sharma, N., and Zhou, Y. (2008). Stimuli-responsive polymer gels. *Soft Matter* 4, 1151–1157. doi: 10.1039/b714376a

Amjadi, M., Pichitpajongkit, A., Lee, S., Ryu, S., and Park, I. (2014). Highly stretchable and sensitive strain sensor based on silver nanowire–elastomer nanocomposite. *ACS Nano* 8, 5154–5163. doi: 10.1021/nn501204t

Anderson, I. A., Gisby, T. A., McKay, T. G., O'Brien, B. M., and Calius, E. P. (2012). Multi-functional dielectric elastomer artificial muscles for soft and smart machines. *J. Appl. Phys.* 112:041101. doi: 10.1063/1.4740023

Atalay, A., Sanchez, V., Atalay, O., Vogt, D. M., Haufe, F. L., Wood, R. J., et al. (2017). Batch fabrication of customizable silicone-textile composite capacitive strain sensors for human motion tracking. *Adv. Mater. Technol.* 2:1700136. doi: 10.1002/admt.201700136

Belding, L., Baytekin, B., Baytekin, H. T., Rothemund, P., Verma, M. S., Nemiroski, A., et al. (2018). Slit tubes for semisoft pneumatic actuators. *Adv. Mater.* 30, 1704446. doi: 10.1002/adma.201704446

Chossat, J., Park, Y., Wood, R. J., and Duchaine, V. (2013). A soft strain sensor based on ionic and metal liquids. *IEEE Sensors J.* 13, 3405–3414. doi: 10.1109/JSEN.2013.2263797

Chou, C.-P., and Hannaford, B. (1996). Measurement and modeling of mckibben pneumatic artificial muscles. *IEEE Trans. Robot. Autom.* 12, 90–102. doi: 10.1109/70.481753

Cianchetti, M., Ranzani, T., Gerboni, G., Nanayakkara, T., Althoefer, K., Dasgupta, P., et al. (2014). Soft robotics technologies to address shortcomings in today's minimally invasive surgery: the stiff-flop approach. *Soft Robot.* 1, 122–131. doi: 10.1089/soro.2014.0001

Connolly, F., Polygerinos, P., Walsh, C. J., and Bertoldi, K. (2015). Mechanical programming of soft actuators by varying fiber angle. *Soft Robot.* 2, 26–32. doi: 10.1089/soro.2015.0001

Connolly, F., Walsh, C. J., and Bertoldi, K. (2017). Automatic design of fiber-reinforced soft actuators for trajectory matching. *Proc. Natl. Acad. Sci. U.S.A.* 114, 51–56. doi: 10.1073/pnas.1615140114

Cramphorn, L., Wardcherrier, B., and Lepora, N. F. (2016). A biomimetic fingerprint improves spatial tactile perception. *IEEE Robotics and Automation Letters*. 418–423.

Culha, U., Nurzaman, S. G., Clemens, F., and Iida, F. (2014). Svas3: strain vector aided sensorization of soft structures. *Sensors* 14, 12748–12770. doi: 10.3390/s140712748

Daerden, F., and Lefeber, D. (2001). The concept and design of pleated pneumatic artificial muscles. *Int. J. Fluid Power* 2, 41–50. doi: 10.1080/14399776.2001.10781119

Du, Y., Yang, Q., and Huang, J. (2017). Soft prosthetic forefinger tactile sensing via a string of intact single mode optical fiber. *IEEE Sensors J.* 17, 7455–7459. doi: 10.1109/JSEN.2017.2759907

Felt, W., Chin, K. Y., and Remy, C. D. (2016). Contraction sensing with smart braid mckibben muscles. *IEEE ASME Trans. Mechatron.* 21, 1201–1209. doi: 10.1109/TMECH.2015.2493782

Galloway, K. C., Polygerinos, P., Walsh, C. J., and Wood, R. J. (2013). "Mechanically programmable bend radius for fiber-reinforced soft actuators," in *2013 16th International Conference on Advanced Robotics (ICAR)* (IEEE), 1–6.

Gavrilović, M., and Marić, M. (1969). Positional servo-mechanism activated by artificial muscles. *Med. Biol. Eng.* 7, 77–82. doi: 10.1007/BF02474672

Gent, A. (1996). A new constitutive relation for rubber. *Rubber Chem. Technol.* 69, 59–61. doi: 10.5254/1.3538357

Guo, X. (2001). Large deformation analysis for a cylindrical hyperelastic membrane of rubber-like material under internal pressure. *Rubber Chem. Technol.* 74, 100–115. doi: 10.5254/1.3547631

Harnett, C. K., Zhao, H., and Shepherd, R. F. (2017). Stretchable optical fibers: threads for strain-sensitive textiles. *Adv. Mater. Technol.* 2:1700087. doi: 10.1002/admt.201700087

Hawkes, E. W., Christensen, D. L., and Okamura, A. M. (2016). "Design and implementation of a 300% strain soft artificial muscle," in *2016 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE), 4022–4029.

Hill, K. O., and Meltz, G. (1997). Fiber bragg grating technology fundamentals and overview. *J. Lightwave Technol.* 15, 1263–1276. doi: 10.1109/50.618320

Hu, J., Zhu, Y., Huang, H., and Lu, J. (2012). Recent advances in shape–memory polymers: structure, mechanism, functionality, modeling and applications. *Prog. Polym. Sci.* 37, 1720–1763. doi: 10.1016/j.progpolymsci.2012.06.001

Hu, W., Niu, X., Zhao, R., and Pei, Q. (2013). Elastomeric transparent capacitive sensors based on an interpenetrating composite of silver nanowires and polyurethane. *Appl. Phys. Lett.* 102:083303. doi: 10.1063/1.4794143

Ilievski, F., Mazzeo, A. D., Shepherd, R. F., Chen, X., and Whitesides, G. M. (2011). Soft robotics for chemists. *Angew. Chem.* 123, 1930–1935. doi: 10.1002/ange.201006464

Jani, J. M., Leary, M., Subic, A., and Gibson, M. A. (2014). A review of shape memory alloy research, applications and opportunities. *Mater Design* 56, 1078–1113. doi: 10.1016/j.matdes.2013.11.084

Jung, K., Kim, K. J., and Choi, H. R. (2008). A self-sensing dielectric elastomer actuator. *Sensors Actuat. A Phys.* 143, 343–351. doi: 10.1016/j.sna.2007.10.076

Kellaris, N., Venkata, V. G., Smith, G. M., Mitchell, S. K., and Keplinger, C. (2018). Peano-hasel actuators: muscle-mimetic, electrohydraulic transducers that linearly contract on activation. *Sci. Robot.* 3:eaar3276. doi: 10.1126/scirobotics.aar3276

Kim, S., Laschi, C., and Trimmer, B. (2013). Soft robotics: a bioinspired evolution in robotics. *Trends Biotechnol.* 31, 287–294. doi: 10.1016/j.tibtech.2013.03.002

Kim, S.-J., Lee, D.-Y., Jung, G.-P., and Cho, K.-J. (2018). An origami-inspired, self-locking robotic arm that can be folded flat. *Sci. Robot.* 3:eaar2915. doi: 10.1126/scirobotics.aar2915

Kramer, R. K., Majidi, C., Sahai, R., and Wood, R. J. (2011a). Soft curvature sensors for joint angle proprioception. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE), 1919–1926.

Kramer, R. K., Majidi, C., and Wood, R. J. (2011b). Wearable tactile keypad with stretchable artificial skin. In *2011 IEEE International Conference on Robotics and Automation* (IEEE), 1103–1107.

Kruusamäe, K., Punning, A., Aabloo, A., and Asaka, K. (2015). Self-sensing ionic polymer actuators: a review. *Actuators* 4, 17–38. doi: 10.3390/act4010017

Li, S., Vogt, D. M., Rus, D., and Wood, R. J. (2017). Fluid-driven origami-inspired artificial muscles. *Proc. Natl. Acad. Sci. U.S.A.* 114, 13132–13137. doi: 10.1073/pnas.1713450114

Lipomi, D. J., Vosgueritchian, M., Tee, B. C. K., Hellstrom, S. L., Lee, J. A., Fox, C. H., et al. (2011). Skin-like pressure and strain sensors based on transparent elastic films of carbon nanotubes. *Nat. Nanotechnol.* 6, 788–792. doi: 10.1038/nnano.2011.184

Liu, H., Li, J., Song, X., Seneviratne, L., and Althoefer, K. (2011). Rolling indentation probe for tissue abnormality identification during minimally invasive surgery. *IEEE Trans. Robot.* 27, 450–460. doi: 10.1109/TRO.2011.2127210

Luo M., Skorinaerik, E. H., Tao, W., Chen, F., Ozel, S., Sun, Y., et al. (2017). Toward modular soft robotics: proprioceptive curvature sensing and sliding-mode control of soft bidirectional bending modules. *Soft Robot.* 4, 117–125. doi: 10.1089/soro.2016.0041

Majidi, C., Kramer, R. K., and Wood, R. J. (2011). A non-differential elastomer curvature sensor for softer-than-skin electronics. *Smart Mater. Struct.* 20:105017. doi: 10.1088/0964-1726/20/10/105017

Martinez, R. V., Fish, C. R., Chen, X., and Whitesides, G. M. (2012). Elastomeric origami: programmable paper-elastomer composites as pneumatic actuators. *Adv. Funct. Mater.* 22, 1376–1384. doi: 10.1002/adfm.201102978

Miyashita, S., Guitron, S., Li, S., and Rus, D. (2017). Robotic metamorphosis by origami exoskeletons. *Sci. Robot.* 2:eaao4369. doi: 10.1126/scirobotics.aao4369

Mooney, M. (1940). A theory of large elastic deformation. *J. Appl. Phys.* 11, 582–592. doi: 10.1063/1.1712836

Mosadegh, B., Polygerinos, P., Keplinger, C., Wennstedt, S., Shepherd, R. F., Gupta, U., et al. (2014). Pneumatic networks for soft robotics that actuate rapidly. *Adv. Funct. Mater.* 24, 2163–2170. doi: 10.1002/adfm.201303288

Mu, J., Hou, C., Wang, H., Li, Y., Zhang, Q., and Zhu, M. (2015). Origami-inspired active graphene-based paper for programmable instant self-folding walking devices. *Sci. Adv.* 1:e1500533. doi: 10.1126/sciadv.1500533

Ogden, R. W. (1997). *Non-linear Elastic Deformations.* Dover Civil and Mechanical Engineering (Dover Publications).

O'Halloran, A., O'Malley, F., and McHugh, P. (2008). A review on dielectric elastomer actuators, technology, applications, and challenges. *J. Appl. Phys.* 104:9. doi: 10.1063/1.2981642

Onal, C. D., Wood, R. J., and Rus, D. (2013). An origami-inspired approach to worm robots. *IEEE ASME Trans. Mechatron.* 18, 430–438. doi: 10.1109/TMECH.2012.2210239

Ozel, S., Keskin, N. A., Khea, D., and Onal, C. D. (2015). A precise embedded curvature sensor module for soft-bodied robots. *Sensors Actuat. A Phys.* 236, 349–356. doi: 10.1016/j.sna.2015.09.041

Paez, L., Agarwal, G., and Paik, J. (2016). Design and analysis of a soft pneumatic actuator with origami shell reinforcement. *Soft Robot.* 3, 109–119. doi: 10.1089/soro.2016.0023

Pang, M., Zhang, M., Wang, L., Zou, Q., Kuang, W., Wang, D. N., et al. (2007). Phase mode-matching demodulation scheme for interferometric fiber-optic sensors. *IEEE Photon. Technol. Lett.* 19, 39–41. doi: 10.1109/LPT.2006.889003

Park, Y.-L., Chen, B.-r., Pérez-Arancibia, N. O., Young, D., Stirling, L., Wood, R. J., et al. (2014). Design and control of a bio-inspired soft wearable robotic device for ankle–foot rehabilitation. *Bioinspir. Biomimet.* 9:016007. doi: 10.1088/1748-3182/9/1/016007

Polygerinos, P., Ataollahi, A., Schaeffter, T., Razavi, R., Seneviratne, L., and Althoefer, K. (2011). Mri-compatible intensity-modulated force sensor for cardiac catheterization procedures. *IEEE Trans. Biomed. Eng.* 58, 721–726. doi: 10.1109/TBME.2010.2095853

Polygerinos, P., Correll, N., Morin, S. A., Mosadegh, B., Onal, C. D., Petersen, K., et al. (2017). Soft robotics: Review of fluid-driven intrinsically soft devices; manufacturing, sensing, control, and applications in human-robot interaction. *Adv. Eng. Mater.* 19:1700016. doi: 10.1002/adem.201700016

Pratt, G. A., and Williamson, M. M. (1995). "Series elastic actuators," in *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots* (IEEE) 1, 399–406.

Rahimi, R., Ochoa, M., Yu, W., and Ziaie, B. (2014). A sewing-enabled stitch-and-transfer method for robust, ultra-stretchable, conductive interconnects. *J. Micromechan. Microeng.* 24:095018. doi: 10.1088/0960-1317/24/9/095018

Rus, D., and Tolley, M. T. (2015). Design, fabrication and control of soft robots. *Nature* 521:467. doi: 10.1038/nature14543

Saad, R. E., Bonen, A., Smith, K. C., and Benhabib, B. (1995). Distributed-force recovery for a planar photoelastic tactile sensor. *IEEE Trans. Instrum. Meas.* 45, 541–546. doi: 10.1109/19.492783

Sanan, S., Lynn, P. S., and Griffith, S. T. (2014). Pneumatic torsional actuators for inflatable robots. *J. Mech. Robot.* 6:031003. doi: 10.1115/1.4026629

Sareh, S., Jiang, A., Faragasso, A., Noh, Y., Nanayakkara, T., Dasgupta, P., et al. (2014). Bio-inspired tactile sensor sleeve for surgical soft manipulators. In *2014 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE), 1454–1459.

Shapiro, Y., Kosa, G., and Wolf, A. (2014). Shape tracking of planar hyper-flexible beams via embedded pvdf deflection sensors. *IEEE ASME Trans. Mechatron.* 19, 1260–1267. doi: 10.1109/TMECH.2013.2278251

Soleimani, M., and Funnell, W. R. J. (2016). Deformation and stability of short cylindrical membranes. *Int. J. Mechan. Sci.* 119, 266–272. doi: 10.1016/j.ijmecsci.2016.10.017

Sun, Y., Song, Y. S., and Paik, J. (2013). "Characterization of silicone rubber based soft pneumatic actuators," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE), 4446–4453.

Teeple, C. B., Becker, K. P., and Wood, R. J. (2018). Soft curvature and contact force sensors for deep-sea grasping via soft optical waveguides. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 1621–1627.

Terryn, S., Brancart, J., Lefeber, D., Van Assche, G., and Vanderborght, B. (2018). A pneumatic artificial muscle manufactured out of self-healing polymers that can repair macroscopic damages. *IEEE Robot. Autom. Lett.* 3, 16–21. doi: 10.1109/LRA.2017.2724140

Tsouti, V., Mitrakos, V., Broutas, P., and Chatzandroulis, S. (2016). Modeling and development of a flexible carbon black-based capacitive strain sensor. *IEEE Sensors J.* 16, 3059–3067. doi: 10.1109/JSEN.2016.2524508

Villegas, D., Van Damme, M., Vanderborght, B., Beyl, P., and Lefeber, D. (2012). Third–generation pleated pneumatic artificial muscles for robotic applications: development and comparison with mckibben muscle. *Adv. Robot.* 26, 1205–1227. doi: 10.1080/01691864.2012.689722

Vogt, D. M., Park, Y., and Wood, R. J. (2013). Design and characterization of a soft multi-axis force sensor using embedded microfluidic channels. *IEEE Sensors J.* 13, 4056–4064. doi: 10.1109/JSEN.2013.2272320

Wall, V., Zoller, G., and Brock, O. (2017). A method for sensorizing soft actuators and its application to the rbo hand 2. In *2017 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE), 4965–4970.

Wang, H., De Boer, G., Kow, J., Alazmani, A., Ghajari, M., Hewson, R. W., et al. (2016a). Design methodology for magnetic field-based soft tri-axis tactile sensors. *Sensors* 16:1356. doi: 10.3390/s16091356

Wang, H., Totaro, M., and Beccai, L. (2018). Toward perceptive soft robots: progress and challenges. *Adv. Sci.* 5:1800541. doi: 10.1002/advs.201800541

Wang, H., Zhang, R., Chen, W., Liang, X., and Pfeifer, R. (2016b). Shape detection algorithm for soft manipulator based on fiber bragg gratings. *IEEE ASME Trans. Mechatron.* 21, 2977–2982. doi: 10.1109/TMECH.2016.2606491

Wardcherrier, B., Cramphorn, L., and Lepora, N. F. (2017). Exploiting sensor symmetry for generalized tactile perception in biomimetic touch. *IEEE Robotics and Automation Letters* 2, 1218–1225.

Wardcherrier, B., Pestell, N., Cramphorn, L., Winstone, B., Giannaccini, M. E., Rossiter, J., et al. (2018). The tactip family: soft optical tactile sensors with 3d-printed biomimetic morphologies. *Soft Robot.* 5, 216–227. doi: 10.1089/soro.2017.0052

Wong, R. D. P., Posner, J. D., and Santos, V. J. (2012). Flexible microfluidic normal force sensor skin for tactile feedback. *Sensors Actua. A Phys.* 179, 62–69. doi: 10.1016/j.sna.2012.03.023

Yamamoto, Y., Kure, K., Iwai, T., Kanda, T., and Suzumori, K. (2007). Flexible displacement sensor using piezoelectric polymer for intelligent fma. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems.* 765–770.

Yeo, J. C., Kenry, Yu, J., Loh, K. P., Wang, Z., and Lim, C. T. (2016). Triple-state liquid-based microfluidic tactile sensor with high flexibility, durability, and sensitivity. *ACS Sensors* 1, 543–551. doi: 10.1021/acssensors.6b00115

Yi, J., Chen, X., Song, C., and Wang, Z. (2018). Fiber-reinforced origamic robotic actuator. *Soft Robot.* 5, 81–92. doi: 10.1089/soro.2016.0079

Zhao, H., Jalving, J., Huang, R., Knepper, R. A., Ruina, A., and Shepherd, R. F. (2016a). A helping hand: soft orthosis with integrated optical strain sensors and EMG control. *IEEE Robot. Autom. Magaz.* 23, 55–64. doi: 10.1109/MRA.2016.2582216

Zhao, H., O'Brien, K., Li, S., and Shepherd, R. F. (2016b). Optoelectronically innervated soft prosthetic hand via stretchable optical waveguides. *Sci. Robot.* 1:eaai7529. doi: 10.1126/scirobotics.aai7529

Zhuang, W., Sun, G., Li, H., Lou, X., Dong, M., and Zhu, L. (2018). Fbg based shape sensing of a silicone octopus tentacle model for soft robotics. *Optik* 165, 7–15. doi: 10.1016/j.ijleo.2018.03.087

Zook, J. D., Herb, W. R., and Burns, D. W. (2000). Fiber-optic vibration sensor based on frequency modulation of light-excited oscillators. *Sensors Actuat. A Phys.* 83, 270–276. doi: 10.1016/S0924-4247(99)00379-9

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

frontiers in Neurorobotics

ORIGINAL RESEARCH
published: 15 January 2021
doi: 10.3389/fnbot.2020.600885

Gait Optimization Method for Humanoid Robots Based on Parallel Comprehensive Learning Particle Swarm Optimizer Algorithm

Chongben Tao [1,2], Jie Xue [1], Zufeng Zhang [1,3,4], Feng Cao [5], Chunguang Li [6] and Hanwen Gao [1]*

[1] School of Electronic and Information Engineering, Suzhou University of Science and Technology, Suzhou, China, [2] Suzhou Automobile Research Institute, Tsinghua University, Suzhou, China, [3] Department of Automation, Tsinghua University, Beijing, China, [4] Wuhan Electronic Information Institute, Hubei, China, [5] School of Computer and Information Technology, Shanxi University, Taiyuan, China, [6] School of Computer Information and Engineering, Changzhou Institute of Technology, Changzhou, China

To improve the fast and stable walking ability of a humanoid robot, this paper proposes a gait optimization method based on a parallel comprehensive learning particle swarm optimizer (PCLPSO). Firstly, the key parameters affecting the walking gait of the humanoid robot are selected based on the natural zero-moment point trajectory planning method. Secondly, by changing the slave group structure of the PCLPSO algorithm, the gait training task is decomposed, and a parallel distributed multi-robot gait training environment based on RoboCup3D is built to automatically optimize the speed and stability of bipedal robot walking. Finally, a layered learning approach is used to optimize the turning ability of the humanoid robot. The experimental results show that the PCLPSO algorithm achieves a quickly optimal solution, and the humanoid robot optimized possesses a fast and steady gait and flexible steering ability.

Keywords: RoboCup3D, humanoid robot, PCLPSO, parallel distributed, layered learning

OPEN ACCESS

Edited by:
Keke Tang,
Guangzhou University, China

Reviewed by:
Yaguang Zhu,
Chang'an University, China
Lan Anh Trinh,
Mälardalen University College, Sweden
Eiji Uchibe,
Advanced Telecommunications Research Institute International (ATR), Japan

***Correspondence:**
Zufeng Zhang
zhangzufeng@tsari.tsinghua.edu.cn

Received: *31 August 2020*
Accepted: *07 December 2020*
Published: *15 January 2021*

Citation:
Tao C, Xue J, Zhang Z, Cao F, Li C and Gao H (2021) Gait Optimization Method for Humanoid Robots Based on Parallel Comprehensive Learning Particle Swarm Optimizer Algorithm. Front. Neurorobot. 14:600885. doi: 10.3389/fnbot.2020.600885

INTRODUCTION

Gait planning is a research hotspot for humanoid robots, and it provides some technical support for humanoid robots walking like humans. The methods of gait planning can be broadly divided into three categories: human walking parameter-based methods (Baoping et al., 2015; Hereid et al., 2018), humanoid walking model-based methods (Sato et al., 2010; Winkler et al., 2018), and intelligent algorithm-based methods (Huan and Anh, 2015; Elhosseini et al., 2019). The method based on human walking parameters makes the gait of humanoid robots more similar to the way humans walk, but it costs a lot of time to find suitable gait parameters from human walking data and apply them to humanoid robots. Paparisabet et al. (2019) proposed a similar function for human-like motion, formulated kinematic constraints for humanoid robots in contact with the ground, and finally proposed humanoid walking with very high similarity to human motion. In Weon and Lee (2018), Weon et al. proposed a method for generating humanoid robot motion based on motion capture data, which corrects extracted joint trajectories based on a reprogrammed zero-moment point (ZMP) trajectory. Researchers have extensively investigated bipedal walking model-based approaches. In Graf and Röfer (2011), the three-dimensional linear inverted pendulum model (3D-LIPM) proposed is one of the most widely used simplified dynamics models for humanoid robots. The 3D-LIPM approximates the humanoid robot in the three-dimensional space to an inverted pendulum model composed of mass points and massless legs connecting the points to the support

points and constrains the center of mass to move on the constrained plane. Jadidi and Hashemi (2016) proposed a closed-loop 3D-LIPM gait for RoboCup standard platform and implemented a full range of walking on the NAO robot. The 25 degrees of freedom of the NAO robot make it have excellent omnidirectional walking and full-body motion performance. The RoboCup 3D simulation team uses the NAO robot as a reference model. At the same time, this model is also widely used in robot simulation competitions at all levels at home and abroad. Astudillo et al. (2018) used the 3D-LIPM as a model for robot and ZMP to map joint angles to achieve a humanoid robot walking on a slippery platform.

As the degrees of freedom of humanoid robots increase, the complexity of systems will increase as well. Bipedal walking model-based methods will not be sufficient for the development of humanoid robot control (Fayong et al., 2014). Besides, a variety of intelligent control methods are developed, which do not require accurate modeling (MacAlpine et al., 2015; Hong and Lee, 2016; Bonyadi and Michalewicz, 2017). However, the process of adjusting motion parameters and posture based on various models is very tedious and time-consuming. When the given parameters are not reasonable, walking instability and robots moving at a low speed may occur. This shortcoming can have a significant impact on the coordination between the needs of speed, stability, and flexibility. Therefore, various intelligent algorithms are used for robot gait planning. Many researchers have applied the central pattern generator (CPG) to generate gait trajectories, but the parameter optimization of this method is a challenge (Bai et al., 2019). Zhong et al. (2017) transformed phase signal from CPG output into a trajectory signal for the legs of a six-legged robot by adjusting it. Wang et al. (2019) proposed a gait planning method based on a reactive neuromuscular controller and CPG to achieve a power-saving human-like large walk, and the controller parameters were optimized based on an optimization algorithm in the paper. Common optimization algorithms such as central force optimization (CFO) and genetic algorithm (GA) have also been successfully used for the gait planning of humanoid robots. Kumar et al. (2018) applied GA to optimize parameters for a triple-linked humanoid robot to achieve numerical simulation of energy-controlled stable walking. Huan et al. (2018a) used CFO to optimize the foot lift amplitude of a humanoid robot, which caused an efficient and stable gait. PSO is a common intelligent optimization algorithm that solves global optimization problems simply and efficiently (Kennedy and Eberhart, 1995). Huan et al. (2018b) applied PSO to optimize joint angles to achieve stable walking for a humanoid robot with 10 degrees of freedom. Mandava et al. proposed a multi-objective particle swarm optimization algorithm method for the gait optimization of humanoid robots for the trolley table model. This method uses a sliding mode controller to optimize robust tracking control and realizes the 3D simulation walking of a humanoid robot (Mandava and Vundavilli, 2018). Gülcü and Kodaz (2015) improved the performance of comprehensive learning particle swarm optimizer (CLPSO) through parallel computation and proposed PCLPSO. Most of the studies mentioned earlier are devoted to single movements such as straight, rotating, and going up and down stairs (Faraji et al., 2019). There are relatively few studies that comprehensively consider bipedal robot forward and rotation and their arbitrary motion connection transitions. Also, during the simulation process, the individual simulation platforms always add noise in the same way, resulting in similar movements of the bipedal robots. Parallel optimization algorithms can be a good solution to this problem. Muniz et al. (2016) optimized the keyframe movements of a humanoid robot (getting up, kicking, etc.) through parallelization to improve the motion performance of the robot. However, parallel algorithms suffer from low fault tolerance when dealing with distributed tasks in RoboCup3D, and it is difficult to ensure the correctness and stability of the running process.

Based on the considerations mentioned earlier, this p*aper se*lects 13 key parameters that affect speed and stability based on the gait planning method of natural ZMP trajectories and designs two evaluation functions to address the problems of humanoid robot walking. By changing the cluster structure of the PCLPSO algorithm, a parallel distributed multi-robot gait training environment is established by using the RoboCup3D simulation platform. The training environment consists of multiple computers that can operate independently. The nodes use the computer network for information transfer to achieve a common task (gait optimization for humanoid robots). The computational efficiency is improved by running in parallel in a distributed environment. A layered learning approach is used to optimize the evaluation function layer by layer. Experimental results show that the optimized humanoid robot has a faster and more stable straight gait and excellent turning ability and has less wobble when switching between straight and turning.

GAIT PARAMETER SELECTION BASED ON NATURAL ZERO-MOMENT POINT TRAJECTORY PLANNING

In this paper, after setting a natural ZMP trajectory from heel to toe movement based on a 3D-LIPM in the single-leg support phase, a mass-centered trajectory equation is obtained (Graf and Röfer, 2011). In the double-leg support phase, a linear pendulum model was used to generate mass-centered trajectory equations. Equations for multistep planning of mass-centered trajectories in a unified coordinate system are also given. After planning the walking trajectory by natural ZMP-based mass-centered trajectory planning method, the key gait parameters are selected and optimized based on the experience of manual tuning.

Multistep Trajectory Planning in a Unified Coordinate System

During the movement of the humanoid robot, if only the front and back and up and down movements are considered and the left and right movements are ignored, it is easy to cause the robot to lose its balance and fall. Therefore, it is

necessary to extend the linear inverted pendulum to a three-dimensional environment and model the robot with a three-dimensional linear inverted pendulum. However, the process of directly analyzing and researching the multi-link structure of biped robots is often cumbersome, so this article equivalently simplified the robot to a reasonable mathematical model, which is convenient for research. Regarding the body of the robot as a mass point and the legs as massless support rods, a three-dimensional linear inverted pendulum model can be built. It does not need to know the parameters of the robot, such as the mass and the inertia of each joint, but is derived from the model for easy calculation. According to 3D-LIPM, a humanoid robot is simplified to an inverted pendulum with only a center of mass and a retractable massless pendulum, and the height of mass is assumed to remain constant, as shown in **Figure 1A**.

Assume that the height of the center of mass is fixed at z_s and the acceleration of gravity is g, the equation of motion between ZMP trajectory $p_{sx}(t)$ and the center of mass in the x-axis direction is:

$$p_{sx}(t) = x_s(t) - \frac{z_s}{g}\ddot{x}_s(t) \tag{1}$$

To make the ZMP trajectory of humanoid robot walking similar to that of human walking, this paper uses a linear equation to represent the ZMP trajectory, as shown in **Figure 1B**. Assuming that in the single-leg support phase, ZMP moves in the x-axis direction in the range of $[p_{sx\,min},\ p_{sx\,max}]$, the walking period of the single-leg support phase is T_s; the center of the foot is the origin of a coordinate system, and the trajectory of ZMP is given by the following equation:

$$p_{sx}(t) = b_1 t + b_0 \tag{2}$$

where, $t \in [0, T_s]$, $b_0 = p_{sx\,min}$, $b_1 = (p_{sx\,max} - p_{sx\,min})/T_s$.

By defining $w_s = \sqrt{\frac{z_s}{g}}$ by substituting Equation (1) into Equation (2) and solving a differential equation, one has:

$$x_s(t) = C_1 e^{t/w_s} + C_2 e^{-t/w_s} + b_1 t + b_0 \tag{3}$$

$$\dot{x}_s(t) = C_1 e^{t/w_s}/w_s - C_2 e^{-t/w_s}/w_s + b_1 \tag{4}$$

If the position and velocity $x_s(0)$ of the center of mass at the initial moment of the single-leg support phase and $\dot{x}_s(0)$ are known, then one has:

$$C_1 = \left(\left(x_s(0) - b_0\right) + \left(\dot{x}_s(0) - b_1\right) w_s\right)/2 \tag{5}$$

$$C_2 = \left(\left(x_s(0) - b_0\right) - \left(\dot{x}_s(0) - b_1\right) w_s\right)/2 \tag{6}$$

If the positions of the center of mass at the initial moment of single-leg support phase and the positions of the center of mass at the moment of termination $x_s(0)$ and $x_s(T_s)$ are known, there are:

$$C_1 = \left(\left(x_s(0) - b_0\right) e^{-T_s/w_s} - \left(x_s(T_s) - b_1 T_s - b_0\right)/ \left(e^{-T_s/w_s} - e^{T_s/w_s}\right)\right. \tag{7}$$

$$C_2 = \left(\left(x_s(0) - b_0\right) e^{T_s/w_s} - \left(x_s(T_s) - b_1 T_s - b_0\right)/ \left(e^{T_s/w_s} - e^{-T_s/w_s}\right)\right. \tag{8}$$

According to Equations (2–4), the centroid trajectory planning based on natural ZMP can be realized in the single-leg support phase.

The direct application of the single-leg support phase method for walking requires the assumption that the support leg switch is instantaneous. This will cause the center of mass acceleration to jump from the maximum to the minimum. To obtain a smooth center-of-mass velocity trajectory, the legs support phase is introduced. In this paper, a linear pendulum model is used to realize natural ZMP trajectory planning of the double-leg supporting phase.

In the double-leg support phase, according to the linear pendulum model, as shown in **Figure 1C**, the equation of the relationship between the position of robot center of mass and acceleration is given as follows:

$$x_d(t) - D = \frac{z_d}{g}\ddot{x}_d(t) \tag{9}$$

where $z_d < 0$, $t \in [0, T_d]$, T_d is double-leg support phase walk period, and D is an x-axis coordinate of the fixed end of the linear pendulum.

With $w_d = \sqrt{\frac{-z_d}{g}}$, from Equation (9) can have:

$$x_d(t) = (x_d(0) - D)\cos(t/w_d) + \dot{x}_d(0)\, w_d \sin(t/w_d) + D \tag{10}$$

$$\dot{x}_d(t) = \dot{x}_d(0)\cos(t/w_d) - \frac{x_d(0) - D}{w_d}\sin(t/w_d) \tag{11}$$

In the double-leg support phase, the starting position and speed and the ending position and the speed of the center of mass can be known, that is, $x_d(0)$, $\dot{x}_d(0)$, $x_d(T_d)$, and $\dot{x}_d(T_d)$ can be known.

$$D = \frac{\dot{x}_d(0)^2 w_d^2 - \dot{x}_d(T_d)^2 w_d^2 + x_d(0)^2 - x_d(T_d)^2}{2(x_d(0) - x_d(T_d))} \tag{12}$$

$$T_d = w_d \arccos \frac{(x_d(T_d) - D)(x_d(0) - D) + \dot{x}_d(0)\,\dot{x}_d(T_d)\, w_d^2}{(x_d(0) - D)^2 + \dot{x}_d(0)^2 w_d^2} \tag{13}$$

A smooth center-of-mass trajectory based on natural ZMP trajectory can be achieved in the double-leg support phase according to Equations (10–13).

The methods mentioned earlier have their own coordinate systems in the single-leg support phase and double-leg support phase, which do not facilitate multistep planning calculations for footprint planning. To do so, the equations mentioned earlier need to be unified in the same coordinate system. The equation for the position and velocity of the center

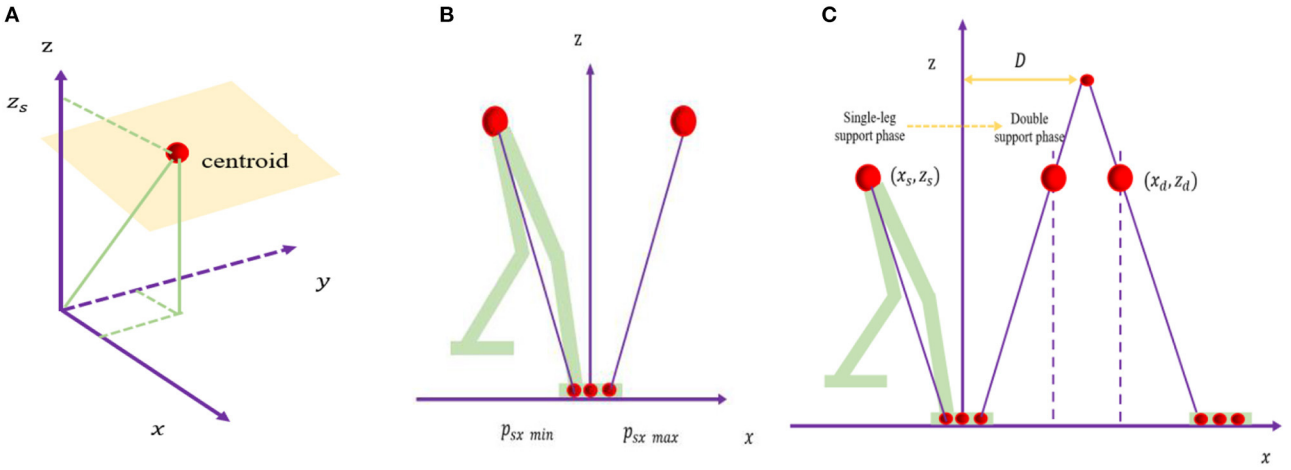


FIGURE 1 | (A) Three-dimensional linear inverted pendulum model. **(B)** Diagram of natural zero moment point. **(C)** Linear pendulum model.

of the mass in multistep planning can be expressed by the following equation:

$$x(t) = \sum_{i=0}^{n} (x_{si}(t) + x_{di}(t)) \tag{14}$$

$$\dot{x}(t) = \sum_{i=0}^{n} (\dot{x}_{si}(t) + \dot{x}_{di}(t)) \tag{15}$$

The robot gait realized by Equations (14, 15) yields a natural ZMP trajectory.

Swing Leg Trajectory Planning

Cosine functions and Bessel curves can all be used to plan swing-leg trajectories (Kajita et al., 2002; Grzelczyk et al., 2016). However, the humanoid robot is divided into two phases of single-leg support phase and double-leg support phase during walking, and only when the speed and acceleration of swing-leg start and landing are zero can the humanoid robot maintain stability when switching between single-leg support and double-leg support (Tang et al., 2003). To obtain a smoother trajectory of the oscillating leg, this paper chooses the method of simple harmonic motion synthesis:

$$z_{sw}(t) = \begin{cases} H_{sw}\left(\frac{2}{T_s}t - \frac{1}{2\pi}\sin\left(\frac{4\pi}{T_s}t\right)\right), t\epsilon[0, T_s/2] \\ H_{sw}\left(-\frac{2}{T_s}t + \frac{1}{2\pi}\sin\left(\frac{4\pi}{T_s}t\right) + 2\right), t\epsilon[T_s/2, T_s] \end{cases} \tag{16}$$

$$x_{sw}(t) = D_{sw}\left(\frac{t}{T_s} - \frac{1}{2\pi}\sin\left(\frac{2\pi}{T_s}t\right)\right) \tag{17}$$

where D_{sw} and H_{sw} are the maximum height of step length and leg lift, respectively.

Selection of Gait Parameters

There are two kinds of bipedal walking pattern generation methods. The first method uses precise knowledge of robot dynamics parameters, such as mass, centroid position, and inertia of each joint to configure walking mode. Therefore, the method mainly depends on the accuracy of the model data. In contrast, the second method uses limited knowledge of dynamics, such as the total position of the center of mass, the total angular momentum, etc. The simplified model method (3D-LIMP) used in this paper belongs to the second type. After the optimized trajectory is obtained, the robot can execute according to the corresponding trajectory, and an excellent walking gait can be obtained. By means of a gait generation method based on natural ZMP trajectory planning, humanoid robot walking is summarized in the following algorithmic steps.

Algorithm 1

1.Set the walking cycle T_s and walking parameters (D_{sw}, H_{sw}), initial foothold ($p_x^{(0)}$, $p_y^{(0)}$);
2.Initialization time $T = 0$, number of walking units $n = 0$;
3.Calculate the inverted pendulum equation from time T to $T + T_s$ and get the equation of mass center trajectory;
4. $T = T + T_s$, $n = n + 1$;
5. Calculate and determine the next foothold of the biped robot ($p_x^{(n)}$, $p_y^{(n)}$);
6. Give the next position of the bipedal robot;
7. Return to step 3.

The RoboCup3D server communicates with the robot once every 20 ms. In this article, the robot is also controlled once in 20 ms. One step is eight times, so the walk period T_s is 0.16 s. As algorithm 1 only considers the robot walking in a straight line, for the case of steering walk. If the step length is set to zero, the robot will only walk laterally, and if the step width is set to zero, the robot will only walk in a straight line. However, in a simulation match, the flexibility of the player is enhanced by changing the

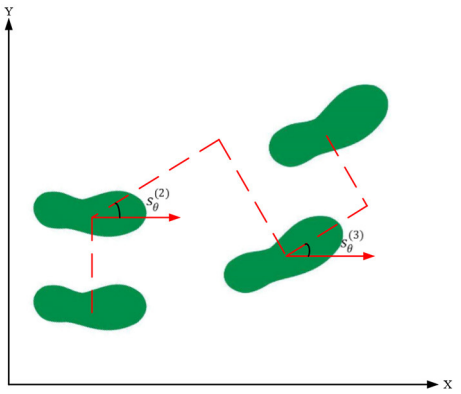


FIGURE 2 | Diagram of turning parameters.

TABLE 1 | Optimized gait parameters.

Parameter	Description
FootLength	Step size of one step (D_{sw})
FootWidth	Step width of one step
FootHeight	Maximum height of swinging leg (H_{sw})
s_θ	One step turning angle
T_s	One step time
BodyHeight	Body height in walking
Thigh	Leg length
Hip	Hip height
$Com_{off}(x,y)$	Deviation of centroid
Com_{max}	Offset of maximum center of mass
Com_{next}	Next centroid position
Arm(x,y)	Amplitude of arm swing
Zmp_{off}	Lateral offset of zero moment point

direction of travel while walking quickly. To be able to walk at any angle, additional information about the direction is required. The direction of each step is specified as s_θ, as shown in **Figure 2.**

Equation (18) represents the position of the nth step foothold of a humanoid robot $(p_x^{(n)}, p_y^{(n)})$

$$\begin{bmatrix} p_x^{(n)} \\ p_y^{(n)} \end{bmatrix} = \begin{bmatrix} p_x^{(n-1)} \\ p_y^{(n-1)} \end{bmatrix} + \begin{bmatrix} \cos s_\theta^{(n)} & -\sin s_\theta^{(n)} \\ \sin s_\theta^{(n)} & \cos s_\theta^{(n)} \end{bmatrix} \begin{bmatrix} \Delta x_x^{(n)} \\ -(-1)^n s_y^{(n)} \end{bmatrix} \quad (18)$$

Therefore, only important parameters such as step length, step width, and directional angle of humanoid robot need to be controlled to enable robot to walk. The gait parameters of the biped robot are as many as 40. If each parameter is optimized, it will inevitably affect the convergence speed because of the huge state space. And different gait parameters bias the focus of walking differently. According to natural ZMP-based mass-centered trajectory planning method and the previous experience of manual debugging, 13 key parameters affecting the gait of humanoid robot were selected, as shown in **Table 1**.

GAIT OPTIMIZATION BASED ON PARALLEL MULTIGROUP PARTICLE SWARM ALGORITHM

Aiming at the problem that a lot of manual debugging time is required when planning the robot trajectory by directly using a simplified model, this paper proposes a machine learning method based on the PCLPSO algorithm to optimize gait parameters. First, the important parameters are extracted according to the centroid trajectory planning method based on natural ZMP. Secondly, in the optimization of walking mode, different evaluation functions are set according to the requirements of game gait mode, and the PCLPSO algorithm is used to optimize the omnidirectional walking ability of a humanoid robot.

Parallel Comprehensive Learning Particle Swarm Optimizer Algorithm

PSO algorithm is a fast and efficient optimization algorithm (Kumar et al., 2018). Multiple individuals search for the target in the search area. It is a parallel and random optimization algorithm. Compared with other intelligent algorithms, it has a faster convergence speed and robustness. Each particle in PSO calculates the evaluation value based on the evaluation function. During the search of each particle, two extreme values are compared: the first is the optimal solution *pbest* of a particle; the other is the global optimal solution *gbest*. The speed and position updates of PSO are shown in Equations (19, 20):

$$v_i^{k+1} = v_i^k + c_1 * r_1 * (pbest_i^k - x_i^k) + c_2 * r_2 * (gbest^k - x_i^k) \quad (19)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \quad (20)$$

where $i = 1, \ldots, N$ is the number of populations and $k = 0, \ldots, N_{iter}$ is the number of iterations; $pbest_i^k$ represents the local optimal solution found by the particle itself, whereas $gbest^k$ represents the global optimal solution for all current particles; c_1 and c_2 are two constants greater than zero, which are used to adjust the degree of attraction of local and global optimal to the particle; r_1 and r_2 are both uniformly distributed random numbers in the interval [0,1], which affects the random nature of the algorithm. CLPSO is a valid variant of PSO (Liang et al., 2006). The main difference between CLPSO and PSO is that the original PSO requires the use of *pbest* and *gbest*, whereas for CLPSO, updating the location only requires *pbest*. For PSO, *pbest* only needs its own *pbest* in CLPSO update, which can come from other individuals. The speed update formula is shown as follows:

$$v_i^{k+1} = v_i^k + c * r * (pbest_{f_i(d)}^k - x_i^k) \quad (21)$$

where c is the acceleration factor. r is also uniformly distributed random numbers in the interval [0,1]. $\mathbf{f_i} = [f_i(1), f_i(2), \cdots, f_i(D)]$ is determined by the probability of population P_c of randomly selected particles i. $pbest_{f_i(d)}^k$ represents the pbest value of particle, which is stored in the list $\mathbf{f_i}$ of the particle i of the dth dimension. P_c is calculated, as

shown in Equation (22); 0.05 and 0.45 are the optimal values of hyperparameters set by Liang et al. (2006) based on experience.

$$P_c = 0.05 + 0.45 * \frac{e^{(\frac{10(i-1)}{ps-1})}}{e^{10} - 1} \tag{22}$$

where *ps* is the population size.

In PCLPSO, the particle population is divided into multiple groups, including a master group and several slave groups. All clusters run PCLPSO in the cluster environment at the same time. The *gbest*, *lbest*, and *pbest* in PCLPSO are defined as follows: *gbest* is the global optimal solution for all groups, *lbest* is the local optimal solution for populations, and *pbest* is the optimal solution for particles. Each slave group sends its *lbest* to the master group. The master group chooses the best solution from all *lbest* as *gbest* and sends *gbest* to all slaves. Each slave group randomly selects a particle to receive *gbest* to update its own *pbest*. PCLPSO algorithm updates its own *pbest* by a parallel distributed collaborative strategies. They are improving the quality of the solution and the rate of solving. Each particle in the PSO algorithm is updated by updating *pbest* and *gbest* values. The *gbest* affects the direction of population, and when it falls into a local minimum, the swarm particles tend to fall into this local minimum. PCLPSO adopts a comprehensive learning strategy, and the speed and position of the updated particles depend on all other particles. The *lbest* is selected from the *pbest* of the slave group, and the main group selects the optimal solution *gbest* from all the *lbest* of the slave group to ensure that it will not fall into a local minimum.

Construction of a Parallel Distributed Training Environment

The particles in the group in the PCLPSO algorithm are all running on a single computer, but each football robot in a RoboCup3D match is controlled by a separate client. Therefore, the PCLPSO algorithm cannot be directly applied to the RoboCup3D simulation platform. Because all clients are connected to the RoboCup3D simulated football server, the client and server can run on multiple computers in a distributed environment. Therefore, the PCLPSO algorithm is decomposed into three algorithms by changing the cluster structure to accommodate the RoboCup3D simulation platform in this paper.

In the new cluster, the structure is still based on the master–slave model of parallel computing; only one is a master cluster; the others are slaves, as shown in **Figure 3**. The gait training task of the humanoid robot is decomposed into multiple processes in the slave cluster, and a distributed training system is built to run on multiple computers. The parallel and distributed optimization framework can reduce the scale of solving gait problems. When all slaves are started and connected to the master group, and the master group sends parameters to the slave group, the communication cycle begins. Communication takes place only between the master group and slave groups; there is no communication between slave groups. In the cluster structure of this paper, the master node has no group, and its main function is to send initial parameters to the group of slave nodes (Algorithm 2, line 6), and the slave group sends its own *lbest* to the master group (Algorithm 2, line 8). The master group collects the received *lbest,* including its own *lbest,* into a pool called Elite Pool (EP). The master group finds *gbest* from the EP and eventually sends *gbest* to the slave group (Algorithm 2, lines 9, 10, and 11). The detailed algorithm of the master is shown in Algorithm 2. The slave group exchange algorithm is shown in Algorithm 3. Add an exchange program to slave group clients for exchanging data between the master client and all the clients of the slave group. In this paper, Local Pool (LP) is added to the slave group to collect all *pbest* in the group. The slave group finds *lbest* among all *pbest* and sends its own *lbest* to the master group. The master server adds them to EP, finds *gbest*, and shares it with the slave server. A robot is a member of a group during gait optimization training, a slave group has seven clients, and a client controls a humanoid robot, and eventually, multiple humanoid robots are trained in a RoboCup3D simulation court. The group client is used to calculate ZMP trajectory, COM trajectory, and swing leg trajectory when a humanoid robot is trained by the PCLPSO algorithm to walk. This part also calculates joint angle information and adaptation values for a cycle of humanoid robot walking. The detailed algorithm from the group client is shown in Algorithm 4. The gait optimization process for the client-based PCLPSO algorithm is shown in **Figure 4**.

Algorithm 2 Master algorithm

1. int *n*, *k*, *P*, *m*
2. double *w*, *c1*, *c2*
3. Array EP
4. Initialize the parameters *n*, *k*, *P*, *m*, *w*, *c1*, *c2*
5. Wait until all slave swarms have been launched and connected to this Master
6. The master sends the parameters to the slaves
7. **repeat**
8. Wait until all slave swarms have sent *lBest* to the master
9. The master stores the *lBests* into the EP
10. The master finds the *gBest* in the EP and empties the EP
11. The master sends the *gBest* to the slaves
16. **until** the stopping criterion is met
17. **return**

Algorithm 3 Switching algorithm of a slave swarm

1. int *n*, *k*, *P*, *m*
2. double *w*, *c1*, *c2*
3. Array LP
4. Connect to the Master until the parameters are received from the master
5. int $d = 0$
6. **repeat**
7. *d*++
8. Store the *pBests* into the LP until all *pBests* in this slave swarm are received
9. **if** *d* mod *P*==0 **then**
10. Find the *lBest* in the LP
11. Send the *lBest* to the master
12. Wait until the *gBest* is received from the master
13. Randomly update a *lBest* with the *gBest* in the LP
14. **end if**
15. Send the LP to the all clients in this swarm
16. **until** the stopping criterion is met
17. **return**

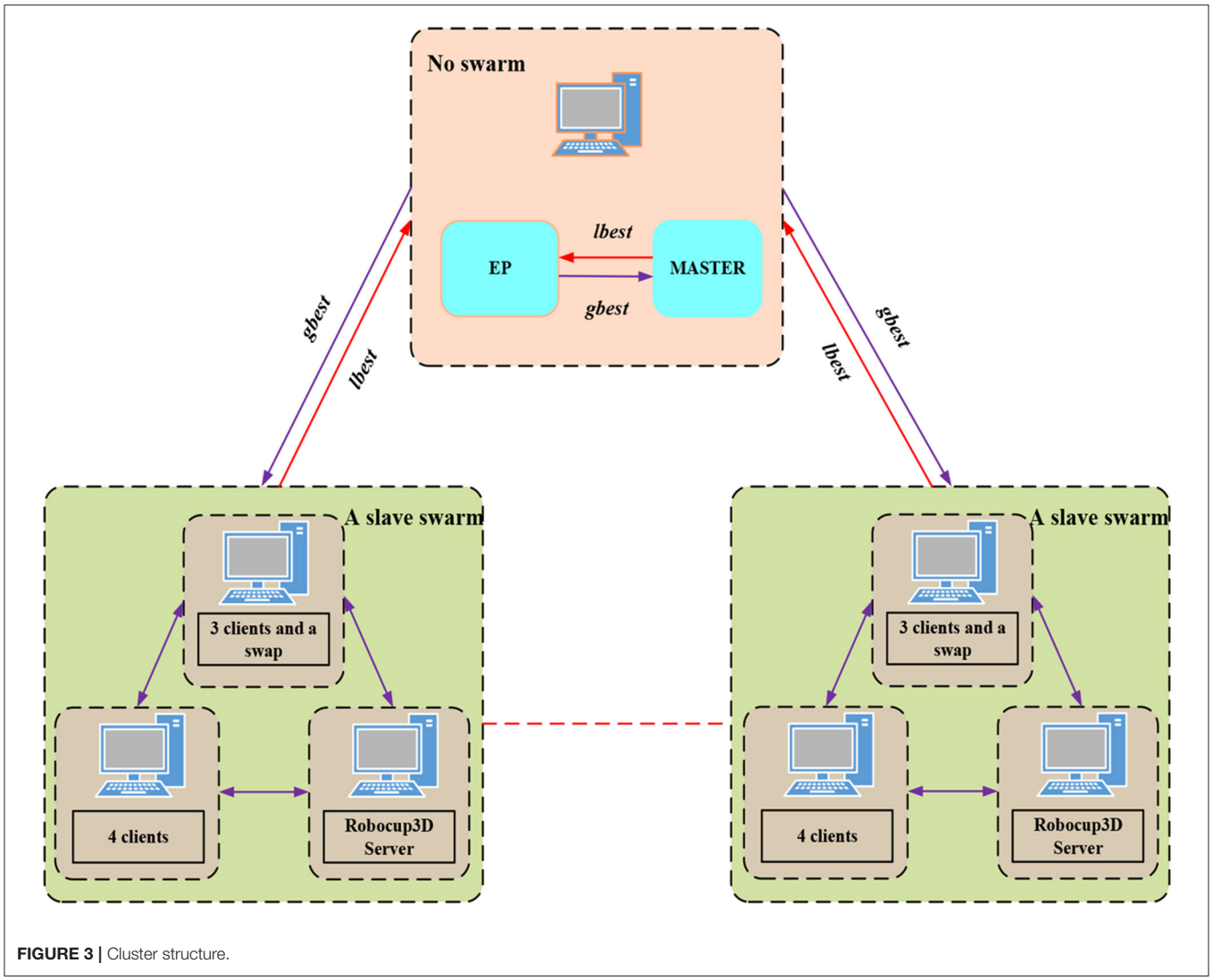


FIGURE 3 | Cluster structure.

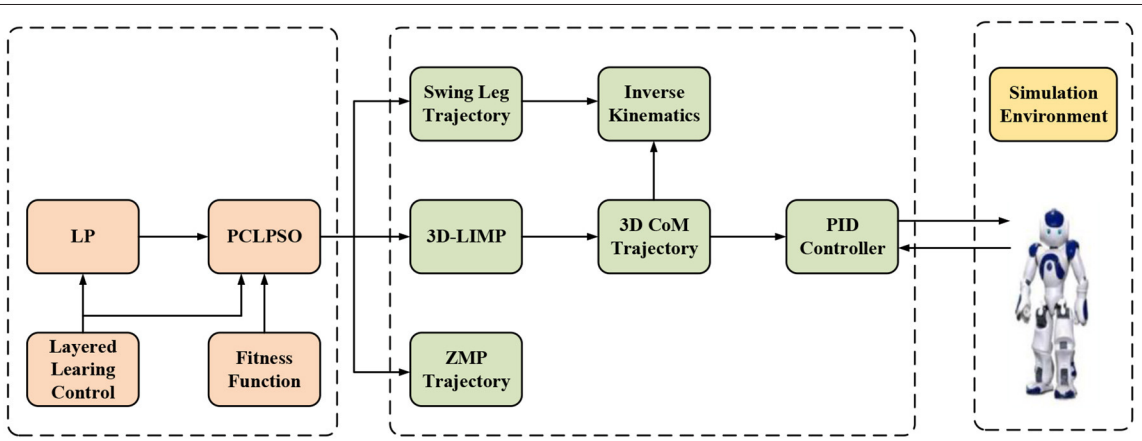


FIGURE 4 | Flowchart of PCLPSO-based gait optimization.

Algorithm 4 Algorithm in a client of a slave swarm

1. int *n*, *k*, *P*, *m*
2. double *w*, *c1*, *c2*
3. Array LP
4. Connect to the swap program in a swarm
5. Wait until the parameters are received from swap program in a swarm
6. Initialize the parameters and velocity
7. Calculate Swing leg Trajectory and all joint angles of Robot NAO in a walk cycle
8. Humanoid robot performs a walking training and calculates the fitness value
9. Update *pBest*
10. Send the *pBest* to the swap program until the LP is received from the swap program
11. Update the LP and find the *lBest* in the LP
12. **repeat**
13. Update the velocity and position
14. Calculate Swing leg Trajectory and all joint angles of Robot NAO in a walk cycle
15. Humanoid robot performs a walking training and calculates the fitness value
16. Update *pBest*
17. Send the *pBest* to the swap program until the LP is received from the swap program
18. Update the Local Pool and find the *lBest* in the LP
19. **until** the stopping criterion is met
20. **return**

Design of Evaluation Functions

The design of evaluation criteria is critical to achieving an excellent result (MacAlpine and Stone, 2018). In this paper, gait optimization results are judged based on the following criteria:

1. The distance the humanoid robot travels per unit time.

$$f_{dis} = a_1 * \|\varphi_{end} - \varphi_{start}\| \tag{23}$$

where φ_{start} and φ_{end} are the coordinates of start and end of the training, respectively, and a_1 is the weight.

2. Whether zero force matrix is always within the supported polygon and robot walks without falling all the time. in this paper, the zmp coordinates of the walking action sequence are chosen as the basis for calculation.

$$f_{zmp} = a_2 * \sum_{k=1}^{N} \sqrt{(P_x(k))^2 + (P_y(k))^2} \tag{24}$$

where $P_x(k)$ and $P_y(k)$ are the ZMP coordinates for each gait action sequence, and a_2 is the weight.

3. Humanoid robot in the fast walking process torso always maintains stability; body torso does not shake. In a simulation match, the body of players is in frequent contact with each other, and it is very easy to be knocked down by the opponent if the torso is not stable. At the initial time of double-support phase, the expected com coordinates are 2 ft on the center. In this paper, torso sway is detected by comparing the two coordinates of com and the center of feet during the initial stage of the dual support phase.

$$x_f = c_x - \frac{x_{foot}R + x_{foot}L}{2} \tag{25}$$

$$f_{shake} = \begin{cases} 0 & f_{abs}(x_f) < th_\theta \\ c & otherwise \end{cases} \tag{26}$$

where $x_{foot}R$ and $x_{foot}L$ are the coordinates of initial time 2 ft of the double support phase, th_θ is the set threshold, and f_{shake} and c are penalty and normal numbers, respectively.

If the robot falls during training, a constant $f_{falling}$ will be given as a penalty value. The evaluation function for speed and stability training is shown in Equation (27).

$$F_1 = f_{dis} - f_{zmp} - f_{falling} - f_{shake} \tag{27}$$

Add the lateral walking and the turning angles of the humanoid robot to the above, and the maximum distance of lateral walking for each step is limited to 0.04 m, and the maximum turning angle is 15°. In the optimization process, two different target points are set for the bipedal walking robot to train its gait. If the target point is reached, then the training of the next target point is quickly stopped. The evaluation function is as follows:

$$F_2 = f_{dis} - f_{falling} - f_{punish} + f_{reward} \tag{28}$$

where f_{punish} is the penalty for not completing the task within the specified time, and f_{reward} is the reward for completing the task.

Layered Learning

In the RoboCup3D simulation game, players can be roughly divided into two categories. The first category is to hold the ball, avoid the defense of the opponent, intercept the ball in the shortest time, and send the ball to the goal of the opponent; the second category is to run according to the situation on the field players and walk to the designated location as soon as possible. Players without the ball often complete tasks such as running and intercepting opposing players with the ball up and down. For players to complete the tasks assigned by their superiors in the shortest possible time, the walking speed of robots is very important. The player is the defensive object of an opponent after getting the ball, so frequent collisions and turns are inevitable. Improve the stability of the robot and steering ability to avoid falling and taking advantage of collisions. The omnidirectional walking of a biped robot can be decomposed into forward walking and steering motion. This paper uses a layered learning method to optimize each decomposition action of omnidirectional walking, and the final omnidirectional walking optimization is shown in **Figure 5**. Each sub-module requires the optimization algorithm to be trained through the corresponding evaluation function. Use manually adjusted parameters to drive the robot to walk, and then learn to get a fast mode with speed and stability as the goal. The final optimized parameters of the fast and stable mode are used as the initial state of the steering mode, and the learning is continued with the goal of steering stability. Through hierarchical learning of the humanoid robot, two different walking parameter sets (straight and turning) can be obtained. When different walking tasks are to be completed, the parameter sets can be switched at any time, and the flexible connection transition during switching is also obtained through learning.

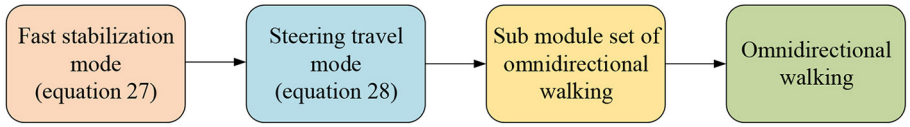


FIGURE 5 | Layered learning for omnidirectional walking process.

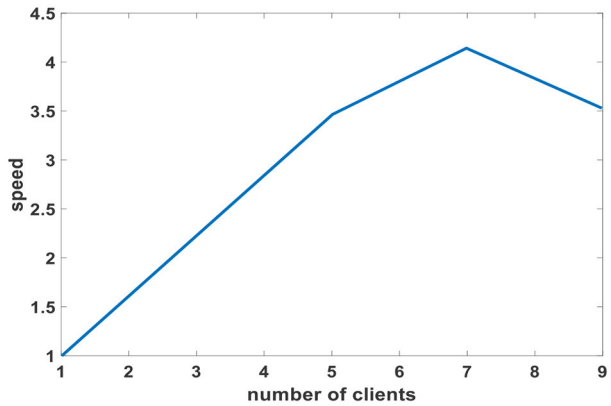


FIGURE 6 | Number of clients from the group and training speed.

FIGURE 7 | Training scenario diagram.

EXPERIMENTAL RESULTS AND ANALYSIS

This article uses a 3D-LIMP model to generate gait and opens the function of adjusting walking engine parameters to the public. OpenAI Gym acts as a bridge between optimization algorithm and environment and accepts the gait parameters computed by an optimization algorithm, performs a training session in the environment, and then provides the necessary information back to the optimization algorithm. To verify the effectiveness of the PCLPSO algorithm for gait optimization, this paper compares PCLPSO with three commonly used optimization algorithms, GA, CFO, and CMA-ES (Rongyi and Chunguang, 2018), in terms of the speed, stability, and turn capability of gait. The angle of view of players is only−120–120°. To get accurate data during training, the value of setViewCones in the server is changed from 120 to 360°. It is also necessary to change the game mode to fast mode during training so that the training is not limited by time. The mathematical properties of the four optimization algorithms of GA, CFO, CMA-ES, and PCLPSO are evolutionary algorithms. Each training uses the same population size (10) and the same number of variables (13). First, randomly produce 10 sets of parameters. The humanoid robot walks according to the 10 sets of parameters. The walking speed and stability will be different. Choose the optimal one, and then iterate the formula. Generate the next generation, and repeat the execution with 10 iterations. In this article, the coding method of the GA algorithm is standard binary coding. The number of parameters determines the length of a chromosome. When calculating the evaluation function value, the binary chromosome string should be decomposed and decoded to get the real number parameters. The specific parameters of the GA algorithm are as follows: take population size $N_p = 40$, evolutionary algebra $T = 200$, and crossover probability $P_c = 0.7$, and variation probability $P_m = 0.05$. The specific parameters of the CFO algorithm are as follows: $\alpha = 0.3, \beta = 0.3, \gamma_{start} = 0$, $\gamma_{stop} = 1$, and $\gamma = 0.1$.

Firstly, Equation (27) is chosen as an evaluation function to optimize the speed and stability of the robot. The relationship between the number of clients in a single slave group and training speed is shown in **Figure 6**. The training speed of multiple clients is compared to a single client when it takes a single client to train to an optimal value 1. The training speed increases linearly between 1 and 7 clients. The training speed reaches its maximum when the slave group contains seven clients. So, in the following comparison experiment, the slave group structures of PCLPSO are trained by seven clients controlling seven humanoid robots. The training scenario of a humanoid robot in the RoboCup3D scenario is shown in **Figure 7**.

Figure 8A shows the current optimal fitness values of four algorithms, all of which increase with the increase in the number of training. The evaluation function of the PCLPSO algorithm reaches the optimal value after 6,500 trainings, and the optimal value is 5.86. The evaluation function of the CMA-ES algorithm reaches the optimal after 14,000 trainings, and the optimal value is 4.73. GA algorithm reaches the optimal evaluation function after 17,500 trainings, and the optimal value is 3.92. The evaluation function of the CFO algorithm reaches the optimal after 16,000 trainings, and the optimal value is 4.2. The evaluation function value is averaged every 50 times of training, as shown in **Figure 8B**. As the number of iterations increases,

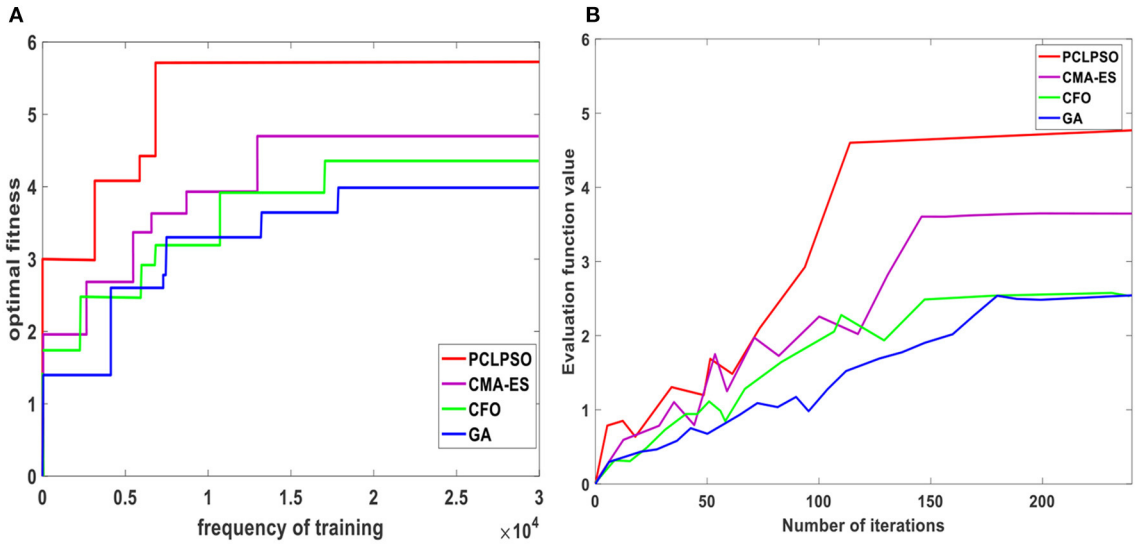


FIGURE 8 | Comparison of evaluation function values of four algorithms. **(A)** Current optimal evaluation function value. **(B)** Average function value.

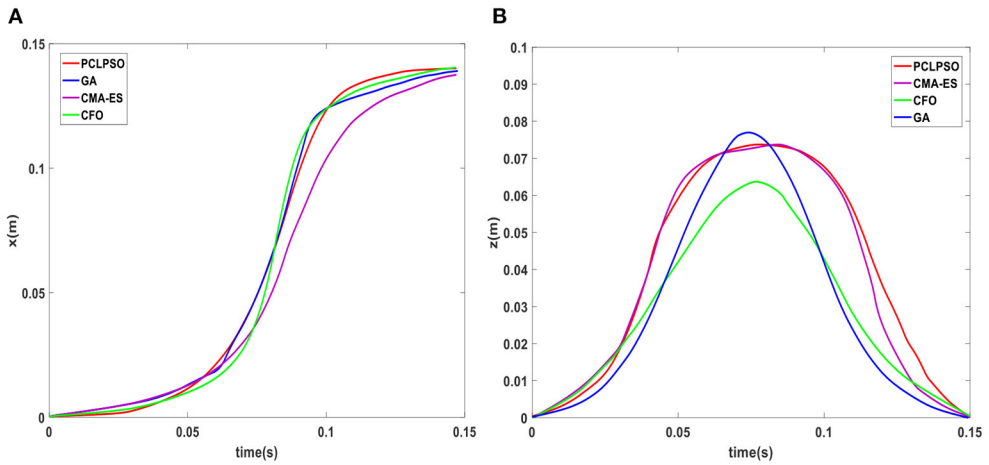


FIGURE 9 | (A) x-axis trajectory. **(B)** z-axis trajectory.

the evaluation function of all four algorithms increases steadily. One hundred twelve iterations for PCLPSO, 146 iterations for CMA-ES, 176 iterations for GA, and 148 iterations for CFO are stable. When the parallel distributed optimization framework interacts with RoboCup3D, many low-cost data samples can be obtained. Decomposition of gait problems can reduce the scale of problem-solving. Make sure to get a better solution faster.

The walking stability measured by Equation (25) shows that the x_f fluctuation range of the PCLPSO algorithm is−0.2 to 0.2 mm. The CMA-ES algorithm fluctuates between - 0.3 and 0.5. The x_f range of GA is−0.7 to 1.4 mm, and that of the CFO algorithm is−1.3 to 0.8 mm. The PCLPSO algorithm has the smallest x_f fluctuation range, and the humanoid robot is more stable. The optimized trajectories of swing leg x-axis and z-axis of four algorithms are shown in **Figures 9A,B**. At the moment of takeoff and landing, the optimized trajectory of the swing leg of the PCLPSO algorithm is parallel to the ground, which ensures stability when switching between the single-support phase and double-support phase.

The real-time changes of hip deflection pitch joint angle, hip transverse roll joint angle, and hip pitch joint angle for four algorithms are shown in **Figure 10**. The hip angle changes of the PCLPSO algorithm are very stable and better than those of the CMA-ES, CFO, and GA algorithms.

A comparison of changes in the center of mass landing points for four methods is shown in **Figure 11A**. The landing point of the center of mass in the double-leg support phase of the robot using the parameters optimized by the PCLPSO algorithm is always on the center of two-legged linkage and remains stable, whereas the landing point of the robot center of mass using

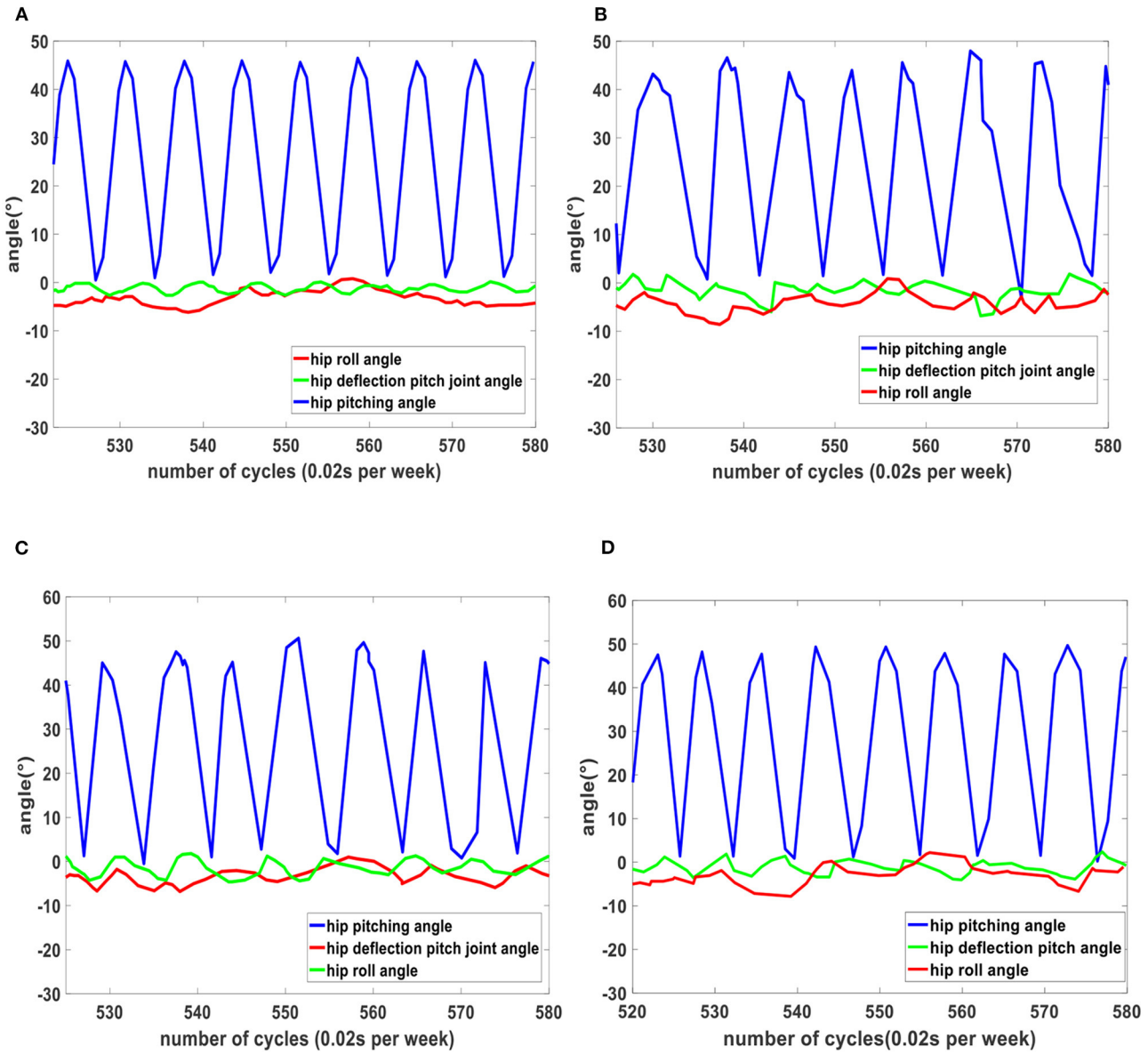


FIGURE 10 | Changes in hip angle for four algorithms. **(A)** Hip angle variation of PCLPSO algorithm. **(B)** Hip angle variation of CFO algorithm. **(C)** Hip angle variation of GA algorithm. **(D)** Hip angle variation of CMA-ES algorithm.

the parameters optimized by CFO and GA is unstable. CMA-ES algorithm optimized humanoid robot mass center fallout is more stable than the CFO and GA algorithms but inferior to the PCLPSO algorithm. ZMP trajectory after the optimization of three algorithms is shown in **Figure 11B**. The ZMP trajectory of the CMA-ES, GA, and CFO algorithms is close to the edge of the support polygon, whereas the whole trajectory curve of the PCLPSO algorithm moves toward the middle of the support polygon, in which case the stability margin of humanoid robot ZMP point is larger.

After the training mentioned earlier, it can be seen in humanoid robot gait optimization, and PCLPSO algorithm optimization of humanoid robot already has a fast and stable gait, but in simulation competition in a humanoid robot, the target point is constantly changing. When changing from a linear to a rotating state, the average speed decreases again when the rotation angle is small, and the humanoid robot is extremely unstable during rapid stops. In this paper, to better adapt to dynamic walking, layer learning is used to further optimize gait. The robot not only has a fast and stable gait but also has excellent steering ability. Next, the steering ability of the humanoid robot is optimized using Equation (28) as an evaluation function. Test experiments on the turning ability of humanoid robots were conducted under the SimSpark platform of RoboCup3D, and the

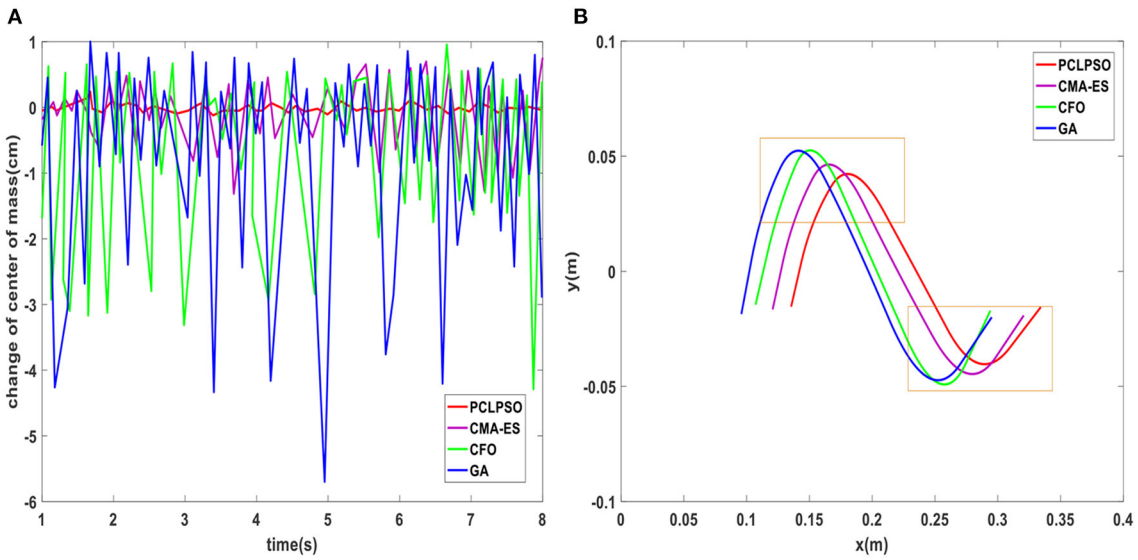


FIGURE 11 | (A) Humanoid robot center of mass landing point p_{shake} variation. **(B)** ZMP trajectory after optimization of four algorithms.

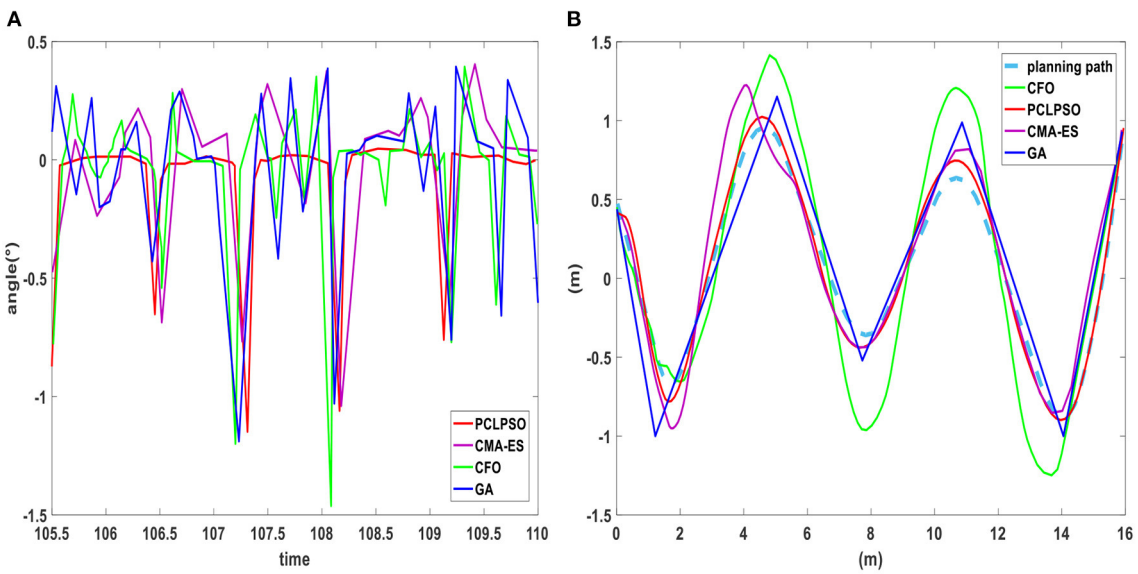


FIGURE 12 | (A) Body rotation angle. **(B)** Turning performance test.

walking path was recorded by Matlab using RoboViz observation. As shown in **Figure 12A**, the body rotation of four algorithms when the robot is optimized to make it turn continuously. The body rotation angle of robots using the PCLPSO algorithm with optimized parameters reaches a maximum of−1.19° during support leg switching, which shows that the robot is very stable during the switching process of walking and turning. Preplanning the walking path of the robot, the trajectory after layer learning using three algorithms is shown in **Figure 12B**. The GA algorithm walks a trajectory that cannot be flexible enough to maintain stability when turning. The humanoid robot optimized with the CMA-ES and CFO algorithms can turn flexibly but requires some adjustment time when turning, and the PCLPSO algorithm walks on a smooth trajectory that is almost identical to the planned path.

Let humanoid robot go straight for 15 m, test each algorithm 100 times, and take the average value. The results are shown in **Table 2**. The PCLPSO algorithm is used to walk with less time and faster speed under the same walking distance. The steering angle represents the angle between the body orientation of the initial position of the humanoid robot and the target point. The four algorithms are tested 100 times when the steering angle is

TABLE 2 | Straight speed comparison.

Algorithm	Walking distance (m)	Average time (s)	Average speed (m/s)
PCLPSO	15	16.83	0.89
CMA-ES	15	21.02	0.71
CFO	15	21.05	0.71
GA	15	25.03	0.60

TABLE 3 | Steering performance test.

Algorithm / time (s) / Angle(°)	PCLPSO	CMA-ES	CFO	GA
30°	21.07	21.32	21.37	22.47
45°	21.45	22.01	22.38	22.58
60°	21.75	22.45	22.67	23.07
90°	22.40	22.73	22.85	23.39

30, 45, 60, and 90°. The average time to reach the target point of each algorithm is shown in **Table 3**. The PCLPSO algorithm has the shortest average time to reach the target point at four angles.

CONCLUSION

This paper has built a RoboCup3D parallel distributed multi-robot gait training environment based on PCLPSO. A layer learning approach was used to optimize the existing problems in layers. It effectively reduces the influence of similar noise of a single simulation platform and has a faster optimization speed than common optimization algorithms. The final experimental results show that the PCLPSO algorithm optimizes faster and walks more quickly and steadily. During turning motion, the PCLPSO algorithm walks with a smoother trajectory and a smaller body rotation angle when switching between straight motion and turning motion, enabling stable, and flexible turning of a humanoid robot. This algorithm can also be extended to other aspects of RoboCup3D, such as the optimization of basic movements of humanoid robots such as goal shooting.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary materials, further inquiries can be directed to the corresponding author/s.

AUTHOR CONTRIBUTIONS

CT: methodology. JX: software. ZZ: funding acquisition, project administration, and supervision. FC and HG: investigation. CL: validation. All authors contributed to the article and approved the submitted version.

FUNDING

This work was supported in part by the National Natural Science Foundation of China (Grants No. 61801323), by the Science and Technology Projects Fund of Suzhou (Grant No. SYG201708, Grant No. SS2019029), by the Construction System Science and Technology Fund of Jiangsu Province (Grant No. 2017ZD066), by the Jiangsu Collaborative Innovation Center for Cultural Creativity (Grant No. XYN1703) and by the Natural Science Foundation of Changzhou Institute of Technology (Grant No. YN1603).

REFERENCES

Astudillo, D., Minchala, L. I., Astudillo-Salinas, F., Vazquez-Rodas, A., and Gonzalez, L. (2018). A simple mapping methodology of gait biomechanics for walking control of a biped robot. In: *2018 IEEE XXV International Conference on Electronics, Electrical Engineering and Computing (INTERCON)* (Lima: IEEE), 1-4. doi: 10.1109/INTERCON.2018.8526395

Bai, L., Hu, H., Chen, X., Sun, Y., Ma, C., and Zhong, Y. (2019). Cpg-based gait generation of the curved-leg hexapod robot with smooth gait transition. *Sensors* 19:3705. doi: 10.3390/s19173705

Baoping, W., Yuqian, D., Xiaoying, S., and Jiayi, L. (2015). Study of humanoid robot gait based on human walking captured data. In: *2015 34th Chinese Control Conference (CCC)* (Hangzhou: IEEE), 4480–4485. doi: 10.1109/ChiCC.2015.7260332

Bonyadi, M. R., and Michalewicz, Z. (2017). Particle swarm optimization for single objective continuous space problems: a review. *Evol. Comp.* 25, 1–54. doi: 10.1162/EVCO_r_00180

Elhosseini, M. A., Haikal, A. Y., Badawy, M., and Khashan, N. (2019). Biped robot stability based on an a-c parametric whale optimization algorithm. *J. Comp. Sci.* 31, 17–32. doi: 10.1016/j.jocs.2018.12.005

Faraji, S., Razavi, H., and Ijspeert, A. J. (2019). Bipedal walking and push recovery with a stepping strategy based on time-projection control. *Int. J. Rob. Res.* 38, 587–611. doi: 10.1177/0278364919835606

Fayong, G., Jianghai, Z., Tao, M., Minzhou, L., and Xiaobo, S. (2014). A modified gait generator for humanoid robots based on height compensation of center of mass. In: *2014 IEEE InternationalConference on Robotics and Biomimetics (ROBIO 2014)* (Bali: IEEE), 1278–1283. doi: 10.1109/ROBIO.2014.7090509

Graf, C., and Röfer, T. (2011). A center of mass observing 3d-lipm gait for the robocup standard platform league humanoid. In: *Robot Soccer World Cup* (Berlin; Heidelberg: Springer), 102–113. doi: 10.1007/978-3-642-32060-6_9

Grzelczyk, D., Stańczyk, B., and Awrejcewicz, J. (2016). Prototype, control system architecture and controlling of the hexapod legs with nonlinear stick-slip vibrations. *Mechatronics* 37, 63–78. doi: 10.1016/j.mechatronics.2016.01.003

Gülcü, S., and Kodaz, H. (2015). A novel parallel multi-swarm algorithm based on comprehensive learning particle swarm optimization. *Eng. Appl. Art. Intell.* 45, 33–45. doi: 10.1016/j.engappai.2015.06.013

Hereid, A., Hubicki, C. M., Cousineau, E. A., and Ames, A. D. (2018). Dynamic humanoid locomotion: a scalable formulation for hzd gait optimization. *IEEE Trans Rob.* 34, 370–387. doi: 10.1109/TRO.2017.2783371

Hong, Y.-D., and Lee, K.-B. (2016). Stable walking of humanoid robots using vertical center of mass and foot motions by an evolutionary optimized central pattern generator. *Int. J. Adv. Rob. Syst.* 13:27. doi: 10.5772/62039

Huan, T. T., and Anh, H. P. H. (2015). Novel stable walking for humanoid robot using particle swarm optimization algorithm. In: *2015 International Conference on Artificial Intelligence and Industrial Engineering* (Phuket: Atlantis Press). doi: 10.2991/aiie-15.2015.90

Huan, T. T., Thy, K. B., Trung, N. H. H., and Anh, H. P. H. (2018a). Stable gait optimization for small-sized humanoid robot using cfo. In: *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*. doi: 10.1109/ICARCV.2018.8581188

Huan, T. T., Van Kien, C., Anh, H. P. H., and Nam, N. T. (2018b). Adaptive gait generation forhumanoid robot using evolutionary neural model optimized with modified differential evolution technique. *Neurocomputing* 320, 112–120. doi: 10.1016/j.neucom.2018.08.074

Jadidi, M. G., and Hashemi, E. (2016). Optimal preview control of the nao biped robot using a ukf-based state observer. In: *2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)* (Banff, AB: IEEE), 52-57. doi: 10.1109/AIM.2016.7576742

Kajita, S., Nagasaki, T., Yokoi, K., Kaneko, K., and Tanie, K. (2002). Running pattern generation for a humanoid robot. In: *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)* (Washington, DC: IEEE), 2755–2761.

Kennedy, J., and Eberhart, R. (1995). Particle swarm optimization. In: *Proceedings of ICNN'95-International Conference on Neural Networks (IEEE)*, 1942–1948. doi: 10.1109/ICNN.1995.488968

Kumar, P., Mukherjee, S., Saini, R., Kaushik, P., Roy, P. P., and Dogra, D. P. (2018). Multimodal n gait recognition with inertial sensor data and video using evolutionary algorithm. *IEEE Trans. Fuzzy Syst.* 27, 956–965. doi: 10.1109/TFUZZ.2018.2870590

Liang, J. J., Qin, A. K., and Suganthan, P. N. (2006). Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Trans. Evol. Comp.* 2005:281–295. doi: 10.1109/TEVC.2005.857610

MacAlpine, P., Depinet,M., and Stone, P. (2015). "Ut austin villa 2014: Robocup 3d simulation league champion via overlapping layered learning," in *Twenty-Ninth AAAI Conference on Artificial Intelligence*. doi: 10.1007/978-3-642-39250-4_8

MacAlpine, P., and Stone, P. (2018). Overlapping layered learning. *Art. Intell.* 254, 21–43. doi: 10.1016/j.artint.2017.09.001

Mandava, R. K., and Vundavilli, P. R. (2018). Near optimal pid controllers for the biped robot while walking on uneven terrains. *Int. J. Autom. Comp.* 15, 689–706. doi: 10.1007/s11633-018-1121-3

Muniz, F., Maximo, M. R., and Ribeiro, C. H. (2016). Keyframe movement optimization for simulated humanoid robot using a parallel optimization framework. In: *2016 XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR)* (Recife: IEEE), 79-84. doi: 10.1109/LARS-SBR.2016.20

Paparisabet, M., Dehghani, R., and Ahmadi, A. (2019). Knee and torso kinematics in generation of optimum gait pattern based on human-like motion for a seven-link biped robot. *Multi. Syst. Dynam.* 47, 117–136. doi: 10.1007/s11044-019-09679-z

Rongyi, H., and Chunguang, L. (2018). Gait optimization of robocup3d simulation robot. *Comp. Modern.* 3, 20–25 doi: 10.3969/j.issn.1006-2475.2018.03.004

Sato, T., Sakaino, S., and Ohnishi, K. (2010). Real-time walking trajectory generation method with three-mass models at constant body height for three-dimensional biped robots. *IEEE Trans. Indust. Electr.* 58, 376–383. doi: 10.1109/TIE.2010.2052535

Tang, Z., Zhou, C., and Sun, Z. (2003). Trajectory planning for smooth transition of a biped robot. In: *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)* (Taipei: IEEE), 2455–2460.

Wang, M., Dong, H., Li, X., Zhang, Y., and Yu, J. (2019). Control and optimization of a bionic robotic fish through a combination of cpg model and pso. *Neurocomputing* 337, 144–152. doi: 10.1016/j.neucom.2019.01.062

Weon, I.-S., and Lee, S.-G. (2018). Intelligent robotic walker with actively controlled human interaction. *ETRI J.* 40, 522–530. doi: 10.4218/etrij.2017-0329

Winkler, A. W., Bellicoso, C. D., Hutter, M., and Buchli, J. (2018). Gait and trajectory optimization for legged systems through phase-based end-effector parameterization. *IEEE Rob. Autom. Lett.* 3, 1560–1567. doi: 10.1109/LRA.2018.2798285

Zhong, G., Chen, L., Jiao, Z., Li, J., and Deng, H. (2017). Locomotion control and gait planning of a novel hexapod robot using biomimetic neurons. *IEEE Trans. Control Syst. Technol.* 26, 624–636. doi: 10.1109/TCST.2017.2692727

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

frontiers in Neurorobotics

ORIGINAL RESEARCH
published: 02 February 2021
doi: 10.3389/fnbot.2020.610139

Intention Understanding in Human–Robot Interaction Based on Visual-NLP Semantics

Zhihao Li[1†], *Yishan Mu*[2†], *Zhenglong Sun*[3†], *Sifan Song*[4], *Jionglong Su*[5] *and Jiaming Zhang*[1,6*]

[1] Institute of Robotics and Intelligent Manufacturing, The Chinese University of Hong Kong, Shenzhen, China, [2] School of Statistics, Southwestern University of Finance and Economics, Chengdu, China, [3] School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, China, [4] Department of Mathematical Sciences, Xi'an Jiaotong-Liverpool University, Suzhou, China, [5] School of AI and Advanced Computing, XJTLU Entrepreneur College (Taicang), Xi'an Jiaotong-Liverpool University, Suzhou, China, [6] Research Center on Special Robots, Shenzhen Institute of Artificial Intelligence and Robotics for Society, Shenzhen, China

With the rapid development of robotic and AI technology in recent years, human–robot interaction has made great advancement, making practical social impact. Verbal commands are one of the most direct and frequently used means for human–robot interaction. Currently, such technology can enable robots to execute pre-defined tasks based on simple and direct and explicit language instructions, e.g., certain keywords must be used and detected. However, that is not the natural way for human to communicate. In this paper, we propose a novel task-based framework to enable the robot to comprehend human intentions using visual semantics information, such that the robot is able to satisfy human intentions based on natural language instructions (total three types, namely clear, vague, and feeling, are defined and tested). The proposed framework includes a language semantics module to extract the keywords despite the explicitly of the command instruction, a visual object recognition module to identify the objects in front of the robot, and a similarity computation algorithm to infer the intention based on the given task. The task is then translated into the commands for the robot accordingly. Experiments are performed and validated on a humanoid robot with a defined task: to pick the desired item out of multiple objects on the table, and hand over to one desired user out of multiple human participants. The results show that our algorithm can interact with different types of instructions, even with unseen sentence structures.

Keywords: human–robot interaction, intention estimation, scene understanding, visual-NLP, semantics

OPEN ACCESS

Edited by:
Zheng Wang,
Southern University of Science and Technology, China

Reviewed by:
Jing Guo,
Guangdong University of Technology, China
Bin Fang,
Tsinghua University, China
Yuquan Wang,
UMR5506 Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM), France

***Correspondence:**
Jiaming Zhang
zhangjiaming@cuhk.edu.cn

[†] *These authors share first authorship*

Received: *25 September 2020*
Accepted: *18 December 2020*
Published: *02 February 2021*

Citation:
Li Z, Mu Y, Sun Z, Song S, Su J and Zhang J (2021) Intention Understanding in Human–Robot Interaction Based on Visual-NLP Semantics. Front. Neurorobot. 14:610139. doi: 10.3389/fnbot.2020.610139

1. INTRODUCTION

In recent years, significant progress has been achieved in robotics in which human–computer interaction technology plays a pivotal role in providing optimal user experience, reduces tedious operations, and increases the degree of acceptance of the robot. Novel human–computer interaction techniques are required to further advance the development in robotics, with notably the most significant one being a more natural and flexible interaction method (Fang et al., 2018, 2019; Hatori et al., 2018). It requires robots to process external information as a human in many application scenarios. For home service robots, visual and auditory information is the most direct way for people to interact and communicate with them. With continual advancement in statistical

modeling, speech recognition has been widely adopted in robots and smart devices (Reddy and Raj, 1976) to realize natural language-based human–computer interaction. Furthermore, substantial development in the field of image perception has been carried out, even achieving human-level performance in some tasks (Hou et al., 2020; Uzkent et al., 2020; Xie et al., 2020). By fusing visual and auditory information, robots are able to understand human natural language instructions and carry out required tasks.

There are several existing home service robots that assist humans in picking up specific objects based on natural language instructions. (Kollar et al., 2010) proposed to solve this problem by matching nouns and the target objects. Eppe et al. (2016) focuses on parsing natural language instructions by Embodied Construction Grammar (ECG) analyzer. Paul et al. (2018) utilizes probabilistic graph models for natural language comprehension, but objects are required to be described in advance through natural language. With the development of neural networks, some researchers tried to tackle the problem of natural language comprehension as a classification problem and connect the natural language representations of objects with objects in images (Matuszek et al., 2014; Alonso-Martín et al., 2015), although it turned out that classification plays an important role, and they rely on human intervention heavily, leading to less autonomous level. Shridhar et al. (2020) proposes an end-to-end INGRESS algorithm to generate textual descriptions of the objects in the image, and then relevancy clustering is performed with the object descriptions of human instructions for extracting the object with the highest matching score. Additionally, for multiple ambiguous objects, the robot can remove the ambiguity by identifying the objects. Hatori et al. (2018) uses the Convolutional Neural Network (CNN) and Long short-term memory (LSTM) to extract the features of the image and the text, respectively, and subsequently fuses visual and auditory information by a multi-layer perceptron. Magassouba et al. (2019) employed the Multimodal Target-source Classifier Model (MTCM) to predict region-wise likelihood of the target for selecting the object mentioned by instructions. Some works learn models for color, shape, object, haptics, and sound with predefined unique feature channels have resulted in successful groundings (Mooney, 2008; Dzifcak et al., 2009; Richards and Matuszek, 2019) explores using a set of general features to learn groundings outside of predefined feature channels. Despite these methods being relatively flexible to determine the target object described by natural language instructions, they cannot enable robots to understand the connections between different concepts. The capacity of understanding these connections determines the adaptability and flexibility of processing unstructured natural language instructions. If robots are able to flexibly parse and infer natural language sentences, users may have better experiences. For example, we expect robots to understand that "I am thirsty after running that far in such a hot day" means "Grasp a bottle to me," and "I need to feed the little rabbit" means "Grasp a carrot to me."

In order to achieve this goal, we propose a task-based framework combining both visual and auditory information to enable robots understand human intention from natural language dialogues. We first utilize the conditional random field (CRF) to extract task-related information from instructions, and complement a number of new sentences based on the matching rule. Then we apply Mask R-CNN (He et al., 2017) for instance segmentation and classification, and use sense2vec (Trask et al., 2015) to generate structured robot control language (RCL) (Matuszek et al., 2013); RCL is a robot-executable command for instruction. It represents the high-level execution intended by the person. It enables robots to perform actions in the specified tasks satisfying human requirements. To evaluate the efficacy of our approach, we classify human instructions into the following three types: Clear Natural Language Instructions, saying object names or synonyms clearly; Vague Natural Language Instructions, only providing object characteristics (hypernyms, related nouns, related verbs, etc.) without saying their names or synonyms; Feeling Natural Language Instructions, describing feelings of users in the scene without saying object names or synonyms. In such a manner, by transforming unstructured natural language instructions into robot-comprehensible structured language (RCL), robots can understand human intentions without the restriction of explicit expressions, and can comprehend connections between demand concepts and objects.

2. METHODS

2.1. Image Recognition

In this work, we mainly use the Mask R-CNN for image recognition. The Mask R-CNN is improved on the basis of Fast R-CNN (Girshick, 2015) and Faster R-CNN (Ren et al., 2015). The architecture of Faster R-CNN integrates feature extraction, region proposal selection, bounding box regression, and classification, resulting in a significantly enhanced speed of object detection. The Mask R-CNN is inspired by Faster R-CNN with outputting both bounding boxes and binary masks, so object detection and instance segmentation are carried out simultaneously. In our work, we employ Resnet101-FPN as a backbone and use the result of instance segmentation as the image region to be matched, including the target object and the delivery place.

2.2. Information Extraction From Natural Language Instructions

We first use a rule matching method for preliminarily extracting natural language information. Furthermore, this method provides labels for the conditional random fields process to reduce labor intensity.

2.2.1. Rule Matching

Rule matching uses linguistics as a fundamental principle to segment statements and label sentence components with predefined semantic information. The reason why rule matching is effective in parsing languages is that the languages are regular when they are restricted to a specific domain. Specifically, according to grammatical features, the sentence type is straightforward to identify, and the local feature of specific sentence types can be further utilized to extract key information. In this paper, two variables, i.e., lexical and dependency analysis, are selected. Compared to many existing studies

TABLE 1 | Skills and details of the skills.

Instruction type	Sentence structure	Target object	Delivery place
Feeling type	Subject (user) + tether verb + epithet + other components	Words that are adjective and begin with a tethered verb in dependency analysis, words that are adjective and begin with an adverb in dependency analysis, etc.	Words that are personal pronouns and end in a nominal subject in dependency analysis, etc.
Vague type	Subject + modal verb + intransitive verb + other components	Words that are verbs and are the end of an open subordinate complement in dependency analysis, etc.	Words that are personal pronouns and end in a nominal subject in dependency analysis, etc.
Clear/Vague type	Subject + modal verb + transitive verb + noun + other components	Words that are common nouns and plural nouns, and are at the end of the direct object in dependency analysis.	Words that are personal pronouns and end in a nominal subject in dependency analysis, etc.
Vague type	Predicate + direct object (something) + indirect object + definite article (adjective or verb infinitive) + other constituents	Words that are verbs and end in a modifier in dependency analysis, words that are adjectives and end in an adjective modifier in dependency analysis, etc.	Words that are personal pronouns and end in a noun subject in dependency analysis, words that are personal pronouns and end in an indirect object in dependency analysis, etc.
Feeling type	It (for weather) + verb past tense or verb present progressive + other components	Words that are in the past tense of the verb and begin with the noun subject in dependency analysis, words that are in the present tense of the verb and begin with a non-primary verb in dependency analysis, etc.	System default users, etc.

with grasping robots, ours not only contain the single verb phrase-centered imperative sentence structure but also add many common sentence types for expressing intentions through natural language in the training set. These common sentences are selected from the three types described in section 1. The details of rule matching connecting sentence structure and instruction types are displayed in **Table 1**.

2.2.2. Conditional Random Fields

Although the rule matching method extracts key information from natural language with sufficient accuracy, it is inadequate because it still requires grammatical features to identify sentence types before parsing natural language. However, when the length and complexity of the instructions increase, the fixed rule may classify sentence types of the instructions incorrectly or extract unexpected information because of the interference by redundant information. Besides, high-frequency word features are not contained in the grammatical rule due to the limited and time-consuming enumeration work. Therefore, for further extraction of natural language information, a statistical model is necessary to integrate grammar and high-frequency words for mining specific local features.

We use the CRF model for information extraction, whose training data are labeled by the rule matching described previously. The process of extracting information from a sentence can be considered as sequence labeling. The model analyzes input natural language sequences, i.e., sentences, and outputs the label corresponding to each word. In this paper, the tag set is item, target, none, where "item" represents the keyword of the target object, "target" corresponds to the keyword of the delivery place, and "none" is the other components of the sentence.

The CRF is a common and efficient method for addressing the sequence labeling problem, and its principle is based on a probabilistic vectorless graph. In this paper, any sentence $x(x1, x2, ..., xn)$ has $3n$ possible label sequences $y(y1, y2, ..., yn)$, where (xi, yi) represents (word, word label). The probability of labeled sequence y is written as:

$$p(y|x) = \frac{e^{score(y|x)}}{\sum_{y'} e^{score(y'|x)}} \tag{1}$$

$$score(y|x) = \sum_{j=1}^{m} \sum_{i=1}^{n} \lambda_j f_j(x, i, j) \tag{2}$$

where $f_j(x, i, j)$ is j^{th} feature function at position i and usually is a binary function, generated by a feature template, which is broader in this study according to the variety of the instructions. At position i, $(y|x)$ takes 1 when it satisfies the j^{th} feature function, otherwise takes 0. Parameter λ is the parameter to be learned. The objective of training model is to maximize the probability of the correctly labeled sequence. The size of m depends on the variety of training corpus, the number of variables, and the maximum offset.

2.3. RCL Generating

In order to enable the robot to understand the highly arbitrary instructions provided by users and to grasp the target object to the delivery place, unstructured natural language instructions should be transformed into structured RCL. The RCL format utilized in this paper is "Grasp A to B," where A and B represent the target object and the delivery place, respectively. In this work, the RCL format is generated from natural language instructions by extracting the keyword of the target object and place based on the information extraction module of CRF. Simultaneously, the image recognition module of Mask R-CNN is utilized for instance segmentation and classification. We map the extracted features of natural language instructions and images in the same feature space, and compare the degree of match between each object and

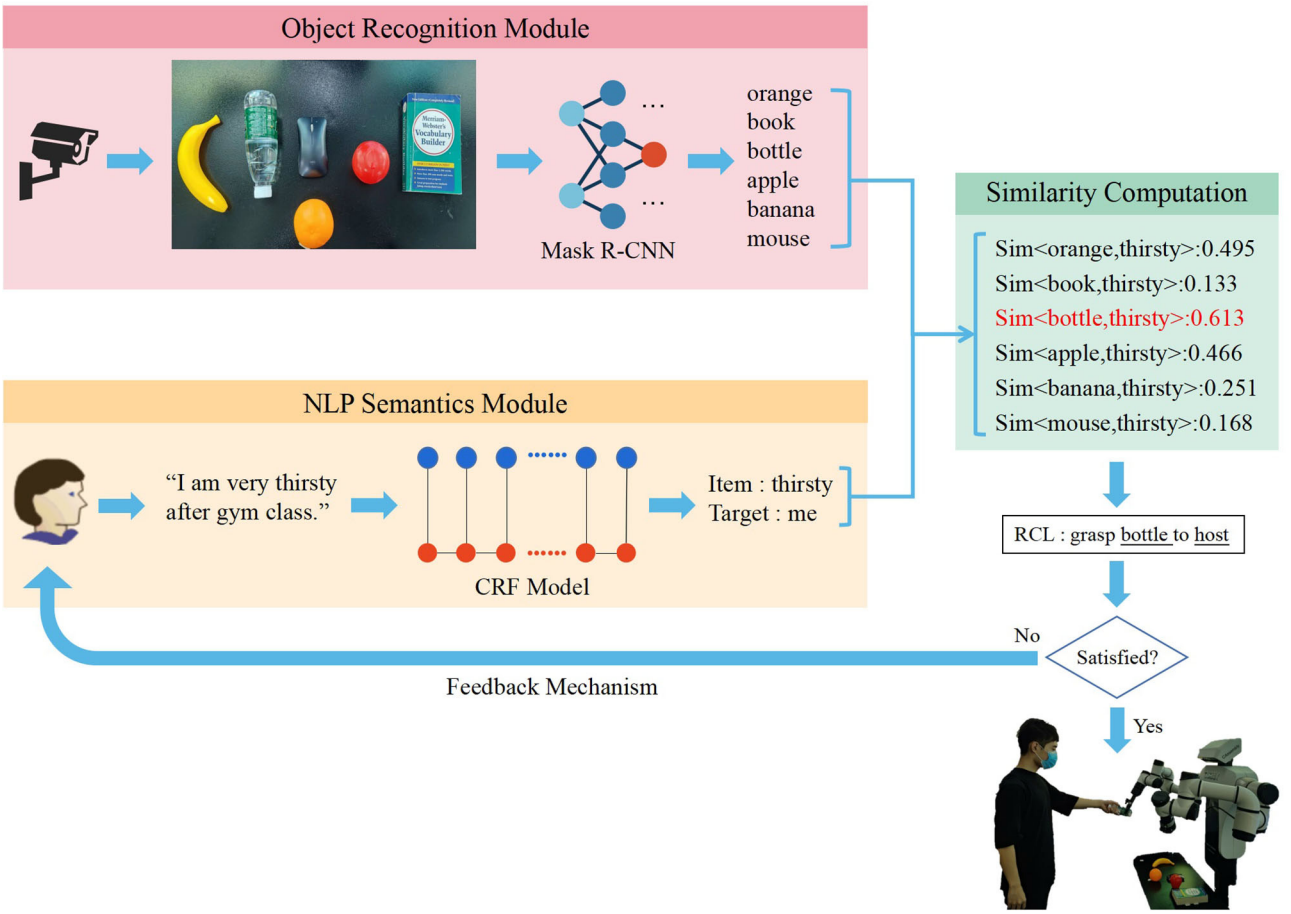


FIGURE 1 | The overall framework: Once natural language information extraction and image recognition are completed, the features of natural language and the image are transformed into the same space for finding the object with the highest matching score. We transform both the object recognized from the image and the task-related information into word vectors to obtain the word vectors with maximum similarity. The vectors are then put into RCL (grasp object to place) to generate the structured language comprehended by robots.

two keywords. The two objects with the highest scores are A and B for generating the structured RCL language, "Grasp A to B." The overall framework is shown in **Figure 1**.

We use the sense2vec model, which is an improved version of word2vec model, to transform the key information of images and natural language instructions to the same feature space. When words are fed into this model, the corresponding sense information is also required. Compared to the word vectors computed without context, those generated by sense2vec model contain contextual information and single vectors of corresponding compound words. Hence, the sense2vec model has more flexibility than the word2vec model. The sense2vec model employs CBOW, SG and structure-SG of word2vec, and uses token rather than a word as a semantic unit. Moreover, the same tokens with different tags are considered as different semantic units. The training process of the model is twofold. First, every token is labeled by a sense tag in the corresponding context. Second, the common models of word2vec, e.g., CBOW and SG, are fitted to the labeled data of the first step.

After the sense2vec model is used to obtain the objects according to the similarity between the information of target objects and object names in the scene, the degree of match is calculated. The object with the highest matching score is the target to grasp. We utilize cosine similarity, which is commonly used in word vector models, as an indicator of the degree of match between the objects and the keywords in instructions. The similarity is calculated as Equation (3), where *ITEM* denotes the item in the image and A denotes the word that is extracted by CRF, and $V(w)$ is the sense vector of w.

$$sim\ (ITEM, A) = \frac{V(ITEM) \cdot V(A)}{||V(ITEM)|| \times ||V(A)||} \tag{3}$$

2.4. Feedback Mechanisms

To make the robot grasp the item that humans want and be more robust, our system uses a feedback mechanism. When a user gives an instruction, the robot determines the target object and delivery place according to the instruction, and it asks the user whether the result is right.

TABLE 2 | Examples of the collected instructions.

Clear natural language instructions	Vague natural language instructions	Feeling natural language instructions
Can I have a cup of tea?	I'm going to feed my monkey.	I am thirsty.
I want to play sports ball.	I need to control TV.	I am hungry.
I'm so thirsty that I need a large cup of cola.	The dark clouds shows that it will rain soon.	I'm tired.

FIGURE 2 | Experimental setup for robot experiments. Our system uses Cobot CAssemblyC2 for experiment.

We divide user feedback into three types. The first type is positive feedback, the user thinks robot's judgment is right. In this situation, the robot grasps the target object to the delivery place. The second type is negative feedback without any other valid information. In this situation, the robot chooses the object with the second largest matching score as the target object. The last type is negative feedback with other valid information. The algorithm uses CRF to extract the information related to the target object and uses sense2vec to calculate a new matching score between the information of target objects and object names in the scene, and then it chooses a new target object according to the updated matching score. The new target object is chosen by the following formula:

$$object = \arg\min_{i}(\sum_{j=0}^{n} sim(item_i, A_j)) \tag{4}$$

where *object* denotes the target object, and $item_i$ denotes the i^{th} item in the image, and A_j and n denote the word extracted by CRF in the j^{th} time and number of feedback, respectively, and *sim* denotes the similarity calculated by Equation (3).

For example, there is a scene with an apple, an orange, a banana, a bottle, and a book. The instruction is "I want to eat fruit." Then the robot asks the user "Do you mean grasp the apple to host?" The feedback is "No, I want to eat something sour." Algorithm can choose "sour" as valid information and use sense2vec to calculate a new matching score. Then it can grasp the orange to host.

2.5. Grasp Object

Current data-driven methods have significantly increased the accuracy of grasping objects (Mahler et al., 2016, 2019; Kalashnikov et al., 2018; Quillen et al., 2018) and they provide the technical basis for human–computer interaction.

We are inspired by a state-of-the-art method Dexnet4.0 (Mahler et al., 2019) and use end-effectors based on parallel gripper in the implementation of this study. We first generate a series of candidate grasps by pre-computation and utilize Grasp Quality Convolutional Neural Network (GQ-CNN) to score these grasps. The grasp with the highest score is implemented by robots. Since we only employ the parallel gripper, only pre-trained parallel gripper policy is utilized.

The full process of grasping is as follows. After the RCL is generated, the robot can use it to grasp the object. The RCL format in this paper is "Grasp A to B." The system matches A and the results of image recognition. The matching result is a mask image. B is one of the predefined users. The mask image is the input of Dex-net2.0 that is used to determine the object to be grasped. Dex-net2.0 can generate a grasp position of the object. Then the robot arm will move to the position and grasp the object to the predefined user.

3. RESULTS

We design experiments as follows. Microsoft COCO is a dataset for image recognition, and it provides many items that often appear in the home environment. We exclude items that are inappropriate to application scenarios from the Microsoft COCO (Lin et al., 2014). A total of 41 items remain and are categorized into 7 classes (animal, accessory, kitchen, sports, electronic, indoor, and food). Each experiment contains 3 categories of items and each category has some corresponding items, and we call it a scenario. Thus, there are altogether 35 scenarios, and each scenario includes more than 20 items. In each scenario, 8

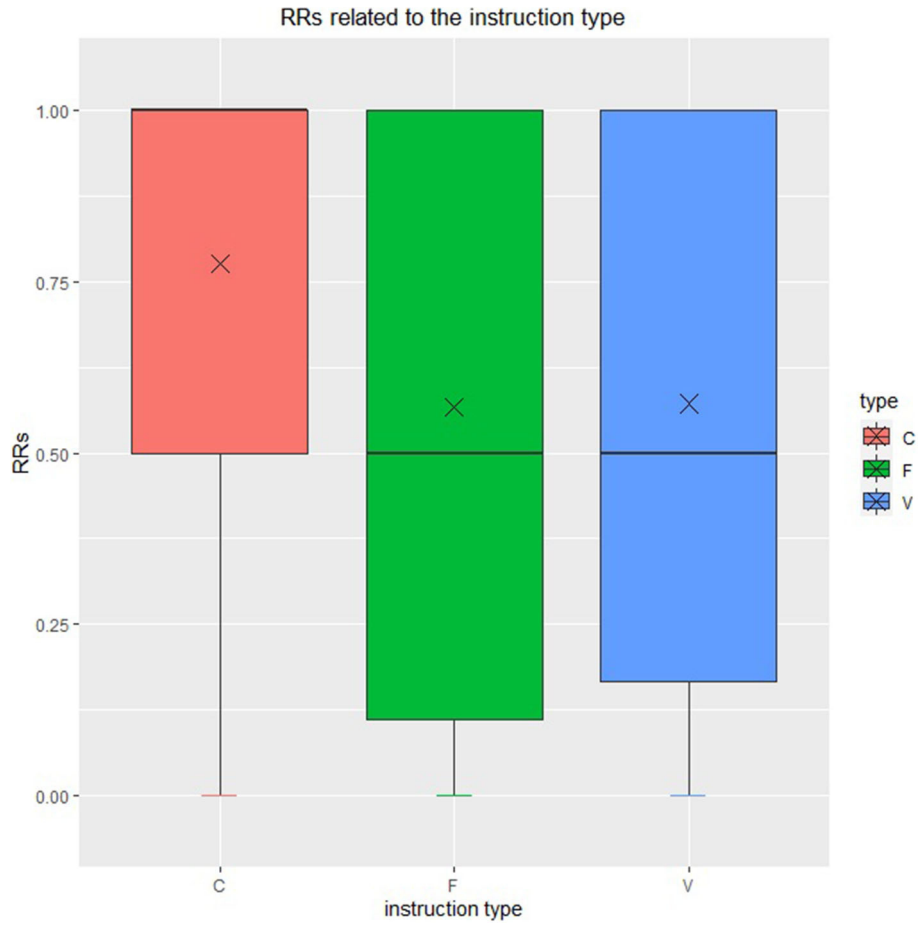


FIGURE 3 | Type-specific reciprocal rank. Red, green, and blue represent clear natural language instruction, feeling natural language instruction and vague natural language instruction.

subjects provide random instructions to the robots. Each subject provides 3 instructions containing the objects in the scene and lists of expected items for each instruction. There are 21 natural language instructions in each scenario, and 735 instructions in total. We show some examples of the collected instructions in **Table 2**.

3.1. Accuracy of Information Extraction

To enable robots to accurately parse complicated sentence structures, we apply the CRF model to extract information. The rule matching method is only for generating and evaluating the data of the CRF model. Therefore, quantitative evaluation of this method is not involved in this study.

We use 735 sentences collected before to test the accuracy of our CRF model's ability to extract the target object and the delivery place. We evaluate our CRF model in clear natural language instructions, vague natural language instructions, and feeling natural language instructions, respectively. The formula is as follows:

$$accuracy = \frac{\sum_{i=0}^{n} Is\_true(object_i) * Is\_true(place_i)}{n} \quad (5)$$

where *accuracy* denotes the accuracy of the algorithm, and *Is_true* denotes whether the $object_i$ is true. n denotes the number of instructions, and $object_i$ and $place_i$ denote the target object and place that are output by the algorithm.

The accuracy of the CRF model for clear natural language instructions, vague natural language instructions, and feeling natural language instructions are 0.710, 0.656, and 0.711, respectively. This result indicates that our method has consistent performance over all three types of instructions. By analyzing the failure cases, we found that the wrong inferred item and the wrong inferred target are most likely due to the deficiency in training data that reflect their local features. The local features are referred to words, positions, and dependency.

3.2. Evaluation of Human–Robot Interaction

To obtain meaningful results, we evaluate our system's human–robot interaction ability in the scenarios. There are 21 instructions that are provided by 8 subjects in each scenario. The experimental setup is shown in **Figure 2**.

Our system uses a feedback mechanism. The robot has a ranking list according to matching score. If a user gives negative

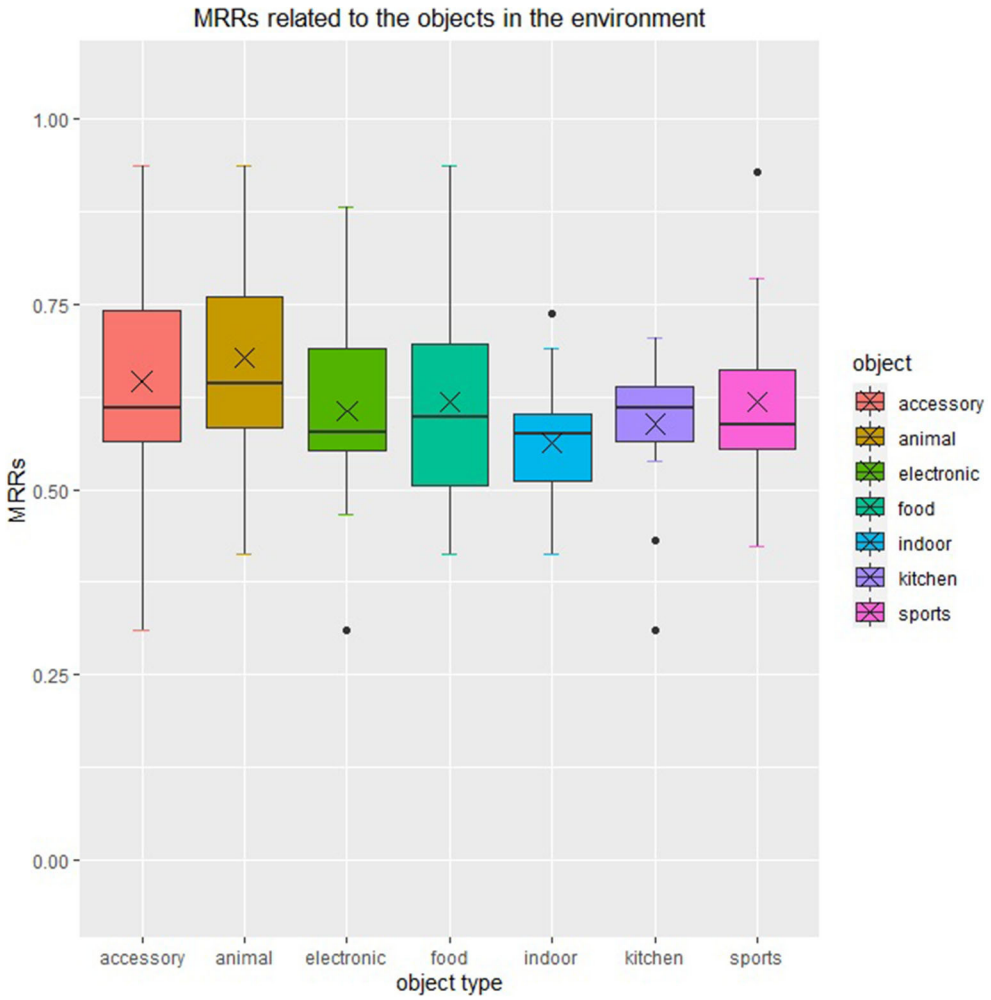


FIGURE 4 | Scene-specific mean reciprocal rank. This figure shows the relevance between scene-specific reciprocal rank and the categories in each scene.

feedback without any other valid information, the robot is able to choose the object with the second largest matching score as the target object, and so on. Therefore, we use reciprocal rank (RR) as the evaluation of our system. RR is a measure to evaluate systems that return a ranked list of answers to queries, and mean reciprocal rank (MRR) is the mean of the sum of RR. The formulas are given by:

$$RR_i = \frac{1}{Position(item)} \tag{6}$$

$$MRR = \frac{\sum_{i=1}^{N} RR_i}{N} \tag{7}$$

where $Position(ITEMi)$ represents the position of the real target object in the matching score list, and N is the number of instructions in each scenario, and RR_i is the reciprocal rank of ith instruction within each scenario.

The distributions of type-specific RR are demonstrated in **Figure 3**. The mean reciprocal ranks of clear natural language instruction, feeling natural language, and vague natural language is 0.776, 0.567, and 0.572, respectively. The medians is 1 for clear natural language instruction, which shows that the robot can grasp the correct object at the first attempt according to clear natural language instruction in most cases. The mean reciprocal rank of all instructions is 0.617, which means the robot need about 1–2 attempts to grasp the correct object according to the three types of instruction at the average level. Thus, we draw a conclusion that the robots infer the expected item effectively, and especially, the robots make inference most effectively and most steadily according to clear natural language instructions among the three types of instructions defined as before. The result also shows our framework's ability to interact with people.

We group the MRR by categories in their corresponding scene, with intersections existing among groups. The result of our experiment is shown in **Figure 4**, which indicates that the robots

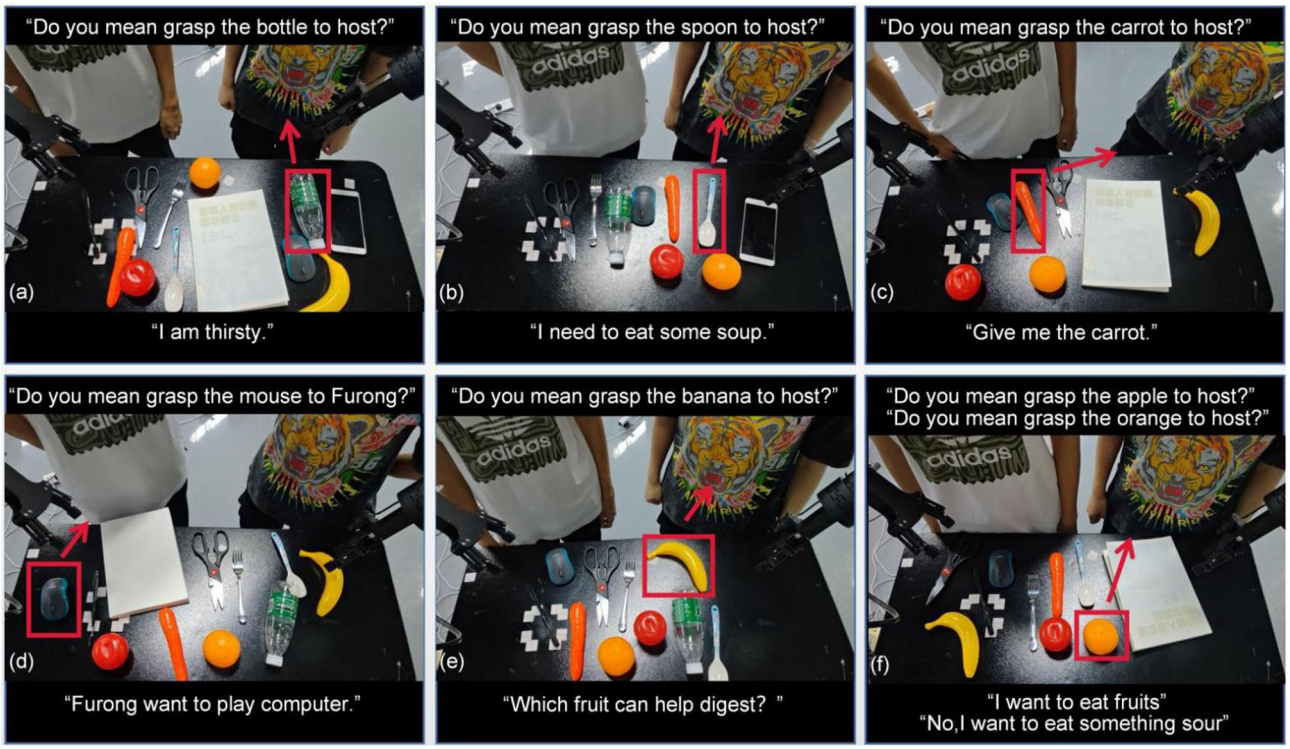


FIGURE 5 | Some examples of human–robot interaction. There are two user in these scenarios. The one on the right is host, and the other one is Furong. Red boxes indicate the target objects chosen by our method. Red arrows indicate the delivery directions.

perform best and relatively steadily when items in "animal" category appear in the scene, and perform worst and relatively unsteadily when items in "indoor" and "food" categories appear in the scene. It is because that the items in these categories always appear in a similar context. It is also related to the word embedding model.

The human–robot interaction ability of our system is shown in **Figure 5**. **Figures 5a–c** illustrate the interaction for feeling natural language instruction, vague natural language instruction, and clear natural language instruction, respectively. **Figure 5d** illustrates that our method can grasp objects to a different user. **Figure 5e** illustrates our method's ability to adapt to instructions that have untrained sentence structures, which is an interrogative question in this case. **Figure 5f** shows the feedback mechanism of our method. The robot can grasp the orange because of the feedback information that says he wants to eat something sour.

3.3. The Ability to Deal With Unseen Sentence

We also note that this algorithm has a generalization capability to some extent. It can analyze a question like "Which item can help me use computers more efficiently?," even though this sentence type is not involved in the training set. Therefore, we choose 104 instructions that have unseen sentence structures to test the generalization capability of our approach, such as interrogative sentences and complex sentences.

The mean reciprocal rank for instructions that have untrained sentence structures is 0.483, which means the site of the target object is in the second position in the recommended list on average, and the robot can grasp the correct object with about 2–3 attempts at the average level.

This also shows that our model has a generalization capability to interact with complex instructions that have unseen sentence structures.

4. CONCLUSION

Our proposed algorithm transforms unstructured natural language information and environmental information into structured robot control language, which enables robots to grasp objects following the actual intentions of vague, feeling, and clear type instructions. We evaluate the algorithm performance using a human–robot interaction task. The experimental results demonstrate the ability of our algorithm interacting with different types' instructions and a generalization ability of unseen sentence structures. Although some sentence types are not involved in the training set, the carried information still can be effectively extracted, leading to reasonable intention understanding.

In our future work, we would construct the databases based on multiple tasks to extend its skill coverage, and explore its potential in understanding more complex tasks.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

ETHICS STATEMENT

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

AUTHOR CONTRIBUTIONS

ZL proposed the framework to enable the robot to comprehend human intentions from vague natural language instructions. ZS, ZL, and YM modified and finalized the design of this framework. ZS also proposed the feedback mechanism that improve the robust of our algorithm. YM and ZL collect the data of instructions, and they also conducted the experiment together. SS and JS helped edit this paper, and they give us many useful advice. JZ provided theoretical guidance for the proposed method and experiments design. All authors contributed to the article and approved the submitted version.

FUNDING

This work was partially supported by the Shenzhen Science and Technology Innovation Commission (Grants Nos. JCYJ20170410172100520 and JCYJ20170818104502599). This work was also partially supported by the Shenzhen Institute of Artificial Intelligence and Robotics for Society (Grants No. 2019-INT020).

REFERENCES

Alonso-Martín, F., Castro-González, A., de Gorostiza Luengo, F. J. F., and Salichs, M., Á. (2015). Augmented robotics dialog system for enhancing human-robot interaction. *Sensors* 15, 15799–15829. doi: 10.3390/s150715799

Dzifcak, J., Scheutz, M., Baral, C., and Schermerhorn, P. (2009). "What to do and how to do it: translating natural language directives into temporal and dynamic logic representation for goal management and action execution," in *2009 IEEE International Conference on Robotics and Automation* (Kobe: IEEE), 4163–4168. doi: 10.1109/ROBOT.2009.5152776

Eppe, M., Trott, S., and Feldman, J. (2016). "Exploiting deep semantics and compositionality of natural language for human-robot-interaction," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Daejeon: IEEE), 731–738. doi: 10.1109/IROS.2016.7759133

Fang, B., Sun, F., Liu, H., and Liu, C. (2018). 3d human gesture capturing and recognition by the immu-based data glove. *Neurocomputing* 277, 198–207. doi: 10.1016/j.neucom.2017.02.101

Fang, B., Wei, X., Sun, F., Huang, H., Yu, Y., and Liu, H. (2019). Skill learning for human-robot interaction using wearable device. *Tsinghua Sci. Technol.* 24, 654–662. doi: 10.26599/TST.2018.9010096

Girshick, R. (2015). "Fast r-CNN," in *Proceedings of the IEEE International Conference on Computer Vision*, (Santiago) 1440–1448. doi: 10.1109/ICCV.2015.169

Hatori, J., Kikuchi, Y., Kobayashi, S., Takahashi, K., Tsuboi, Y., Unno, Y., et al. (2018). "Interactively picking real-world objects with unconstrained spoken language instructions," in *2018 IEEE International Conference on Robotics and Automation (ICRA)* (Brisbane, QLD: IEEE), 3774–3781. doi: 10.1109/ICRA.2018.8460699

He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). "Mask r-CNN," in *Proceedings of the IEEE International Conference on Computer Vision*, (Venice) 2961–2969. doi: 10.1109/ICCV.2017.322

Hou, J., Wu, X., Zhang, X., Qi, Y., Jia, Y., and Luo, J. (2020). "Joint commonsense and relation reasoning for image and video captioning," in *AAAI*, 10973–10980. doi: 10.1609/aaai.v34i07.6731

Kalashnikov, D., Irpan, A., Pastor, P., Ibarz, J., Herzog, A., Jang, E., et al. (2018). "Scalable deep reinforcement learning for vision-based robotic manipulation," in *Conference on Robot Learning*, (Zürich), 651–673.

Kollar, T., Tellex, S., Roy, D., and Roy, N. (2010). "Toward understanding natural language directions," in *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* (IEEE), 259–266. doi: 10.1109/HRI.2010.5453186

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., et al. (2014). "Microsoft COCO: common objects in context," in *European Conference on Computer Vision* (Amsterdam: Springer), 740–755. doi: 10.1007/978-3-319-10602-1_48

Magassouba, A., Sugiura, K., Quoc, A. T., and Kawai, H. (2019). Understanding natural language instructions for fetching daily objects using gan-based multimodal target-source classification. *IEEE Robot. Automat. Lett.* 4, 3884–3891. doi: 10.1109/LRA.2019.2926223

Mahler, J., Matl, M., Satish, V., Danielczuk, M., DeRose, B., McKinley, S., et al. (2019). Learning ambidextrous robot grasping policies. *Sci. Robot.* 4. doi: 10.1126/scirobotics.aau4984

Mahler, J., Pokorny, F. T., Hou, B., Roderick, M., Laskey, M., Aubry, M., et al. (2016). "DEX-Net 1.0: a cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards," in *2016 IEEE International Conference on Robotics and Automation (ICRA)* (Stockholm: IEEE), 1957–1964. doi: 10.1109/ICRA.2016.7487342

Matuszek, C., Bo, L., Zettlemoyer, L., and Fox, D. (2014). Learning from unscripted deictic gesture and language for human-robot interactions. *AAAI*, 2556–63. doi: 10.13016/M2RN30B52

Matuszek, C., Herbst, E., Zettlemoyer, L., and Fox, D. (2013). "Learning to parse natural language commands to a robot control system," in *Experimental Robotics* (Springer), 403–415. doi: 10.1007/978-3-319-00065-7_28

Mooney, R. J. (2008). "Learning to connect language and perception," in *AAAI*, (Chicago) 1598–1601.

Paul, R., Barbu, A., Felshin, S., Katz, B., and Roy, N. (2018). Temporal grounding graphs for language understanding with accrued visual-linguistic context. *arXiv[Preprint].arXiv: 1811.06966*. doi: 10.24963/ijcai.2017/629

Quillen, D., Jang, E., Nachum, O., Finn, C., Ibarz, J., and Levine, S. (2018). "Deep reinforcement learning for vision-based robotic grasping: a simulated comparative evaluation of off-policy methods," in *2018 IEEE International Conference on Robotics and Automation (ICRA)* (Brisbane, QLD: IEEE), 6284–6291. doi: 10.1109/ICRA.2018.8461039

Reddy, D., and Raj, D. (1976). Speech recognition by machine: a review. *Proc. IEEE* 64, 501–531. doi: 10.1109/PROC.1976.10158

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39, 1137–1149.

Richards, L. E., and Matuszek, C. (2019). "Learning to understand non-categorical physical language for human robot interactions," in *From the RSS Workshop on AI and its Alternatives in Assistive and Collaborative Robotics*, (Messe Freiburg).

Shridhar, M., Mittal, D., and Hsu, D. (2020). INGRESS: interactive visual grounding of referring expressions. *Int. J. Robot. Res.* 39, 217–232. doi: 10.1177/0278364919897133

Trask, A., Michalak, P., and Liu, J. (2015). sense2vec-a fast and accurate method for word sense disambiguation in neural word embeddings. *arXiv[Preprint].arXiv: 1511.06388*.

Uzkent, B., Yeh, C., and Ermon, S. (2020). "Efficient object detection in large images using deep reinforcement learning," in *The IEEE Winter Conference on Applications of Computer Vision*, (Snowmass Village, CO) 1824–1833. doi: 10.1109/WACV45572.2020.9093447

Xie, E., Sun, P., Song, X., Wang, W., Liu, X., Liang, D., et al. (2020). "Polarmask: single shot instance segmentation with polar representation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (Seattle, WA) 12193–12202. doi: 10.1109/CVPR42600.2020.01221

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

frontiers in Neurorobotics

ORIGINAL RESEARCH
published: 08 March 2021
doi: 10.3389/fnbot.2021.570507

Failure Handling of Robotic Pick and Place Tasks With Multimodal Cues Under Partial Object Occlusion

Fan Zhu[1], Liangliang Wang[2], Yilin Wen[1], Lei Yang[1], Jia Pan[1], Zheng Wang[3] and Wenping Wang[1]*

[1] Department of Computer Science, The University of Hong Kong, Hong Kong, Hong Kong, [2] Department of Mechanical Engineering, The University of Hong Kong, Hong Kong, Hong Kong, [3] Department of Mechanical and Energy Engineering, Southern University of Science and Technology, Shenzhen, China

The success of a robotic pick and place task depends on the success of the entire procedure: from the grasp planning phase, to the grasp establishment phase, then the lifting and moving phase, and finally the releasing and placing phase. Being able to detect and recover from grasping failures throughout the entire process is therefore a critical requirement for both the robotic manipulator and the gripper, especially when considering the almost inevitable object occlusion by the gripper itself during the robotic pick and place task. With the rapid rising of soft grippers, which rely heavily on their under-actuated body and compliant, open-loop control, less information is available from the gripper for effective overall system control. Tackling on the effectiveness of robotic grasping, this work proposes a hybrid policy by combining visual cues and proprioception of our gripper for the effective failure detection and recovery in grasping, especially using a proprioceptive self-developed soft robotic gripper that is capable of contact sensing. We solved failure handling of robotic pick and place tasks and proposed (1) more accurate pose estimation of a known object by considering the edge-based cost besides the image-based cost; (2) robust object tracking techniques that work even when the object is partially occluded in the system and achieve mean overlap precision up to 80%; (3) contact and contact loss detection between the object and the gripper by analyzing internal pressure signals of our gripper; (4) robust failure handling with the combination of visual cues under partial occlusion and proprioceptive cues from our soft gripper to effectively detect and recover from different accidental grasping failures. The proposed system was experimentally validated with the proprioceptive soft robotic gripper mounted on a collaborative robotic manipulator, and a consumer-grade RGB camera, showing that combining visual cues and proprioception from our soft actuator robotic gripper was effective in improving the detection and recovery from the major grasping failures in different stages for the compliant and robust grasping.

Keywords: soft robot applications, pick and place, failure handling, visual tracking, proprioception

OPEN ACCESS

Edited by:
Ganesh R. Naik,
Western Sydney University, Australia

Reviewed by:
Hao Su,
City College of New York, United States
Hu Cao,
Technical University of Munich, Germany
Xikai Tu,
North Carolina State University, United States

****Correspondence:***
Zheng Wang
zheng.wang@ieee.org

Received: *08 June 2020*
Accepted: *29 January 2021*
Published: *08 March 2021*

Citation:
Zhu F, Wang L, Wen Y, Yang L, Pan J, Wang Z and Wang W (2021) Failure Handling of Robotic Pick and Place Tasks With Multimodal Cues Under Partial Object Occlusion. Front. Neurorobot. 15:570507. doi: 10.3389/fnbot.2021.570507

1. INTRODUCTION

The success of a robotic pick and place task depends on the success of the entire procedure: from the planning phase (object detection and grasp planning), to the grasping phase (actually establishing the grasp), to the lifting and moving phase (transit the object toward target site), and the final releasing phase (descending the object and release the grasp). Being able to detect and recover from grasping failures throughout the entire process is therefore a critical requirement for both the robotic manipulator and the gripper (see **Figure 1**).

Grasp planning aims at generating better grasping proposals to improve the success rate of robotic grasping. It can be categorized as grasp detection based (Kumra and Kanan, 2017; Zito et al., 2019; Li et al., 2020) and direct image-to-grasping manner. The former mainly generates the grasping proposals for the novel objects and it utilizes grasping contacts to compensate for the pose uncertainty. The latter detects structured grasp representations from images by the pose estimation of a known object (Sundermeyer et al., 2018).

When establishing grasp and moving the target to the destination, it is critically important to detect and recover from any accidental failure in real scenarios. Even with an excellent grasp planning, unexpected failure may still occur in the pick and place task due to environmental changes or intrinsic systematic errors. Without an effective failure detection and recovery mechanism, the robotic system may crack accidentally and be less efficient.

Visual servoing (Cowan et al., 2002; Kragic et al., 2002) was popular for guiding the above phases in the robotic system. Some typical object tracking algorithms (Grabner et al., 2006; Bolme et al., 2010; Kalal et al., 2010) have been well studied. Li et al. (2020) built a sensing pipeline through a neuromorphic vision sensor DAVIS to satisfy the real-time features in object detection and tracking. However, preserving visibility of the target has been the key to robust object tracking in these algorithms and the performance of algorithms becomes much weakened when the partial occlusion exists. Robust tracking techniques under the partial object occlusion are of great significance to a robust robotic grasping system.

Meanwhile, under-actuated robotic grippers (Zhou et al., 2017) recently tend to have a variety of advantages over the rigid-bodied counterparts when the gripper is interacting with the environment. There are numerous grippers with novel designs of compliant mechanisms, working as both actuators and sensors to generate movement and provide proprioceptive feedback simultaneously (Su et al., 2020; Zhou et al., 2020). Endowing soft robotic grippers with proprioception enables reliable interactions with environment.

In this paper, we aim to investigate an effective grasping system from the beginning to the endpoint, by considering the partial object occlusion as a normal condition. We especially focus on the failure detection and recovery framework in the grasping system by combining the specific proprioceptive capability of our soft gripper and the visual cues from the highly obstructed view when the failure occurs. The proprioceptive soft gripper used in the paper was developed in our recent work (Wang and Wang, 2020). It was pneumatically driven by soft bellows actuator and the pressure of the actuator was leveraged for sensing the gripper movement and external contact (see **Figure 5**). The main contributions and novelties are listed as follows:

(1) more accurate pose estimation of a known object by considering the edge-based cost besides the image-based cost;
(2) robust object tracking techniques that work even when the object is partially occluded in the system and achieve mean overlap precision (OP) up to 80%;
(3) contact and contact loss detection between the object and the gripper by analyzing internal pressure signals of our gripper;
(4) robust failure handling of robotic pick and place tasks with the combination of visual cues under partial occlusion and proprioceptive cues from our soft gripper to effectively detect and recover from different accidental grasping failures.

2. SYSTEM ARCHITECTURE

2.1. System Modeling

The setup we considered consists of an RGB camera and a proprioceptive gripper, which are equipped on the robot arm. The robot arm is controlled by an operator acting on a master device and interacting with the environment by combining the proprioception of our soft gripper (Wang and Wang, 2020) and the visual cues from the camera view. The relative coordinates of the camera and the testbed are first calibrated, and the depth is accordingly computed. We assume all the objects are put on the same testbed. The system setup is illustrated in **Figure 2**.

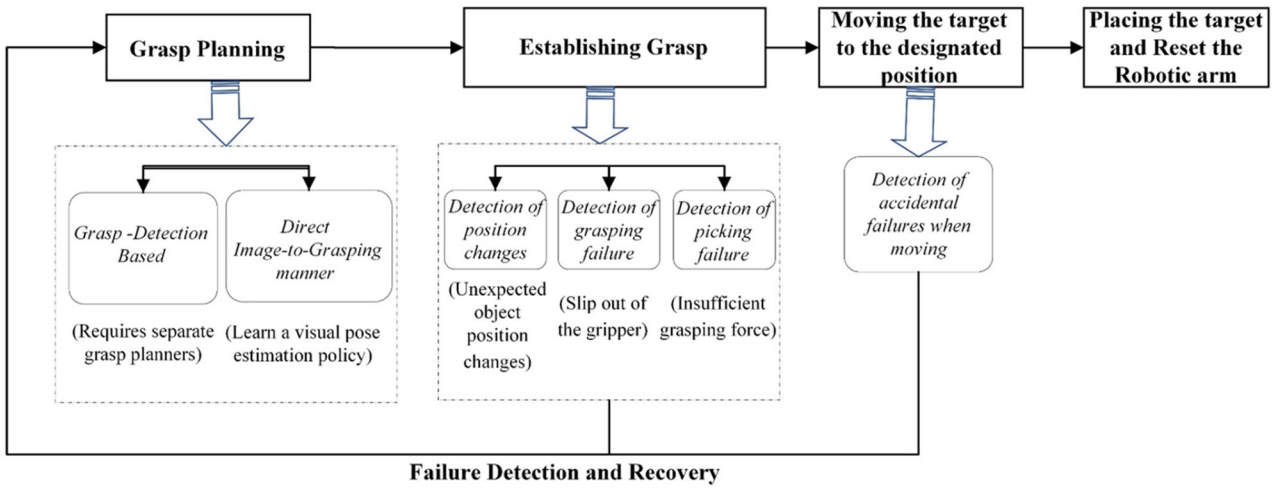
Our system first performs automatic target detection and poses estimation based on an RGB image and an edge map. Then a robust object tracking algorithm continuously works to provide real-time visual cues for failure detection and recovery, even if the object is highly obstructed in the camera view. Meanwhile, the proprioceptive capability of our soft gripper (Wang and Wang, 2020) is utilized in the system to sense the contact between the object and the gripper. We measure the actuation pressure in the soft actuator chambers to extract the external contact force and further reflect the contact status between the gripper and object. The proprioceptive capability is combined with visual cues to guarantee the effectiveness of our system.

2.2. Workflow Illustration

The proposed multi-sensor collaboration architecture aims at facilitating effectiveness of failure detection and recovery in the grasping. A general illustration of the system pipeline is shown in **Figure 2**.

The input of our system is the starting frame recorded by the in-hand camera. We first aim at target detection and pose estimation. The target is first assigned by the user and denoted as the number corresponding to the predefined template. Then the target is detected on the query image and the target's pose is estimated by template retrieval with an image and edge cost. For the determination of the target's pose, previous work (Zhou et al., 2017) prefer to first detect objects without recognition.

A



B

	Kumra and Kanan,2017	Zito et al.,2019	Li et al.,2020	Ours
Grasp Planning	Grasp-Detection Based	Grasp-Detection Based	Grasp-Detection Based	Direct Image-to-Grasping Manner; Estimate the object pose before Grasping
Vision	RGB-D Visual Sensing System; Partial Occlusion NOT Discussed	RGB-D Visual Sensing System; Partial Occlusion NOT Discussed	Neuromorphic Vision Sensor DAVIS; Partial Occlusion NOT Discussed	RGB Visual Sensing System; Partial Occlusion Discussed
Detection of Real-time Failures after Grasping Planning	No	No	No	Three Phases of Detection
Robotic Gripper	Parallel Plate Gripper(Hard)	Gripper with Multi-joint Fingers(Hard)	Parallel Plate Gripper(Hard)	Self-designed Prospective Gripper with Inner Soft Pneumatic Bellow Actuators

FIGURE 1 | (A) The entire procedure of robotic pick and place task. **(B)** Comparative analysis with other methods.

Then a planner, such as MoveIt planner (Coleman et al., 2014), is implemented to generate multiple motion plans and intuitively determine the pose. Compared with our previous work (Zhou et al., 2017), pose estimation in this paper is more efficient based on object recognition. Because any accidental changes or failures affect subsequent steps in grasping, we design three phases of detection by combining the visual and proprioceptive cues to improve the effectiveness of our system. The first detection is designed for disturbance from external factors. For example, the target may be accidentally moved as the gripper is approaching. Our system detects position changes with the proposed object tracking algorithm. It can still robustly work in the challenging scenario that the target is partially occluded by the gripper in the camera view. If the position change of target is not detected in visual tracking, a grasping trial will be executed. Otherwise, if the target is moved, our system will relocate and track the target in the current camera view. If the target is reported lost in the current view, the arm will be reset. Target detection and pose estimation will be executed in the new camera view. The second detection aims at checking if the last grasping

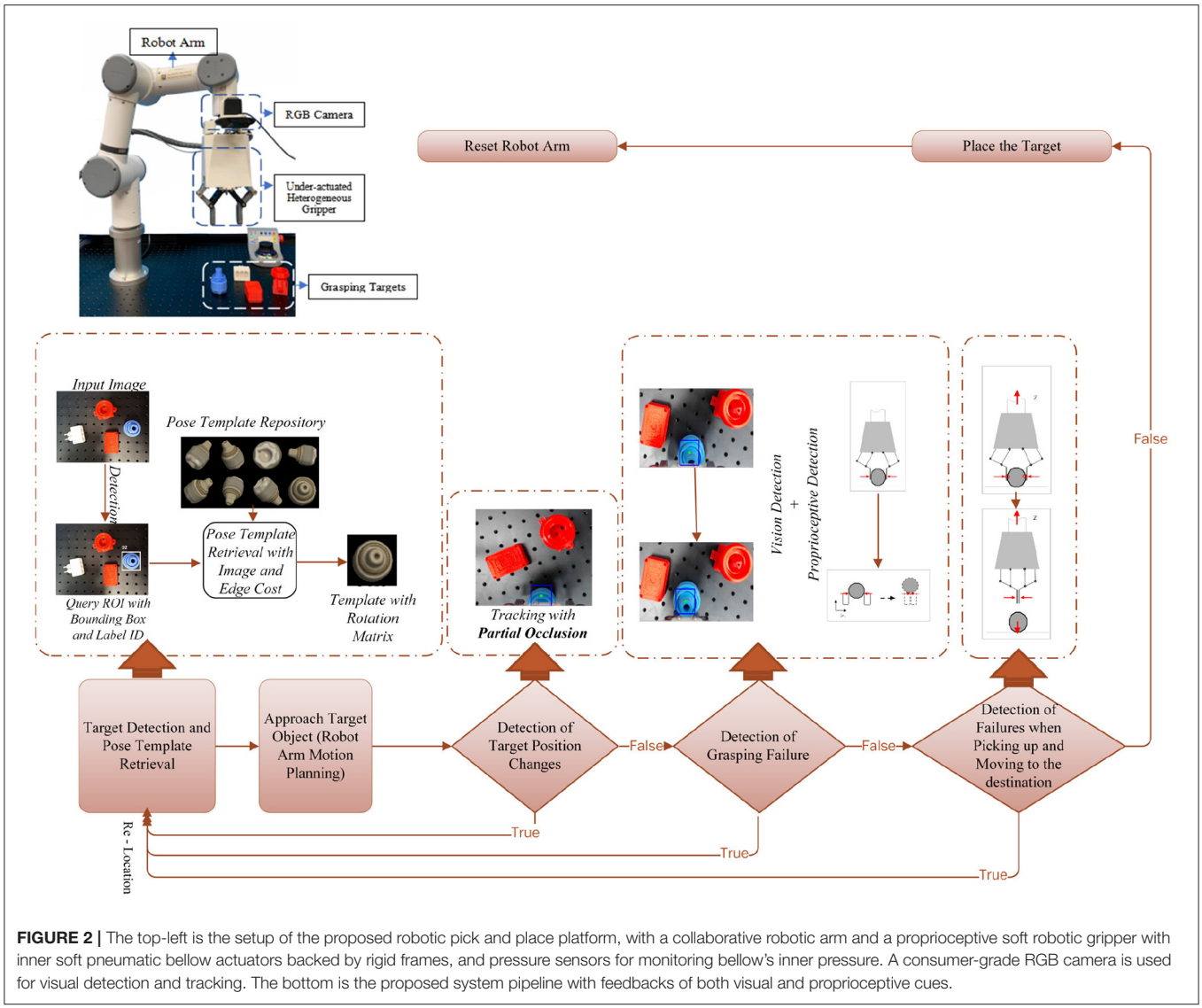


FIGURE 2 | The top-left is the setup of the proposed robotic pick and place platform, with a collaborative robotic arm and a proprioceptive soft robotic gripper with inner soft pneumatic bellow actuators backed by rigid frames, and pressure sensors for monitoring bellow's inner pressure. A consumer-grade RGB camera is used for visual detection and tracking. The bottom is the proposed system pipeline with feedbacks of both visual and proprioceptive cues.

trial is successful. The failure here usually results from internal disturbance, such as the inaccurate pose of the gripper in the former trial. Both visual and proprioceptive cues are utilized by observing whether the coordinate of the target remains the same and the force changes measured by inner pressure sensor have followed the common rules during the grasping trial. Then combined feedbacks will guide the determination of the system. If no failure happens, the system will step into the next phase. Otherwise, the system will timely go back to the very beginning phase. Compared with our previous work (Zhou et al., 2017) without timely failure detection in grasping, the combination of visual and proprioceptive information contributes to the effectiveness of failure reaction in our system. The third detection aims at checking picking failure based on the proprioceptive information. Proprioceptive cues can be sensitively observed from the embedded air-pressure sensor of our soft gripper. Thus, the soft gripper has its specific advantages in our case besides its compliance advantage in grasping. Through simple data processing, the contact force between the object and gripper can be estimated. Picking failure occurs when a sudden decrease in the estimated contact force is detected. In the final phase, if object picking succeeds, the target will be placed in the expected position and the robot arm will be reset.

3. METHODOLOGY

This section clarifies some technical details in our system. It can be divided into three parts. In the first part, we introduce how to automatically detect the target and estimate the pose of the target by the template retrieval. Besides the canonical image-based cost, we introduce the edge-based cost to improve the accuracy of object pose estimation. In the second part, we present the target tracking algorithm that can robustly work even with partial object occlusion. In the third part, we explore the details about the proprioceptive of our soft gripper in the failure detection and recovery system.

3.1. Object Detection and Pose Template Retrieval

In this part, we illustrate our algorithm for object detection and pose template retrieval. As illustrated in **Figure 3**, we implement the network described in Sundermeyer et al. (2018) to compute the image-based cost by first finetuning Single Shot MultiBox Detector (Liu et al., 2016) on synthetic images to help detect and label objects on the query image, and then reducing the pose estimation issue to pose template retrieval, in which we create a pose repository for each object by rendering clean images with different views and inner plane rotations. To retrieve the best pose template from this repository, here we innovatively combine not only the state-of-art work (Sundermeyer et al., 2018) with a deep neural network, but also a canonical edge-based cost (Shotton et al., 2008) to improve robustness. **Figure 3** shows how we combine these two cues for the pose template retrieval problem, while in the following paragraphs we will illustrate these two costs and the way we combine them for the pose template retrieval in detail.

3.1.1. Image-Based Cost

Through the supervised process of reconstructing the object's appearance in the RGB image while eliminating the influence of background clutter, occlusion, geometric, and color augmentation, Sundermeyer et al. (2018) output a descriptor that conveys the 3D orientation information. By looking up the descriptor codebook for poses in the repository, a cosine similarity cost is computed to measure the similarity between the query Region Of Interest (ROI) and the *i*th pose template in the repository:

$$C_i^{IMG} = -\frac{z_q^T z_i}{\|z_q^T\| \cdot \|z_i\|} \tag{1}$$

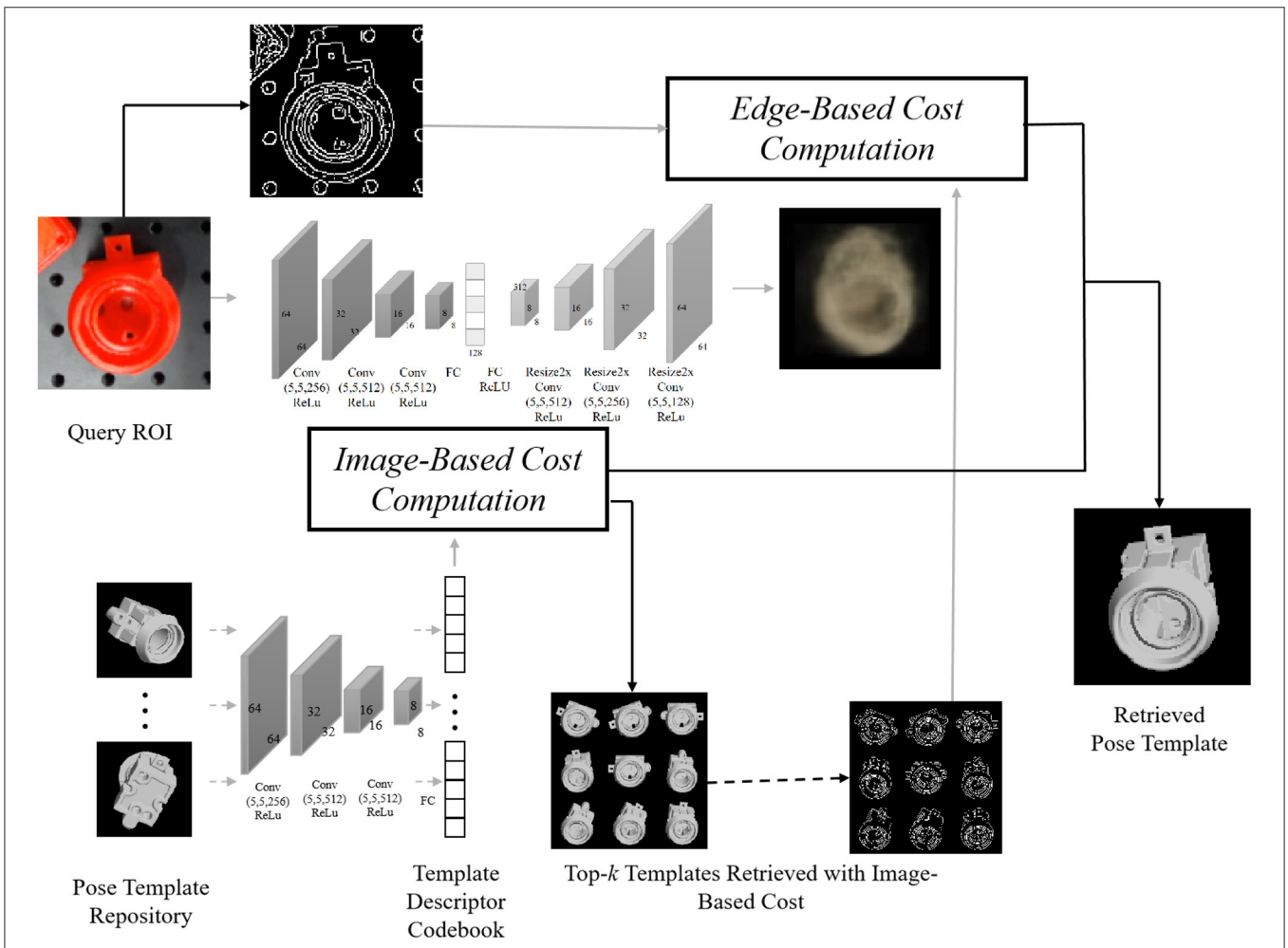


FIGURE 3 | The pipeline for pose template retrieval, where we first compute the image-based descriptor based on the reconstruction of the foreground model and use the image-based cost to select top-*k* template candidates. Among these *k* candidates, the edge-based cost is then computed with the corresponding edge map (see details in section 4.1), followed by a combination of two scores to re-rank. Off-line computation for templates is annotated by dash lines, and online computation by solid lines.

where z_q,$z_i \in R^{128}$ correspond to the computed descriptors for the query ROI and ith pose template in the repository, respectively. Here, we take a negative to ensure a smaller cost indicates a better matching.

3.1.2. Edge-Based Cost

We utilize oriented Chamfer distance (Shotton et al., 2008) to compute the edge-based cost. With a given edge map and the set of edge points T_i for the ith pose template, we define the nearest query edge point V(t) for $t \in T_i$ as:

$$V(t) = \underset{q \in Q}{\operatorname{argmin}} \|q - t\|_1 \tag{2}$$

where Q indicates the set of edge points from the query ROI, and $L1$ distance is used. So we evaluate the edge-based cost:

$$C_i^{EDGE} = \frac{1}{|T_i|} \sum_{t \in T_i} \|V(t) - t\|_1 + \lambda \|\phi(V(t)) - \phi(t)\| \tag{3}$$

where $|T_i|$ indicates the cardinality of the set T_i and $\phi(x)$ is the orientation of edge at edge point x. *lambda* is the weighting factor that balances the distance and orientation differences.

3.1.3. Enhanced Image-and-Edge Costs

Due to the gaps between synthetic training data and real test images in terms of environments and model precision, the image-based cost may fail to retrieve the correct pose reasonably, while the edge-based cost is robust under these changes. Thus, we first use image-based costs to provide top-k pose candidates. Then we use a weight parameter $\mu(0 \leq \mu \leq 1)$ to linearly combine both image and edge-based costs to re-rank these k candidates:

$$C_i = \mu C_i^{IMG} + (1 - \mu) C_i^{EDGE} \tag{4}$$

3.2. Visual Tracking Under Partial-Occlusion Circumstance

In this section, we address the case of continuously tracking the target even though partial occlusion occurs. We use correlation filters to model the appearance of the target and perform robust tracking via convolution. Recently, correlation-filters-based trackers (CFTs), which were widely used in recognition (Savvides et al., 2004) and detection (Bolme et al., 2009), have shown promising performance in object tracking. The CFTs estimate the target's position by correlation filters with different kinds of features. In the Fourier domain, the correlation score is computed by the element-wise multiplication between image features and the complex conjugate of the correlation filter (Bolme et al., 2010). Inverse fast Fourier transform (IFFT) is utilized to transform the correlation back to the spatial domain. The peak correlation score indicates the target's center.

A general illustration of the tracking method, which is feasible when partial occlusion exists, is shown in **Figure 4**. Let f denotes the feature of an image patch and g denotes the desired output, we can get the correlation filter in the Fourier domain (Bolme et al., 2010). The state of the target can be estimated by learning a discriminative correlation filter (DCF) h, which is trained by an image patch I of size $M \times N$ around the target. The tracker considers all circular shifts $f^l_{m.n}$,$(m, n) \in 0, ..., M - 1 \times 0, ..., N - 1$ as features of training patches for training correlation filters, where $l \in 1, ..., d$ is the dimension of features. The correlation filter h^l of each feature is built by minimizing a cost function as follows:

$$h^* = \underset{h}{\operatorname{argmin}} \sum_{m,n} \| \sum_{l=1}^{d} f^l_{m,n} \odot h^l - g(m, n) \|^2 \tag{5}$$

where $\odot$ symbol denotes circular correlation. All the training patches are selected from I by dense sampling. Equation (5) is a linear least square system that transforms tasks from the spatial domain into the frequency domain with a simple element-wise relationship. The Fourier transform of the input image, the filter, and the output can be represented by F^l, H^l, and $G_i, \overline{F^l}, \overline{F^l}$ represent the complex conjugation operations, and above minimization problem takes the form:

$$\min_{H^*} \sum_i \left| \overline{F}^l H^l - G_i \right|^2 \tag{6}$$

By solving for H^l, a closed-form expression is shown as:

$$H^l = \frac{\sum_i \overline{G_i} F^l}{\sum_i \overline{F^l} F^l} \tag{7}$$

To estimate the target's position in the frame t, a new patch z with size $M \times N$ will be cropped out according to the target's position in the frame $t - 1$. Based on the correlation filter, the response output is then computed and transformed back into the spatial domain by IFFT. The location of the maximum value in the response output indicates the shifted center of the target from frame $t - 1$ to frame t.

3.2.1. Tracking With Partial Occlusion

The algorithm performs well under scale variation and partial occlusion. The Peak-to-Sidelobe ratio, which measures the strength of a correlation peak, splits the response of the filter into the maximum value and the "side lobe" that consists of the rest of pixels in the region, including a small window (i.e., 11×11) around the peak. If the occlusion is detected, the tracker should attempt to hallucinate the target until it can be detected again. For occlusion solving, we divide the target into several patches and then compute the Peak-to-Sidelobe ratio of every response map. According to the maximum in the response map, the partially occluded target can be tracked robustly. The occlusion detection and solving techniques ensure the tracker to work robustly and reliably in robotic grasping.

3.3. Proprioceptive Grasp Failure Detection

Object picking and placing tasks are a series of contact involving forces, which cannot be easily monitored by vision. Vision can indicate to the robotic system the position of the target object, but it requires physical contact feedback to fast detect-response to dynamical changes and enable robust grasping. In this section,

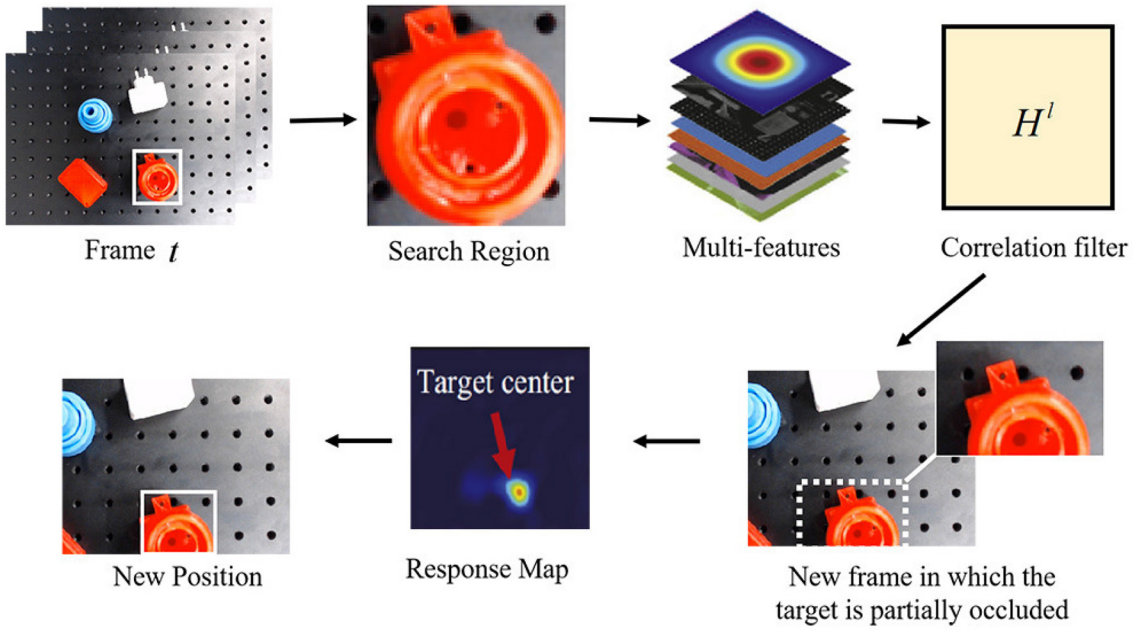


FIGURE 4 | Illustration of the proposed robust correlation-filters-based tracking method for **partial occlusion**.

based on the soft actuated rigid gripper developed in the previous work (Wang and Wang, 2020), we proposed a contact and contact loss monitoring method for the grasp failure detection in the pick-and-place task. **Figure 5** presents the prototype and mechanism of the soft actuated rigid gripper. The soft actuated rigid gripper was constructed with antagonistic bellows and plane six-bar linkages and is pneumatically operated with a simple control system. Two pressure sensors were used for monitoring bellows' inner pressure. We did not attach any traditional force or position sensors on the soft actuated rigid gripper but to leverage the pressure signal of the soft bellows actuators for estimate the joint movements and external contacts. In such configuration, this soft actuated rigid gripper is endowed with so-called proprioceptive capability.

3.3.1. Contact Force Estimation

The contact force at the fingertip was proposed to be estimated by a generalized momentum observer (Wang and Wang, 2020). The observer dynamics is given by

$$r = K_o\left(M(\theta)\dot{\theta} - \int_0^t ((F_a - k_a y)\frac{\partial y}{\partial \theta} + C(\theta,\dot{\theta}) - g(\theta) + r)ds\right) \tag{8}$$

where the monitoring signal r is observer output,K_o is observer gain. Displacement of the actuator y and link angle θ is a set of generalized coordinates to formulate the dynamic model of the gripper system. k_a is the axial stiffness of the actuator, which was theoretically and experimentally calibrated as a constant. The actuation force F_a is estimated by the measured pressures P_1 and P_2 of the active and passive bellow, that is $F_a = A \cdot (P_1 - P_2)$, where A is the effective active area of the air. $M(\theta)$ is the mass inertia, $C(\theta,\dot{\theta})$ is the centrifugal and Coriolis force, and g(θ) is the gravitational torques in the link joints. Detailed deduction of the momentum observer can be seen in Wang and Wang (2020). The contact force at the fingertip F_g can then be estimated via the observer output as

$$F_g = (\frac{\partial x_f}{\partial \theta})^{-1} r \tag{9}$$

where x_f is the displacement of the gripper finger.

3.3.2. Contact Detection

To detect the physical contact between gripper fingertips and the object, a contact detection function $cd(\cdot)$ can be introduced to map the estimated contact force $F_g(t)$ into the two classes *TRUE* or *FALSE*:

$$cd : F_g(t) \rightarrow \{TRUE, FALSE\}$$

Ideally, the binary classification is obtained by

$$cd(F_g(t)) = \begin{cases} TRUE, & \text{if} \quad F_g(t) \neq 0 \\ FALSE, & \text{if} \quad F_g(t) = 0 \end{cases} \tag{10}$$

Considering the error in measurement, modeling, and disturbances, in practice the monitoring signal $F_g(t) \neq 0$ even when no contact occurs. Thus, an appropriate threshold should be considered to obtain a robust contact detection function. Statistical observations of gripper finger open and close motion without grasping any objects, fingertip collision, or external disturbances for a sufficiently long time interval $[0, T]$ lead to a definition of $\mu_{max} = max\left\{|\mu(t)|, t \in [0,t]\right\}$. Considering a safe

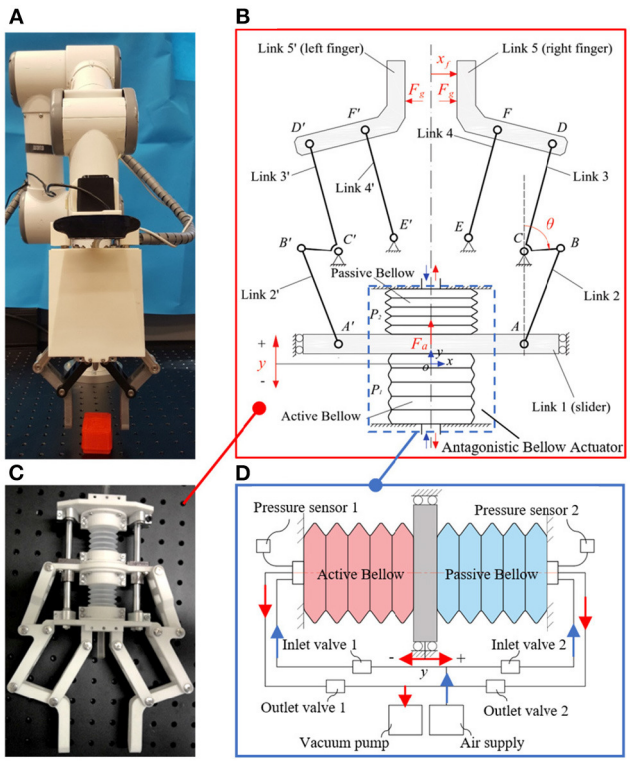


FIGURE 5 | Proprioceptive gripper with proprioceptive sensing capability: **(A)** prototypical setup of the proprioceptive gripper; **(B)** mechanism of the proprioceptive gripper; **(C)** internal gripper structure showing the bellows soft actuator; **(D)** schematic of the pneumatic bellow actuator.

margin $\varepsilon_{safe} > 0$, the contact detection function can be decided using a conservative threshold $\sigma = \mu_{max} + \varepsilon_{safe}$:

$$cd(F_g(t)) = \begin{cases} TRUE, & \text{if} \quad F_g(t) > \sigma \\ FALSE, & \text{if} \quad F_g(t) \leq \sigma \end{cases} \tag{11}$$

3.3.3. Contact Loss Detection

In case of sudden contact loss due to error in grasping pose configuration, external disturbance, or insufficient contact force, the gripper finger accelerates in the same direction as the grasping force applied to the surface of the object. Therefore, the contact force will suffer a rapid decrease. A binary function can be introduced to recognize contact loss by monitoring the changes in the contact force signal between two suitable time intervals $\triangle F_g(kT) = F_g[NT] - F_g[(N-k)T]$, where T is the sampling time and kT is the time interval. Similarly, considering the noise in the estimated contact force, a threshold $\triangle$>0 is used to decide the contact loss detection function $cld(\cdot)$

$$cld(\triangle F_g(kT)) = \begin{cases} TRUE, & \text{if} \quad \triangle F_g(kT) < \triangle \\ FALSE, & \text{if} \quad \triangle F_g(kT) \geq -\triangle \end{cases} \tag{12}$$

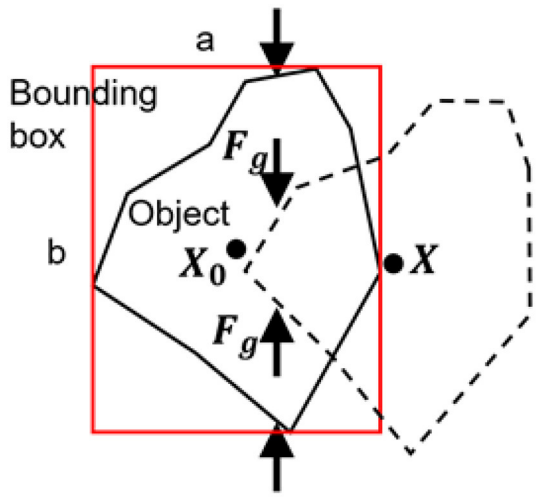


FIGURE 6 | Illustration of grasping failure in the second phase of detection.

3.4. Cooperative Work Between Visual Cues and Proprioception of Our Soft Gripper

The object detection and pose estimation algorithm contributes to an initial grasp plan with higher accuracy by considering the edge-based cost besides the image-based cost. Then the tracking algorithm provides the visual cues by robustly reporting the object's real-time position, even if the target is partially occluded in tracking. Systematic failures can be detected from unexpected position changes reflected by the visual cues. Furthermore, immediately after the failure was reported, visual cues can efficiently help the systematic recovery by real-time relocation and pose estimation of the target. Visual cues take effect in both the 1*st* phase (detection of position changes) and 2*nd* phase of detection (detection of grasping failure).

Meanwhile, the proprioception of our soft gripper contributes to the contact detection between the gripper and the object by contact force estimation with the internal air pressure sensor. In the 2*nd* phase of detection, see **Figure 6**, if grasping failure occurs, besides the object position changes reflected by the visual tracking algorithm, the sudden changes of contact force will simultaneously be reported by the internal air pressure. We combine the visual and proprioceptive signal for detection of grasping failure. Let us assume that the maximum contact force during grasping is F_{max} and after grasping is F_g, the object position before grasping is X_0, and the object position after grasping is X, and the side length of the rectangle bounding box are a and b. Intuitively, the object is being stably grasped if larger force F_g retains and X is close to X_0 after contact. Using maximum entropy principle, we predict whether contact loss occurs based on the visual or proprioceptive cues and then blend the prediction results for arbitration.

The probability μ_p of grasping failure predicted by proprioceptive cues can be formulated as

$$\mu_p = \frac{e^{-\alpha|F_g - F_{max}|^2}}{e^{-\alpha|F_g - F_{max}|^2} + e^{-\alpha F_g^2}} \tag{13}$$

where α is an adjustable and negative parameter. The probability μ_v of grasping failure predicted by visual cues can be formulated as

$$\mu_v = \begin{cases} \frac{e^{-\beta\|X - X_0\|^2}}{e^{-\beta(\sqrt{a^2+b^2} - \|X - X_0\|)^2} + e^{-\beta\|X - X_0\|^2}}, & \|X - X_0\|^2 < a^2 + b^2 \\ 1, & \|X - X_0\|^2 \geq a^2 + b^2 \end{cases} \tag{14}$$

where β is an adjustable and negative parameter.

To formulate a confident arbitration, a blending function can be implemented by

$$\mu^* = (1 - \lambda)\mu_p + \lambda\mu_v \tag{15}$$

where $\lambda \in [0, 1]$ is a blending factor that represents the confidence on the visual cues or proprioceptive cues for predicting grasping failure. Considering a threshold μ_0, the grasping failure can be detected by a binary classification.

$$cd(F_g, X) = \begin{cases} TRUE, & \text{if} \quad \mu^* > \mu_0 \\ FALSE, & \text{if} \quad \mu^* \leq \mu_0 \end{cases} \tag{16}$$

In the 3*rd* phase of detection, proprioceptive cues are utilized again to inspect the state of picking (see Equation 12). Failures may sometimes occur here because of insufficient grasping force. The detection result in this phase determines whether the internal air pressure needs to be increased.

4. EXPERIMENTAL VALIDATION

This section introduces experimental details to validate the outstanding performance of our system. The experimental setup is shown in **Figure 2**. A consumer-grade RGB camera (Logitech C920) is utilized for object detection and tracking. The proprioceptive robotic gripper provides proprioceptive cues for failure detection. They are both mounted to the end joint of a 6-DoF robot arm (E6, SANTIFICO Ltd.).

4.1. Validation of Accuracy Improvement in Object Pose Estimation After Introducing Edge-Based Cost

In experiments of this paper, we use the canonical Canny (1986) to compute edge maps for both pose templates in the repository (off-line computation) and detected query ROIs (online computation). For each object, we generate 3,240 pose templates by evenly sampling the unit sphere space and utilize the image-based cost to select $k = 20$ templates for further re-ranking. We set $\lambda = 10$ in edge-based cost and $\mu = 0.9$ for

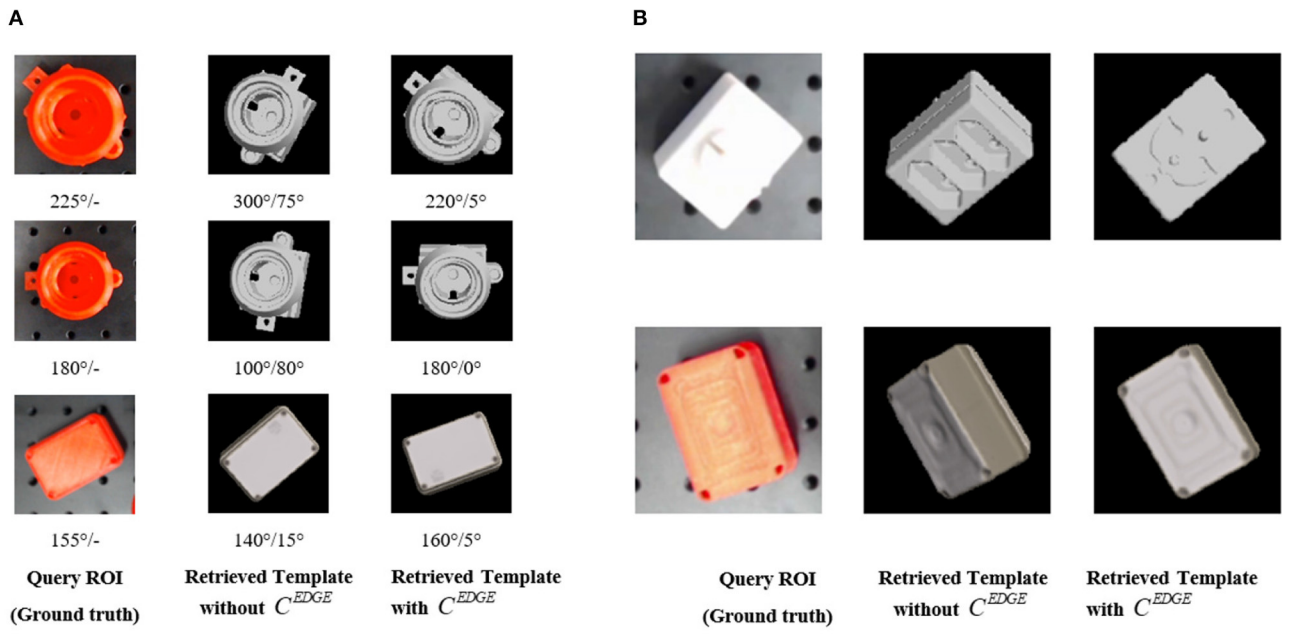


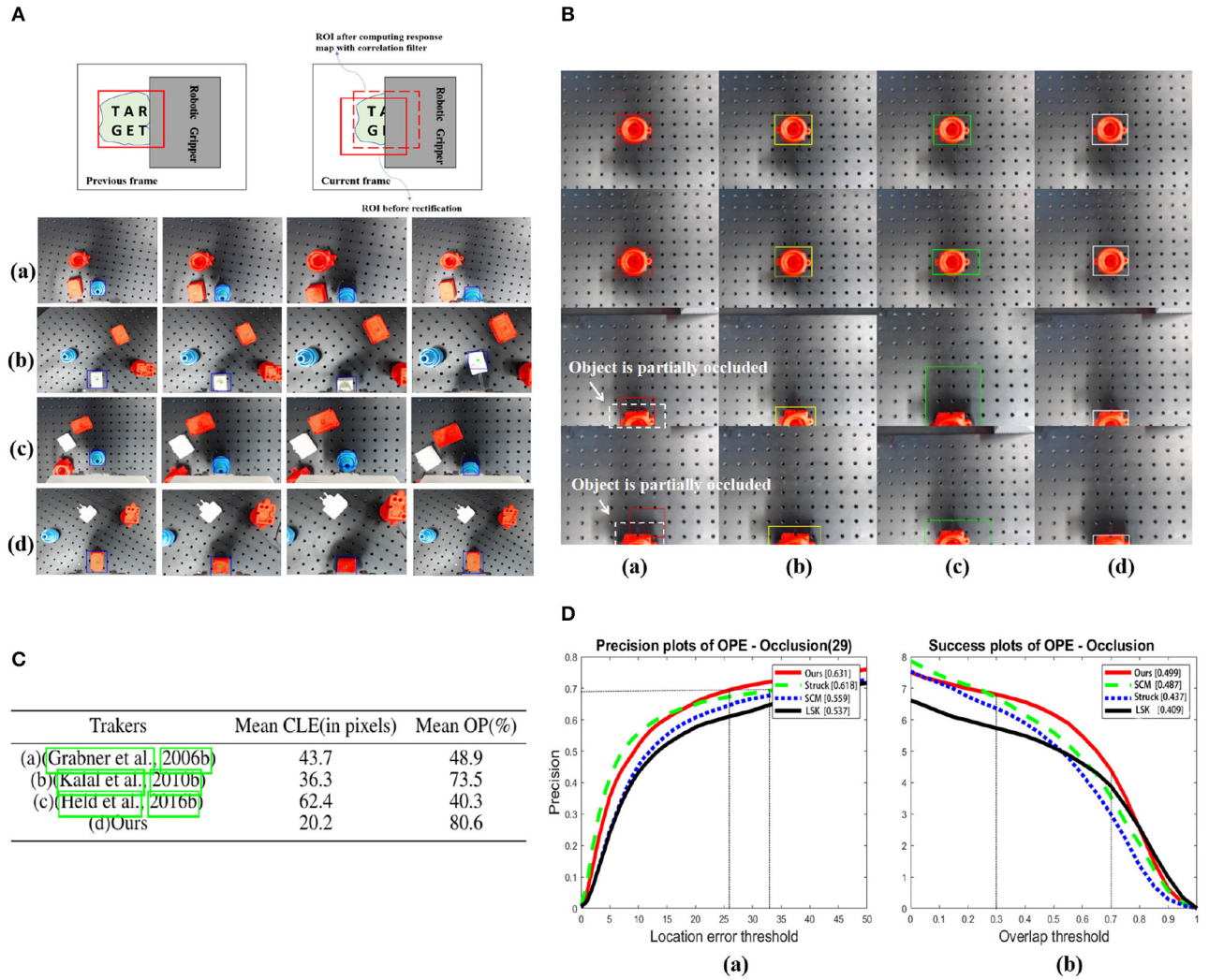
FIGURE 7 | (A) Quantitative comparisons that show the improvements brought by the introduction of edge-based cost. The number at the bottom of each row (before "/") illustrates the rotation angle (in degree) of protruding handles in ground-truth or the estimated pose, as well as (after "/") the difference between estimated and ground-truth value. The first column is the ground truth of the object pose. The second column is the pose estimated by the method in Sundermeyer et al. (2018), which only considers image-based cost. The third column is the estimated pose of our method considering both image-based cost and the edge-based cost. Our method helps improve the robustness and accuracy of the object pose estimation. **(B)** Qualitative comparisons that illustrate that the introduction of edge cue could help rectify the orientation and provide more accurate pose estimation results. Compared with the color cue, the edge cue is more robust under the variants of lighting conditions.

the enhanced image-and-edge cost. To illustrate the advantage of our re-ranking strategy with edge-based cost, we evaluate a model (see **Figure 7**) with protruding handles, which are crucial for gripping. The image-based cost alone fails to accurately evaluate the orientation of these handles, but the introduction of edge information improves robustness on these detailed but crucial parts.

Figure 7 presents examples with qualitative and quantitative comparisons between two settings that either combines edge information and re-ranking or not. To analyze the estimated result quantitatively, we compute the absolute difference (the error of pose estimation) of rotation angle between estimated and ground-truth pose for the handles referring to the axis perpendicular to the image plane. It is apparently validated that our method, which introduces the edge cost besides the image cost, have advantages over the state-of-art work (Sundermeyer et al., 2018). More reasonable templates are retrieved with the aid of edge-based costs.

4.2. Validation of Object Tracking Under Partial Occlusion

Object tracking plays an important role in three phases of detection in our system. However, partial occlusion, which results from the body part of the gripper or external disturbance, may



Trakers	Mean CLE(in pixels)	Mean OP(%)
(a)(Grabner et al., 2006b)	43.7	48.9
(b)(Kalal et al., 2010b)	36.3	73.5
(c)(Held et al., 2016b)	62.4	40.3
(d)Ours	20.2	80.6

FIGURE 8 | (A) Robust object tracking under partial occlusion in different scenarios. *(a)* When no accidental failure occurs. *(b)* Unexpected position changes of the object. *(c)* Grasping failure. *(d)* Picking up failure. Letters "T," "A," "R," "G," "E," and "T" are marked on the target to indicate different portions of it. The blue bounding box indicates the position of the target in the camera view. **(B)** Comparison of tracking methods introduced in *(a)* (Grabner et al., 2006), *(b)* (Kalal et al., 2010), *(c)* (Held et al., 2016) with *(d)* ours while the object is partially occluded. The white dashed boxes indicate the object with partial occlusion and bounding boxes in different colors correspond to the results of each tracking method. **(C)** Comparisons between four tracking methods in the same sequence. The mean CLE (in pixels, the lower the better) and OP (%, the higher the better) are presented (whent_0 = 0.5). The best results for the experiments are shown in the bold format. **(D)** Performance evaluation of the proposed method using precision plot and success plot of OPE (One-Pass Evaluation) for sequences having occlusion in OBT-50. In *(a)*, to achieve the same precision at 0.7 or more, our method demands less location error than others. In *(b)*, the partial occlusion(overlap) is usually 30–70% when the gripper approaching to the bottom to grasp the object, and in this overlap interval, our method achieves higher success rate than others.

accidentally occur during robotic grasping. As shown in **Figure 2**, the target is partially occluded by the body part of the gripper. To provide reliable visual guidance for decision-making in real time, we introduce the tracking method that can work robustly even when occlusion exists.

4.2.1. Object Tracking With Partial Occlusion

With numerous systematic tests, the robustness of our tracking algorithm has been obviously reflected, especially when the accidental failures occur. In **Figure 8A**, we visually presented several tracking examples under different circumstances of our grasping system. Although the object is partially occluded by the body part of our robotic gripper when failures occur, the visual tracking algorithm still robustly provides visual cues to assist the failure recovery of the grasping system.

To demonstrate the advantages of our method over others, we designed the following experiments. With the same experimental setup illustrated in **Figure 2**, when the target is partially occluded in the camera view, we compare the results of three typical visual tracking algorithms (Grabner et al., 2006; Kalal et al., 2010; Held et al., 2016) in robotic applications with our results. The results are shown in **Figure 8B**.

To quantitatively evaluate the performance of each tracker, we adopt the evaluation protocol described in Danelljan et al. (2014a,b). (1) Center location error (CLE), which is the average Euclidian distance between the estimated center location of the target and ground truth, and (2) OP, which is the percentage of frames where overlap score is larger than a given threshold t_0 (e.g., $t_0 = 0.5$). The score is defined as:

$$score = \frac{area(R_T \bigcap R_G)}{area(R_T \bigcup R_G)} \tag{17}$$

where R_G and R_T are the region of tracking results and ground truth, and $\bigcap$ and $\bigcup$ are the intersection and union operations.

We have evaluated each tracker on 21 video sequences, which is recorded in our real experimental tests and the partial occlusion exists. For each video sequence, we run 15 times for each tracker and record the mean values of CLE in pixels and OP (%). **Figure 8C** quantitatively reports the comparative results of each tracking methods. Both the lowest value of mean CLE and the highest value of mean OP obviously indicate that ours is superior to others. Even the latest (Held et al., 2016), whose performance is well-acknowledged in computer vision benchmarks, underperforms when the target is partially occluded. The value of mean OP in our algorithm has sufficiently satisfied the requirement of robust tracking under partial occlusion.

To further validate the tracking performance under partial occlusion, our method is evaluated on the 29 sequences with the

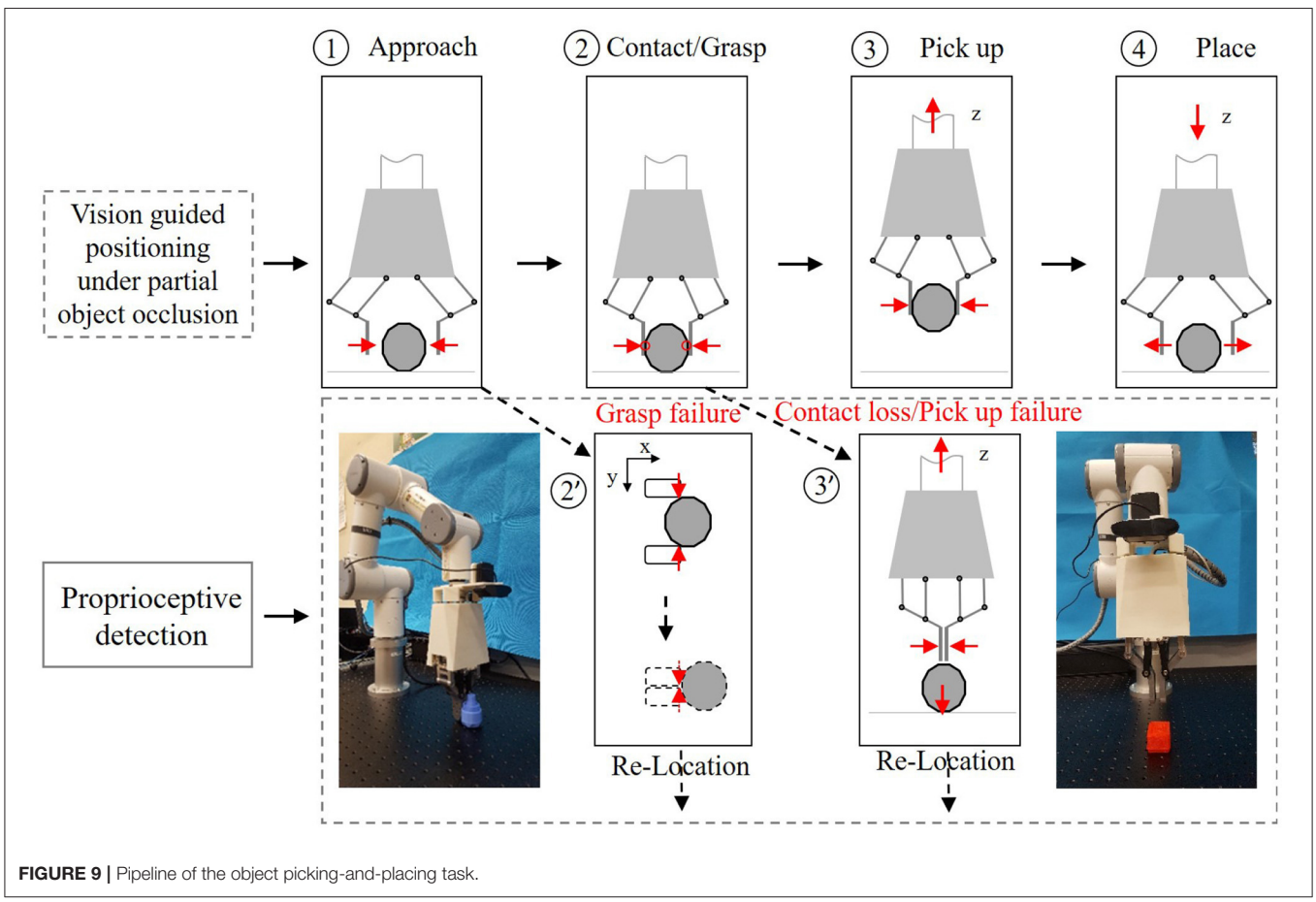


FIGURE 9 | Pipeline of the object picking-and-placing task.

partial occlusion in the OTB-50 benchmark (Wu et al., 2013), in which attributes are fully annotated. We compare our method with the reported TOP 3 tracking algorithms in the benchmark using one-pass evaluation (OPE). The OPE uses the ground truth object location in the first frame and evaluates the tracker based on the average *precision score* or *success rate*.The former is the ratio of successful frames whose OR is larger than a given threshold to the total frames in a sequence, whereas the later is

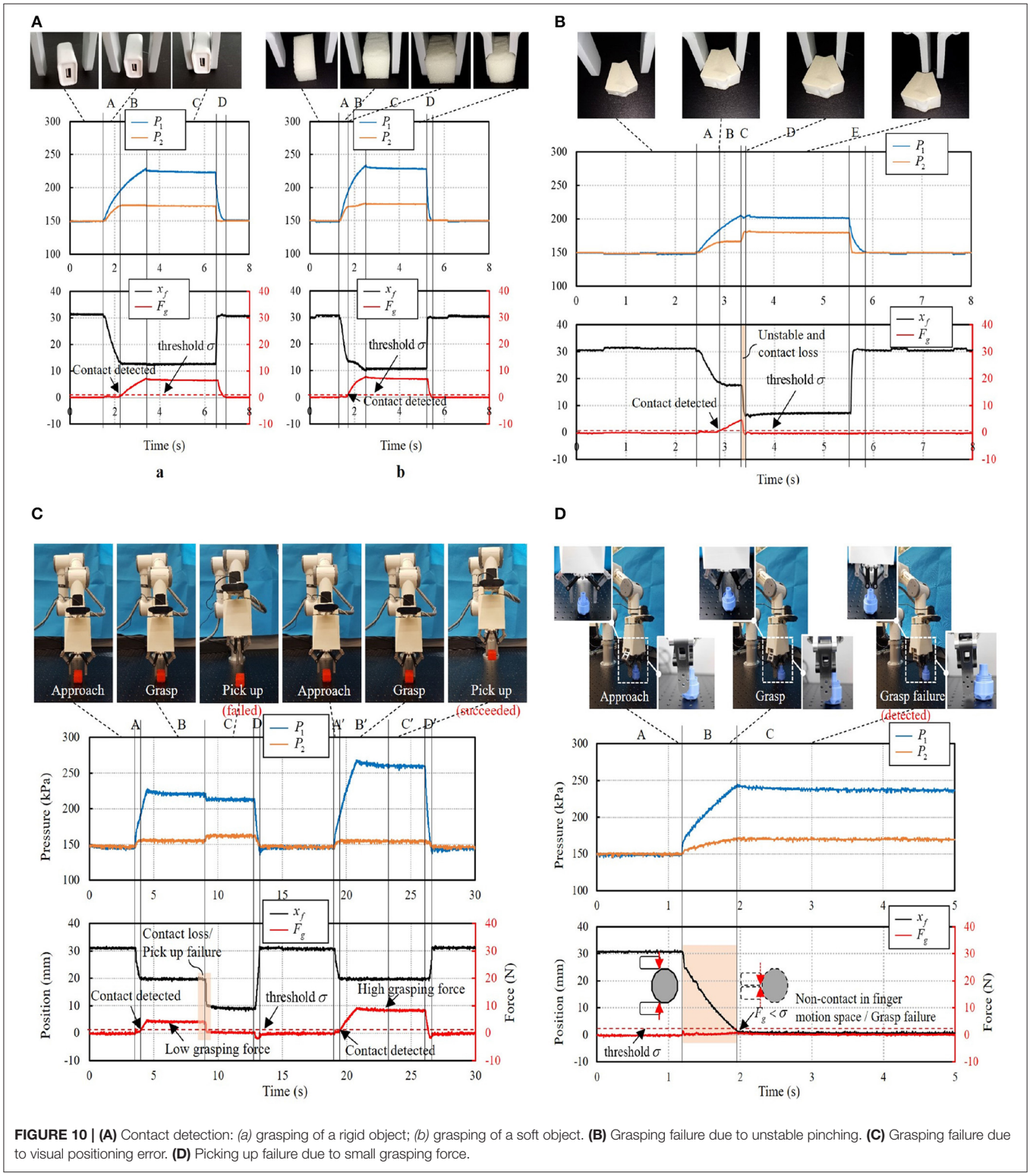


FIGURE 10 | (A) Contact detection: *(a)* grasping of a rigid object; *(b)* grasping of a soft object. **(B)** Grasping failure due to unstable pinching. **(C)** Grasping failure due to visual positioning error. **(D)** Picking up failure due to small grasping force.

the percentage of frames whose CLE is less than a given threshold distance of the ground truth. Further, these success and precision curves are averaged over all the sequences to obtain the overall success and precision plots, respectively. The plots of OPE for the 4 trackers averaging over the OTB-50 sequences having occlusion are shown as **Figure 8D**. In (a), to achieve the same precision at 0.7 or more, our method demands less location error than others. In (b), the partial occlusion (overlap) is usually 30–70% when the gripper approaching to the bottom to grasp the object, and in this overlap interval, our method achieves higher success rate than others.

4.3. Validation of Proprioceptive Grasping

The pipeline of object picking and placing can be divided into four phases: (1) approach; (2) contact and grasp; (3) pick up; (4) place, as illustrated in **Figure 9**. First, the two-finger gripper approaches the target object with suitable pose configuration guided by vision. The two fingers then grasp the object with commanded grasping force. After that, the robot arm will pick up and place the object. It is very common in practice that the system may suffer task failure due to the disturbance in the environment, including visual position error or unstable interaction force. Proprioceptive grasp experiments were conducted using the two-finger gripper, including contact, grasping failure, and picking up failure detection.

4.3.1. Contact Detection

We validate contact detection on both rigid and soft objects. **Figure 10A** presents the recorded data of grasping a rigid object (**Figure 10Aa**) and a soft object (**Figure 10Ab**), reporting the pressures of the actuator $P_1(t)$,$P_2(t)$, finger position $x_f(t)$, and estimated grasping force $F_g(t)$. A constant threshold $\sigma = 0.8N$ was set to trigger the contact detection signal. The system was capable of rapidly detecting the collision with the objects during the grasping. After the contact was detected, the finger motion would stop when grasping a rigid object while the fingers would keep its movement when grasping a soft object as can be seen in phase B in **Figure 10A**.

4.3.2. Grasping Failure Detection

Figure 10B shows a case when grasping an irregular plane object. The object was successfully pinched at time $t = 2.9$ s when $F_g(2.9) > 0.8$ N. But due to the unstable grasp, the object was popped up suddenly at time $t = 3.3$ s. Contact loss was then detected with $\Delta F_g(kT) < -\Delta(k = 3, \Delta = 0.75N)$ and a grasping failure was recognized.

Figure 10C demonstrates another grasping of a bolt part with a cylinder surface. Due to inaccurate object positioning from the visual result, the object was slightly squeezed out from the two fingers against the cylinder surface. During the fingertip closing motion, non-contact event was triggered as the monitoring contact force $F_g(t)$ kept smaller than the threshold σ ($\sigma = 0.8N$). In this case, grasping failure was recognized with finger movement approaching the collision point ($x_f = 0mm$). As grasping "null" was detected, the robot arm stopped the picking up movement and instead to the relocation of the bolt part via the vision system.

4.3.3. Picking Up Failure Detection

As shown in **Figure 10D**, the robot system was commanded to grasp a heavy cuboid part and pick it to the target place. From the vision result, no indication can be provided for how large the grasping force should be. Thus, the system commanded a small grasping force ($F_g = 4N$) and it succeeded in grasping the cuboid object. But it failed in the first trail of picking up the object due to insufficient grasping force. The monitoring contact force suffered a sudden decrease when the object slipped off from the two fingers. Contact loss was then detected and recognized as picking up failure with $\Delta F_g(3T) < -0.75N$). The robot arm stopped the picking up movement and relocation of the cuboid part proceeded. After relocating the cuboid part, the gripper was commanded with a larger grasping force ($F_g = 8N$) and succeeded in picking up the cuboid part in the second trial. In case it failed again, the aforementioned process may be continued until placing the cuboid part.

4.4. Validation of Efficiency Improvement After Failure Detection and Recovery

We have designed the experimental tests to prove our failure detection and recovery system has improved the efficiency of robotic grasping (see **Figure 11**). With the same setup (as shown in **Figure 2**) and the same initial grasp planning (target detection and template retrieval) as our pipeline, the compared grasping system ignored the real-time failure detection and recovery, and

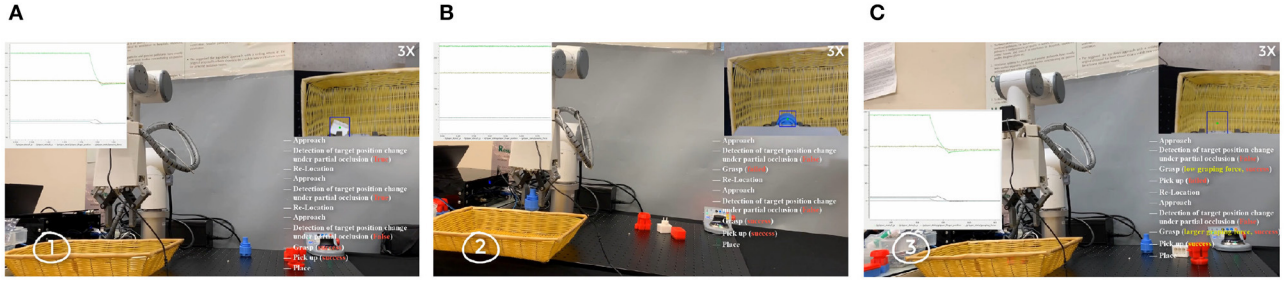


FIGURE 11 | Experiments of the proposed three-phase failure detection in the real-time pick-and-place tasks. **(A)** Detection of target position changes. **(B)** Detection of grasping failure. **(C)** Detection of failures when picking up and moving to the destination.

no matter what failure occurs, the robot arm will complete the pick-and-place operation. If the former grasping is found failed by the worker or other assistive means, a new grasping needs to be planned for another complete pick-and-place operation until the grasping is finally successful.

When accidental failures occur in the grasping, we separately recorded the time–cost (the time from the 1^{st} grasp planning to the final successful object placing) of successfully grasping the object in **Figure 9** with the pipeline without failure detection and recovery and ours in **Figure 2**. For each kind of failure, we separately did 20 tests in each pipeline. The average time–cost is 42.75 and 60.15 s separately for our pipeline and the pipeline without the failure detection and recovery. In our experiments, the average improvement of systematic efficiency is 40.7%.

5. CONCLUSION AND FUTURE WORK

This paper presents an approach for effectively handling failures in the robotic pick and place task by combining multimodal cues under partial occlusion. We achieve more accurate pose estimation of a known object by considering the edge-based cost besides the image-based cost. Robust object tracking method is proposed to work even when the object is partially occluded and achieve mean OP up to 80%. Meanwhile, we take advantage of our proprioceptive soft gripper for the contact and contact loss detection by analyzing internal pressure signals of our gripper. With the combination of visual cues under partial occlusion and proprioceptive cues from our soft gripper, our system can effectively detect and recover from different failures in the entire procedure of robotic pick and place tasks.

To improve the accuracy of pose estimation, we introduced the edge-based cost besides the image-based cost. Meanwhile, a correlation-filter-based tracking approach is proposed to guaranteed the robustness of the grasping system even partial occlusion exists, especially when detecting and recovering from the failures. Yet, proprioception of our soft gripper is proved to be an effective complement to vision in physical interaction, facilitating the system to fast detect-response to dynamic disturbances, such as grasping failure and picking up failure. Experiments have validated the robustness and accuracy of our approaches.

In future work, more varieties of grasping targets will be explored, for example, the jelly-like objects that are non-rigid or dynamic objects. These are both potential targets in real applications. A more precise and closed collaboration of vision and proprioceptive cues will be required for this kind of grasping task. Meanwhile, the problem of target pose estimation is significant for deciding the gripper's pose in real-time robotic grasping. A more flexible and simplified method will be considered to determine the pose of the target by simply moving the camera to a specific position in 3D space and observing the static target in different camera views.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author/s.

AUTHOR CONTRIBUTIONS

FZ developed the visual tracking techniques even with partial occlusion, completed corresponding validation, and wrote the first draft of the manuscript. LW extracted proprioceptive data and perform the statistical analysis. YW contributed to the object recognition and pose estimation in grasping. ZW and WW contributed to the conception and design of the experiments. All authors contributed to manuscript revision, and read and approved the submitted version.

FUNDING

This work was jointly supported by NSFC Grant 51975268, Hong Kong ITF Grant ITS/457/17FP, Hong Kong ITF Grant ITS/305/19FP, SUSTECH-AISONO Joint Lab Grant, and SUSTECH Edcucation Endowment.

REFERENCES

Bolme, D. S., Beveridge, J. R., Draper, B. A., and Lui, Y. M. (2010). "Visual object tracking using adaptive correlation filters," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (San Francisco, CA), 2544–2550. doi: 10.1109/CVPR.2010.5539960

Bolme, D. S., Lui, Y. M., Draper, B. A., and Beveridge, J. R. (2009). "Simple real-time human detection using a single correlation filter," in *2009 Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance* (Snowbird, UT), 1–8. doi: 10.1109/PETS-WINTER.2009.5399555

Canny, J. (1986). A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* 679–698. doi: 10.1109/TPAMI.1986.4767851

Coleman, D., Sucan, I., Chitta, S., and Correll, N. (2014). Reducing the barrier to entry of complex robotic software: a moveit! case study. *J. Softw. Eng. Robot.* 5, 3–16.

Cowan, N. J., Weingarten, J. D., and Koditschek, D. E. (2002). Visual servoing via navigation functions. *IEEE Trans. Robot. Autom.* 18, 521–533. doi: 10.1109/TRA.2002.802202

Danelljan, M., Häger, G., Khan, F., and Felsberg, M. (2014a). "Accurate scale estimation for robust visual tracking," in *British Machine Vision Conference* (Nottingham: BMVA Press). doi: 10.5244/C.28.65

Danelljan, M., Shahbaz Khan, F., Felsberg, M., and Van de Weijer, J. (2014b). "Adaptive color attributes for real-time visual tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Columbus, OH), 1090–1097. doi: 10.1109/CVPR.2014.143

Grabner, H., Grabner, M., and Bischof, H. (2006). "Real-time tracking via on-line boosting," in *BMVC, Vol. 1* (Edinburgh: Citeseer). doi: 10.5244/C.20.6

Held, D., Thrun, S., and Savarese, S. (2016). "Learning to track at 100 fps with deep regression networks," in *European Conference on Computer Vision* (Amsterdam: Springer), 749–765. doi: 10.1007/978-3-319-46448-0_45

Kalal, Z., Mikolajczyk, K., and Matas, J. (2010). "Forward-backward error: automatic detection of tracking failures," in *2010 20th International Conference on Pattern Recognition* (Istanbul), 2756–2759. doi: 10.1109/ICPR.2010.675

Kragic, D., and Christensen, H. I. (2002). Survey on visual servoing for manipulation. *Comput. Vis. Active Percept. Lab. Fiskartorpsv* 15:2002.

Kumra, S., and Kanan, C. (2017). "Robotic grasp detection using deep convolutional neural networks," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Vancouver, BC), 769–776. doi: 10.1109/IROS.2017.8202237

Li, B., Cao, H., Qu, Z., Hu, Y., Wang, Z., and Liang, Z. (2020). Event-based robotic grasping detection with neuromorphic vision sensor and event-stream dataset. *arXiv preprint arXiv:2004.13652*. doi: 10.3389/fnbot.2020.00051

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., et al. (2016). "SSD: Single shot multibox detector," in *European Conference on Computer Vision* (Amsterdam: Springer), 21–37. doi: 10.1007/978-3-319-46448-0_2

Savvides, M., Kumar, B. V., and Khosla, P. K. (2004). "Cancelable biometric filters for face recognition," in *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004, Vol. 3* (Cambridge), 922–925. doi: 10.1109/ICPR.2004.1334679

Shotton, J., Blake, A., and Cipolla, R. (2008). Multiscale categorical object recognition using contour fragments. *IEEE Trans. Pattern Anal. Mach. Intell.* 30, 1270–1281. doi: 10.1109/TPAMI.2007.70772

Su, Y., Fang, Z., Zhu, W., Sun, X., Zhu, Y., Wang, H., et al. (2020). A high-payload proprioceptive hybrid robotic gripper with soft origamic actuators. *IEEE Robot. Autom. Lett.* 5, 3003–3010. doi: 10.1109/LRA.2020.2974438

Sundermeyer, M., Marton, Z.-C., Durner, M., Brucker, M., and Triebel, R. (2018). "Implicit 3D orientation learning for 6D object detection from RGB images," in *Proceedings of the European Conference on Computer Vision (ECCV)* (Munich), 699–715. doi: 10.1007/978-3-030-01231-1_43

Wang, L., and Wang, Z. (2020). Mechanoreception for soft robots via intuitive body cues. *Soft Robot.* 7, 198–217. doi: 10.1089/soro.2018.0135

Wu, Y., Lim, J., and Yang, M.-H. (2013). "Online object tracking: a benchmark," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Portland, OR), 2411–2418. doi: 10.1109/CVPR.2013.312

Zhou, J., Chen, S., and Wang, Z. (2017). A soft-robotic gripper with enhanced object adaptation and grasping reliability. *IEEE Robot. Autom. Lett.* 2, 2287–2293. doi: 10.1109/LRA.2017.2716445

Zhou, J., Chen, Y., Chen, X., Wang, Z., Li, Y., and Liu, Y. (2020). A proprioceptive bellows (pb) actuator with position feedback and force estimation. *IEEE Robot. Autom. Lett.* 5, 1867–1874. doi: 10.1109/LRA.2020.2969920

Zito, C., Ortenzi, V., Adjigble, M., Kopicki, M., Stolkin, R., and Wyatt, J. L. (2019). Hypothesis-based belief planning for dexterous grasping. *arXiv preprint arXiv:1903.05517*.

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

frontiers in Neurorobotics

ORIGINAL RESEARCH
published: 19 March 2021
doi: 10.3389/fnbot.2021.648374

Robot Communication: Network Traffic Classification Based on Deep Neural Network

Mengmeng Ge, Xiangzhan Yu and Likun Liu*

School of Cyberspace Science, Harbin Institute of Technology, Harbin, China

With the rapid popularization of robots, the risks brought by robot communication have also attracted the attention of researchers. Because current traffic classification methods based on plaintext cannot classify encrypted traffic, other methods based on statistical analysis require manual extraction of features. This paper proposes (i) a traffic classification framework based on a capsule neural network. This method has a multilayer neural network that can automatically learn the characteristics of the data stream. It uses capsule vectors instead of a single scalar input to effectively classify encrypted network traffic. (ii) For different network structures, a classification network structure combining convolution neural network and long short-term memory network is proposed. This structure has the characteristics of learning network traffic time and space characteristics. Experimental results show that the network model can classify encrypted traffic and does not require manual feature extraction. And on the basis of the previous tool, the recognition accuracy rate has increased by 8%

Keywords: traffic classification, capsule neural network, encrypted traffic, network security, deep learning

OPEN ACCESS

Edited by:
Zhaoquan Gu,
Guangzhou University, China

Reviewed by:
Tingting Chen,
California State Polytechnic University, Pomona, United States
Longfei Wu,
Fayetteville State University, United States
Aiping Li,
National University of Defense Technology, China

****Correspondence:***
Xiangzhan Yu
yxz@hit.edu.cn

Received: *31 December 2020*
Accepted: *18 February 2021*
Published: *19 March 2021*

Citation:
Ge M, Yu X and Liu L (2021) Robot Communication: Network Traffic Classification Based on Deep Neural Network. Front. Neurorobot. 15:648374. doi: 10.3389/fnbot.2021.648374

INTRODUCTION

With the rapid development of technology, humanoid robots can do more things on behalf of people, such as helping people guide paths, serving coffee, and turning on lights. While humanoid robots liberate people's labor, there are also some risks of security and privacy leakage in these processes.

Robots need to interact with people or server commands when they are working (Gleeson et al., 2013; Mavridis, 2015). When robots and people interact through voice, everyone can hear the commands issued by people. When people want to hide the behavior and content of the commands, people can Use codes instead, such as a cough that means a command to turn on the light. This is the easiest way to hide the content of communication between humans and robots. When the server communicates with the robot, it is impossible for the server to cough and issue a command like a human (Su et al., 2020). He will put the control command in the network message and send it to the robot in a specific protocol format. When a stranger repeatedly observes the behavior of coughing, the robot will light up. He speculates that the coughing behavior may correspond to the command to turn on the light. Therefore, by observing the communication process between the control server and the robot in the network, and through learning and training, the communication protocol between the server and the robot can be identified, and further, the command line in the communication process between the server and the robot can be inferred as the type (Kanda et al., 2010). In this article, we have studied the protocol identification of network messages, which can identify the type and protocol of network communication traffic, which is of great significance to

the discovery of malicious network attack traffic in the communication process of humanoid robots. It can also be used for web traffic detection (Tian et al., 2020) and IoT traffic detection (Shafiq et al., 2020).

The development of traffic classification technology has gone through three stages: port-based, payload-based, and flow-based statistical characteristics. Port-based classification methods infer the types of mobile services or applications by assuming that most applications always use "well-known" TCP or UDP port numbers (Li et al., 2000; Hjelmvik and John, 2009; Wang et al., 2017). However, the emergence of port masquerading, random ports, and tunneling technologies quickly lost these methods Effectiveness. The payload-based method, that is, DPI (Deep Packet Inspection) (Moore and Papagiannaki, 2005; Finsterbusch et al., 2013; Wang et al., 2019) technology cannot handle encrypted traffic because it needs to match the content of the data packet and has a high computational overhead (Madhukar and Williamson, 2006). In order to try to solve the problem of encrypted traffic identification, a flow-based method has emerged, which usually relies on statistics or time series features and uses machine learning (Zuev and Moore, 2005; Liu et al., 2007; Fan and Liu, 2017; Shafiq et al., 2018) algorithms, such as Naive Bayes, Support Vector Machine (Li et al., 2007; Yuan et al., 2010; Groleat et al., 2012; Ebrahimi et al., 2017), Decision tree, random forest (Siahaan et al., 2019), k nearest neighbor (KNN)(Este et al., 2009; Wu et al., 2015; Sun et al., 2018). In addition, some statistical models, such as Gaussian Mixture Model (Alizadeh et al., 2015; Kornycky et al., 2016; Pacheco et al., 2018) and Hidden Markov Model (Yin et al., 2012), are used to identify and classify encrypted traffic.

Although classic machine learning methods can solve many problems that cannot be solved by methods based on ports and payloads, it still has some limitations: (1) It is difficult to obtain manually extracted traffic characteristics, and these characteristics always depend on domain experts' experience. Therefore, it is impossible to automatically extract and select features, which will cause great uncertainty and confusion in classic machine learning methods when ML is applied to mobile service traffic classification. (2) Flow characteristics are easily outdated quickly and need to be constantly updated. (3) How to combine a large, easily accessible unlabeled data set with some expensive labeled data sets for traffic classification to reduce the need for labeled data is a very critical research topic. (4) For traffic classification tasks, category imbalance is not a small problem. However, current data enhancement methods cannot accurately generate samples as close to the original data distribution as possible.

Unlike most traditional machine learning algorithms, deep learning (Gu et al., 2020) can perform automatic feature extraction without manual intervention. This paper uses the algorithm based on the capsule convolutional neural network (Vinayakumar et al., 2017; Rezaei and Liu, 2019) and the self-attention LSTM neural network to identify the encrypted network traffic (Fu et al., 2016; Si et al., 2019). The results show that this method does not require manual feature extraction and has excellent classification effects.

RELATED BACKGROUND CONTENT

Capsule Neural Network

Convolutional Neural Networks (CNN) have good image recognition performance, but they still have some shortcomings. When the photos are crowded and blurred, the classification effect will worsen, and the output of the model does not respond well to small changes in the input. The Capsule Network (CapsNet) (Xiang et al., 2018) solves some of the traditional neural convolutional network problems because the capsule network is composed of directional neuron groups, capsules instead of neurons. The traditional training feature of each neuron, learned which area in the spatial feature is not fixed, it is entirely random, but in the capsule network, each neuron group learns a fixed area in a certain area in the picture Features, such as the eyes and nose of a human face. The capsule network is also composed of multiple layers. As shown in **Figure 1A**, the vector capsule is at the bottom of the network (Zhu et al., 2019). The capsule network (Deng et al., 2018) also has a perceptual domain, just like the traditional CNN. However, for each vector capsule, their perceptual domain is a fixed part of the spatial feature, and they only learn the features of that region. During the training process, the parameters are modified continuously to improve the accuracy of classification and recognition. Further, some small capsules will be gathered into large capsules, called routing capsules, to learn more extensive spatial features. For example, vector capsules identify network traffic packet space features, and routing capsules identify the space of multiple network traffic packets. And then, the characteristics of the entire flow or stream are recognized.

Anti-fragility

The capsule network output is not the individual value output by the traditional artificial neuron but the output vector. The parameters of the dynamic combination of different vector capsules in the routing can be calculated through its unique dynamic routing mechanism, which can be trained for different perspectives.

Robustness

Redundant homomorphism can solve complex problems. Many small features will be trained by one Capsule. Although the parameters between capsules are independent of each other, many capsules will train similar substructures. Simultaneously, the high-level capsule can learn the relationship between the various structures of the bottom-level capsule. Even though the image to be recognized becomes blurred or shifted and other perspective changes, it can be correctly recognized through this redundant network structure.

Interpretability

It is possible to know what each capsule is responsible for training and which sub-structural features each capsule recognizes so that each parameter of the neural network is no longer a complete black box.

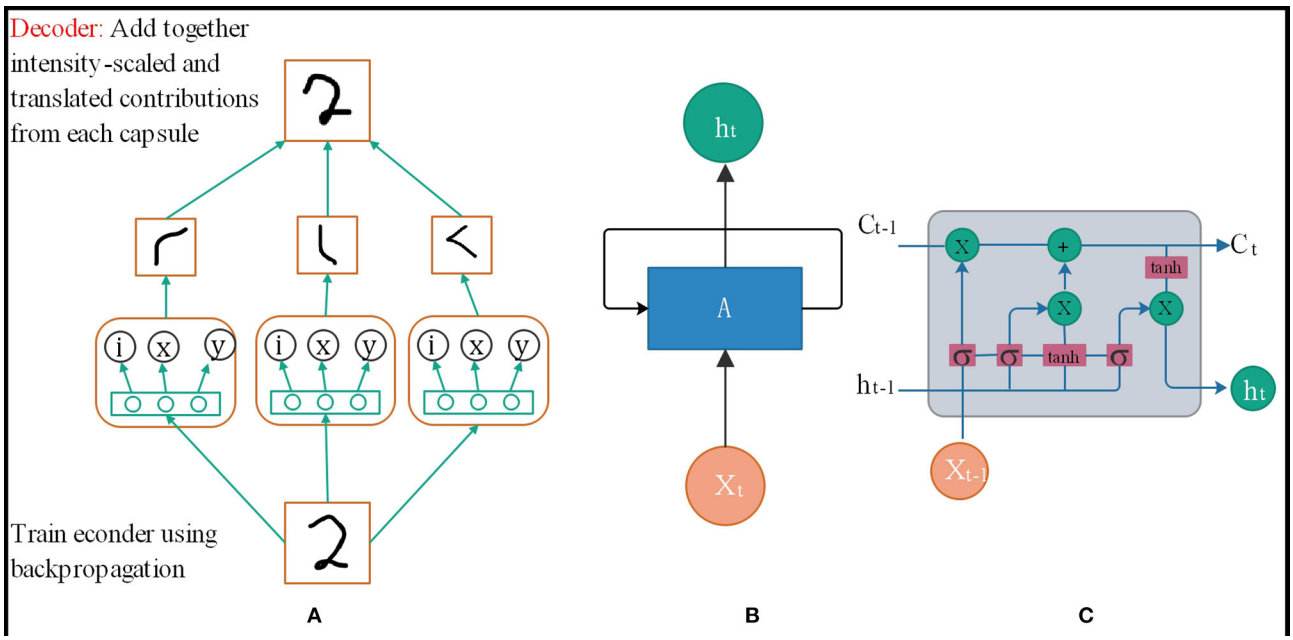


FIGURE 1 | (A) Capsule neural network structure. **(B)** RNN neural network structure. **(C)** LSTM neural network structure.

Self-Attention Long Short-Term Memory

When analyzing the flow characteristics between network sessions, it can be found that it is more like a sentence that conforms to the special established rules, except that it is composed of bytes to form a data packet and then a session. It is very similar to the structure of natural language problems where words form sentences and sentences form paragraphs. The principle of using a time series neural network to identify which application a session belongs to is also similar to the principle of sub-classification. The essence is to use the form of an array instead of the data itself as input. A similar time-series relationship of similar data is obtained through the model to achieve classification.

As shown in **Figure 1B**, the traditional RNN network model cannot fully handle the timing problem because it needs to determine the parameters before it can be predicted, but in the process of training the parameters of the neural network model, the chain derivation rule to determine the gradient increment of the parameter is essential of. In this case, when the problem sequence to be dealt with is relatively long, it will be relatively large, and the upper bound of the derivative of the common activation function is, so when it is very large, there will be many less than the numerical value in the chain derivation rule. When the time sequence is very long, the parameter increments updated by the chain derivation rule will be close, causing the parameters to be unable to be updated. In other words, RNN can handle timing problems, but when the sequence of timing problems is very long, the information cannot be very effective Was saved, so researchers proposed an improved model LSTM[38]. The network structure is shown in **Figure 1C**.

The reason why LSTM can solve the long time sequence problem of t is because the cell state C_i is used to assist in the transfer of data to time t, instead of relying on the hidden layer information at time h_{t-1}, and the update formula of cell state C_i can also be used. Knowing that it is updated through addition. The advantage of this is that when the chain rule is updated, the cell state C_i will not appear as the updated parameter increment in the RNN chain derivation rule will be close to 0.

The attention mechanism was first proposed in the field of visual images, and has gradually been widely used in natural language since 2014. Today, the combination of various attention mechanisms and deep learning network models has achieved good results in natural language problems. As a result, this article also considers adding a self-attention mechanism when using the time-series deep learning model LSTM to process traffic.

Self-attention (Tao et al., 2020). The above process can be abstracted as the calculation of similarity between query, key and value, which can be roughly divided into three stages:(1) query and key_i use the similarity function that matches the task to calculate the similarity, and get the parameter s_i. (2) Normalize s_i with $softmax()$ to get α_i. (3) After multiplying the corresponding α_i and $value_i$, and then summing, the self-attention value can be finally obtained. The calculation process is as follows:

$$f(Q, K_i) = Q^T K_i$$

$$\alpha_i = softmax\left(f(Q, K_i)\right) = \frac{\exp\left(f(Q, K_i)\right)}{\sum_i \exp\left(f(Q, K_i)\right)}$$

$$self - attention(Q, K, V) = \sum_i \alpha_i V_i (Q = K = V)$$

FRAMEWORK AND METHODS

Our framework logically consists of two parts: the "pre-processor" and the "traffic classifier." The former has performed all tasks that allow us to model the network traffic into data, which can easily be handled by a deep learning model. The latter performs specific classification tasks. Before using a deep learning traffic classifier, it must be trained with a large amount of labeled traffic data. The traffic data we use comprises the public dataset ISCX2012 and the data generated by artificially stimulating the apps.

Network Traffic Pre-processing

There are two traffic forms: flow and session. Usually, we use five tuples to determine flow and session. Flow: a time-ordered sequence of packets exchanged between two peers during a single TCP session. f $= < p_1, p_2, \ldots, p_N >$, N is the number of packets consist of flow f. packet $p_i = (a_i, l_i, t_i)$. a_i is a 5-tuple that consists of source address, source port, a destination address, destination port, and the protocol type. l_i is the length of the packet p_i and t_i is the time of packet p_i arrival. The total flow length L $= \sum_{i=1}^{n} l_i$ and $t_1 \leq t_2, \ldots, \leq t_n$. Session and flow are similar. The difference is that the source and destination addresses can be exchanged, which is a bidirectional data flow. Network traffic can be divided into four levels, application layer, transport layer, network layer, and all layers. The input data of the traffic classifier, a combination of different traffic forms and different network layers, are eight types. The pre-processing is shown as **Figure 2**.

Packers Filtering

Due to network congestion, traffic load balancing, or other unpredictable network behaviors, data packets may be lost and arrive out of order. When TCP detects these problems, it will retransmit network data packets and rearrange out-of-order data packets. Repeated transmission packets will affect the characteristics of network traffic, and network flows without data content will not use in identification. Therefore, we filter out retransmissions and packets with only ACK flags with zero payloads. The sequence of network packets has an essential impact on recognition. We will rearrange out-of-order data packets to obtain a correct network flow sequence.

Traffic Data Processing

Traffic classification should remove data related to the hardware environment and network environment, such as IP information in the network layer and MAC addresses in all layers. Therefore, training data and test data may have different physical addresses and IP addresses. We need to remove these features that may change to ensure the feature consistency of the training data and test data.

The input length of the classifier is consistent to ensure the correct subsequent recognition. At the same time, the selected data length has a significant impact on the recognition result. Through analysis of the contribution of bytes in the stream, we found that the first 400 bytes in the stream have an impact on the traffic classification because the recognition contribution is the highest. Therefore, we choose 400 as the input of the convolutional neural network. Data streams with a length of <400 will be dropped, and data streams with a length of more than 400 bytes will only take the first 400 bytes of the data stream as classifier input.

Many previous research works have shown that CNN neural network has a good classification performance for image recognition. Here we take the first 400 bytes of a network stream, and we can get a 20*20 two-dimensional matrix. Each value range of the matrix is 0–255, so each value can be regarded as the gray value of a pixel. Thus, picture data containing the first 400 bytes of information in the data stream can be obtained. When the network traffic is processed and converted into pictures, we get the features of the traffic stream. This paper uses the CapsuleNet neural network, capsule represented by a vector can learn the spatial feature relationship of the flow graph well.

Traffic Classification

Convolutional Neural Network

The first 400 bytes of encrypted data traffic is converted into a 20*20 grayscale image, which is the input of the traffic classifier. The Inception-CapsuleNet network designed in this paper is divided into nine layers. The first four layers extract the characteristics of the traffic, the middle four layers combine the characteristics, and the last layer is the category output layer, as shown in **Table 1**. Before the grayscale image entry the model, the mean value is zeroed first so that the model converges quickly.

The first layer is a convolutional layer, which extracts local features of grayscale images. In order to learn more about the local features of the input data, the step size is 1. The second layer uses batch normalization, which can prevent the data distribution from changing greatly after passing through the previous layer, and can avoid the gradient disappearance and overfitting problems. In the third inception layer, convolution kernels of different sizes are used for feature processing.

Convolutions of different sizes can extract features from different images in different fields of view, which can increase the ability of the network to extract features. Finally, the outputs of different convolution kernels are spliced together to obtain a feature with a dimension of 256. There are 32 capsules in the PrimaryCaps layer, each of which will convolve all the inputs of the previous layer. Here the activation parameter is squash, and an out tensor is u_i, with a shape of 4*4*8*32. The input data in the DigitCaps layer is the vector u_i. The calculation process is as follows:

$$\hat{u}_{j|i} = w_{ij} \cdot u_{ij}$$
$$b_{ij} = b_{ij} + \hat{u}_{j|i} \cdot v_j$$
$$c_{ij} = softmax\left(b_{ij}\right) = \frac{\exp\left(b_{ij}\right)}{\sum_k \exp\left(b_{ik}\right)}$$
$$s_j = \sum c_{ij} \cdot \hat{u}_{j|i}$$
$$v_{ij} = \frac{\| s_j \|^2}{1 + \| s_j \|^2} \cdot \frac{s_j}{\| s_j \|}$$

Where s_j is the final input, the final output vector is v_j and $b_{ij}^{(0)} = 0$. After the activation function Squash, b_{ij}, c_{ij} can be

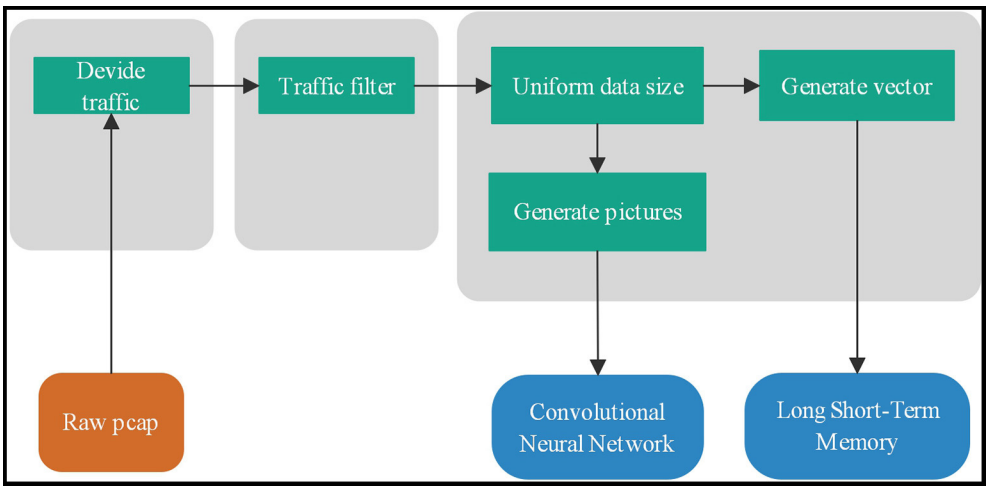


FIGURE 2 | Network traffic data packet processing process: flow division, flow filtering, uniform size and classification.

TABLE 1 | Capsule convolutional neural network.

Layer	Name	Activation function	Input size	Convolution kernel	Step	Output size
1	Condv2	ReLU	20*20*1	9*9*256	1	12*12*128
2	Batch Norm	–	12*12*128	–	–	12*12*128
3	Inception	ReLU	12*12*128	–	–	12*12*256
4	Primary Caps	Squash	12*12*256	6*6*256*8	2	4*4*8*32
5	DigitCaps	Squash	4*4*8*32	–	–	10*12
6	Full Connect	ReLu	10*12	–	–	256
7	Full Connect	ReLU	256	–	–	128
8	Full Connect	ReLU	128	–	–	64
9	Full Connect	Softmax	64	–	–	8

updated by Equations (2) and (3). The best parameter selection can be achieved by continuously repeating the above process. At the same time, in order to prevent over-fitting, the number of iterations here is selected as three, and finally, ten capsules are output, and each capsule is a 12-dimensional vector.

The following are three fully connected layer classification networks, and finally, a vector of length 64 is obtained, and the last one is a fully connected softmax activation function classifier.

Long Short-Term Memory Network

In order to make full use of the characteristics of the network flow, the network structure of this article first uses the self-encoding method to train the data packets into a unified array specification as the information representing the encrypted flow. Then use the time series neural network to extract the timing behavior characteristics of the data packet exchange process at both ends of the conversation, and use the characteristics to classify the encrypted network traffic of the application. Because the flow is in the process of data packet exchange, the encrypted content of the current data packet may be determined by the protocol in a previous data packet. The pure LSTM time series network cannot well capture the characteristic information generated by this behavior. This article will add from The attention mechanism allows each data packet unit to better correspond to its own related data packets during the training process, hoping to obtain more comprehensive network traffic characteristics and achieve higher recognition results.

The data processing process is shown in **Figure 3**. Before the model training is carried out, after the traffic is cleaned, the data packets are sorted according to the flow or session according to the packet sending time of the packet header. After the traffic enters the model, first select the number of reserved data packets, then select the reserved byte length of each data packet, and then convert multiple data packet bytes into an array vector through the method of self-encoder. Finally, it enters the model's feature extraction classifier training stage.

In the stage of training the word vector, the traditional method is to use the word form to represent the word vector. But when the number of dictionaries is too long, the word vector cannot be used in deep learning algorithms. The relevance of the word vector is very poor, so this article uses the Distributed method to train the word vector. The vector obtained by this method can be controlled to a shorter length.

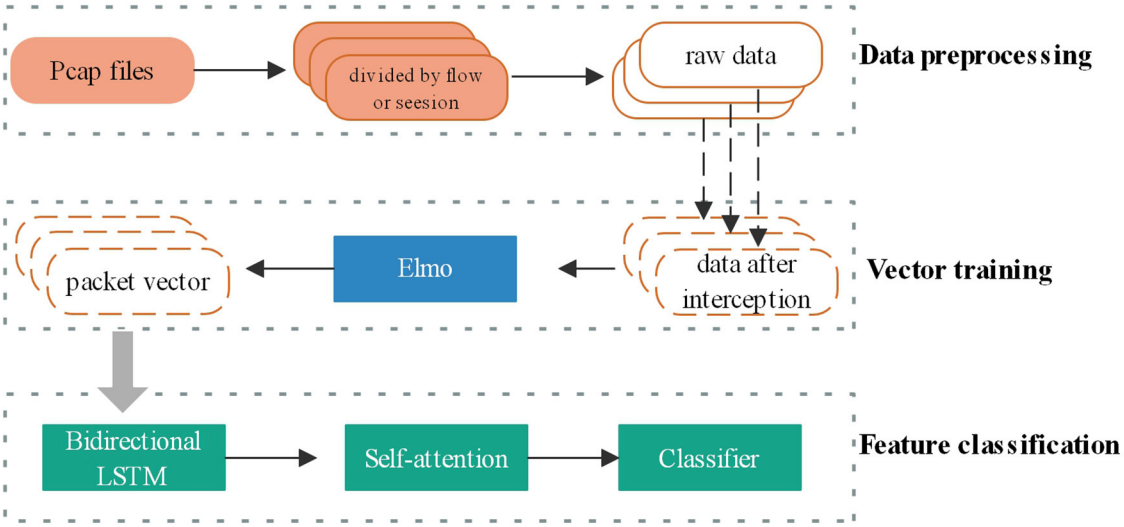


FIGURE 3 | LSTM neural network data processing process.

When training the word vector of the data packet, it is considered that the meaning of the same byte in the encrypted traffic protocol of different applications is very different. If a traditional word vector training method, such as word2vec is used to obtain a fixed word vector, this does not conform to the characteristics of the data in the encrypted traffic recognition model. Elmo's two-layer bidirectional LSTM network pre-training structure can obtain word vectors that meet the complexity of the protocol, and the same byte corresponds to different word vectors in different protocol environments.

Regarding the hexadecimal bytes in the pcap packet, this article converts it into decimal data as the original input, and then intercepts the 96 bytes of the first six packets of each stream or session as input. Less than 96 bytes, using the method of adding 0 to fill in, the form of the array converted into an array of 6*96 dimensions. Its form is similar to the way of analyzing a sentence category in text classification. For a 6-word sentence, the word vector length of each word is fixed at 96. In form, it is similar to the first 6 data contained in a stream or conversation. The information contained in the data packet represents that the two ends of the conversation are communicating through a certain language, so it is reasonable to input it into the training model of natural language word vector Elmo.

In the Elmo stage, the Embedding of the first layer of LSTM and the Embedding of the second layer of LSTM are multiplied by the corresponding weights, and the final Embedding is $< t_1, t_2, \ldots, t_n >$ Then, where n represents the number of data packets, the value here is 6, that is, 6 data packets are selected, and each data packet intercepts 96 bytes as input. The size of the LSTM of the Elmo stage is 128, that is, the size of the vector t_1 length, and then input Elmo's result data into the LSTM+Self-Attention model.

After entering the LSTM+Sefl-Attention stage, after the first layer of LSTM, the activation function is Relu, the word vector feature of the data packet is extracted and the size is 256 as the input of Batch Normalization to ensure that the data distribution remains unchanged while maintaining 256 The length remains unchanged and enters the next layer of Self-Attention. After this network layer, you can learn an encrypted session or stream. The internal structure of the intercepted data packets and the dependency relationship between the protocol features help The model identifies and classifies encrypted traffic. After passing through the Sefl-Attention network layer, the traffic length remains unchanged at 256, and then passes through a layer of Batch Normalization to enter the fully connected layer. The number of neurons in the first fully connected layer is 64, the activation function is Relu, and the second fully connected layer. The number of neurons is 7, the activation function is Softmax, which is used to finally output the probability that the encrypted traffic belongs to the target application.

CNN-LSTM Joint Network

A reasonable network structure plays an important role in the process of deep learning to identify encrypted traffic. This article draws on the idea of bagging, and designs a neural network with convolutional time series to identify encrypted traffic, as shown in **Figure 4**. The model has two inputs for the same sample. On the left side of the model, the input is the overall picture converted from the session bytes recombined from the data packet. This side of the model learns the structure information characteristics of the encryption suite of the traffic data; On the right side of the model, the input is bytes intercepted by multiple consecutive data packets in a session. The right side of the model learns the behavioral communication characteristics between traffic data, combines the two feature vectors together through splicing, and then passes through the neural network Layers are classified. Compared with the previous convolutional neural network and time series neural network, the combined

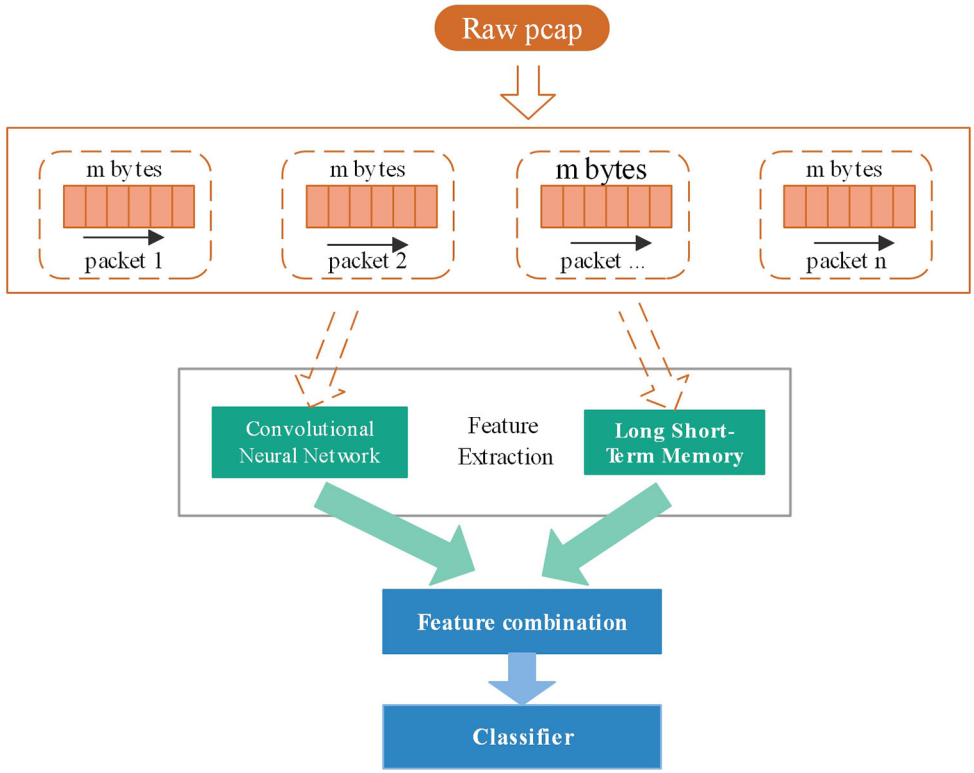


FIGURE 4 | Convolutional neural network and LSTM neural network joint processing model.

TABLE 2 | CNN network structure on the left.

layer	Name	Activation function	Convolution kernel	Kernel number	Step
1	Condv2	ReLU	3*3	256	1
2	Condv2	ReLU	3*3	128	1
3	Batch Norm	–	–	–	–
4	Inception-1	ReLU	1*1	128	1
4	Inception-2	ReLU	1*1 3*3	64	1
4	Inception-3	–	3*3 1*1	64	1
5	Inception-concact	–	–	–	–
6	PrimaryCaps	Squash	6*6*256	8	2
7	DigitCaps	Squash	–	–	–

neural network designed in this paper has an accuracy increase of nearly 4%, which further proves that a reasonable neural network structure is essential for the improvement of encrypted traffic recognition.

In **Figure 4**, two neural network structures are adopted to extract the different characteristics of encrypted traffic. The convolutional neural network on the left learns the overall structural characteristics of the traffic data packet, intercepting the first m data packets of the session, each The data packet intercepts n bytes, and each byte corresponds to two hexadecimal digital representations in the original traffic, which can be converted into a 1–255 decimal representation as input to obtain a matrix of size m*n. The process of the convolutional network refers to the network structure of session2, and uses multiple convolution kernels of different sizes to convolve the same data. The network structure is shown in **Table 2**.

The time series neural network used on the right is used to learn the communication timing characteristics between encrypted traffic data packets. Its input is the same as the data format on the left. It intercepts the first m data packets of

the session, and each data packet intercepts n bytes. A byte corresponds to two hexadecimal digital representations in the original traffic, which can be converted into a 1–255 decimal representation as input, and a two-dimensional array is obtained, so that the input representation is the same as the natural language text analysis The input method of a sentence is the same. Compared with the traditional time series network structure, this paper adds a self-attention mechanism, so that it can capture the dependence of learning data packet communication behavior and learn more timing characteristics. The network structure is shown in **Table 3**. The output data of the network structure on the left and right sides are spliced together as the input of the classifier. This part of the network structure is shown in **Table 4**.

TABLE 3 | LSTM network structure on the right.

Layer	Name	Activation function	Weight matrix
1	Elmo-1-Forward LSTM	ReLU	96*128
1	Elmo-1-ReverseLSTM	ReLU	96*128
2	Elmo-2-Forward LSTM	ReLU	128*128
2	Elmo-2-Reverse LSTM	ReLU	128*128
3	Elmo-concact	–	–
2	Two-Way LSTM	ReLU	128*256
3	Batch Norm	–	–
4	Self-Attention	–	–
5	Batch Norm	–	–

TABLE 4 | Classification layer network structure.

Layer	Name	Activation function	Length
1	Concact	–	–
2	Full Connect	ReLU	512
3	Full Connect	ReLU	64
4	Full Connect	Softmax	7

RESULT

The system of the experimental environment is Ubuntu 16.0, based on Keras running framework. RAM is 96G and video memory is 16G.

Different input byte length will affect the discrimination effect of the classifier, so choosing the appropriate byte input length is very important for the classifier. We have studied the impact of the byte input length from 50 to 750 on the classification results, as shown in **Figure 5A**. As the input byte length increases, the more features the classifier can use, the better the classification effect of the classifier. When the length increases at a certain threshold, the classification effect has not improved significantly. In order to save computing resources and time, we select the input data length as short as possible.

The choice of loss function will affect the classification effect of the classifier. The loss function calculates the distance between the probability distribution p and the probability distribution q predicted by the classifier. This article uses a loss function optimized based on focal loss *Loss(p,q)=L*.

$$L = \sum_{i=1}^{K} \sum_{j=1}^{K'} \alpha_i \left(1 - q\left(x_{ij}\right)\right)^{\gamma} p\left(x_{ij}\right) \lg\left(q\left(x_{ij}\right)\right)$$

For a certain type of sample, the higher the value of $q\left(x_{ij}\right)$ is, the smaller the value of $1 - q\left(x_{ij}\right)$ is, thus reducing the weight of this type of sample. The value of $q\left(x_{ij}\right)$ is small, and the $1 - q\left(x_{ij}\right)$ will be large, which can increases the weight of the sample recognized

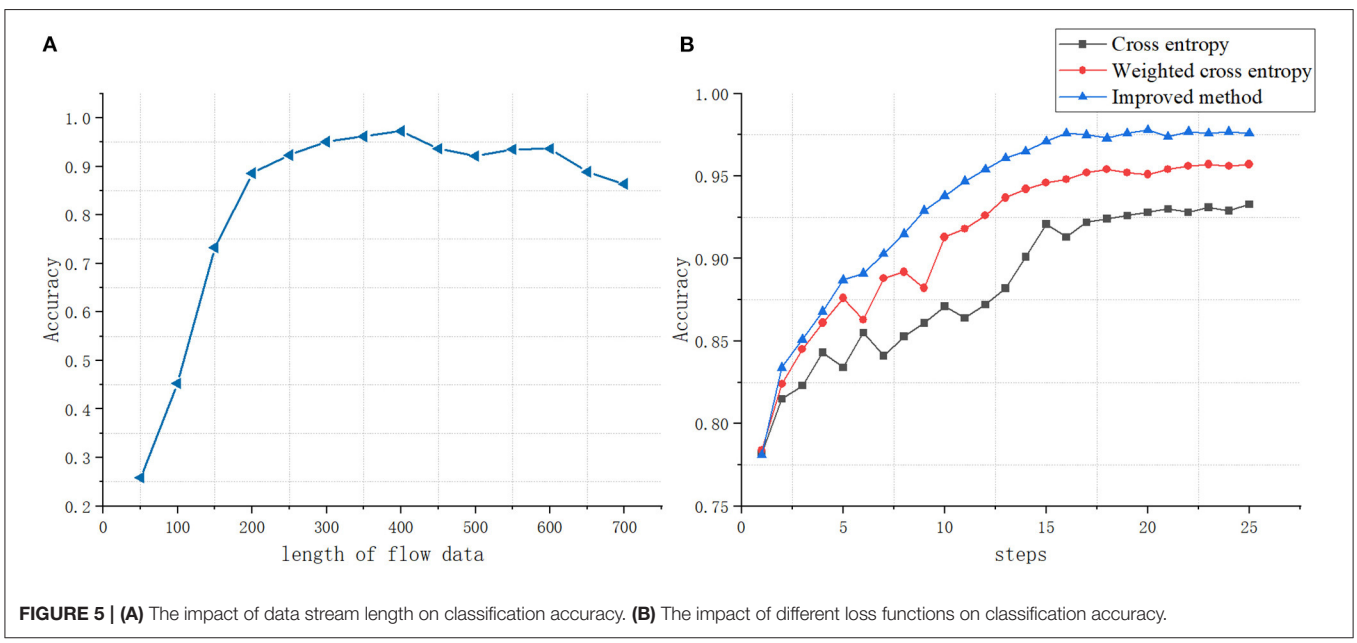


FIGURE 5 | (A) The impact of data stream length on classification accuracy. **(B)** The impact of different loss functions on classification accuracy.

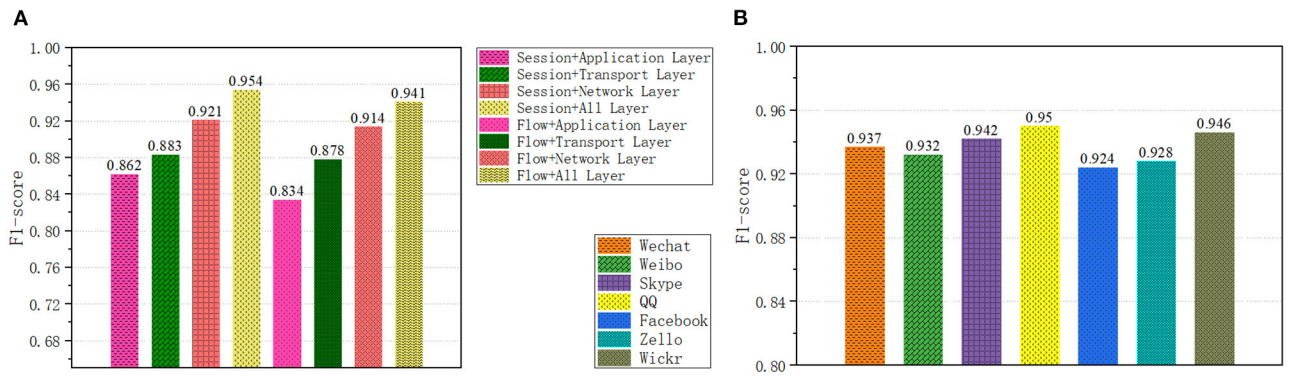


FIGURE 6 | (A) Classification results of different network traffic forms. **(B)** Application classification results of convolutional neural networks.

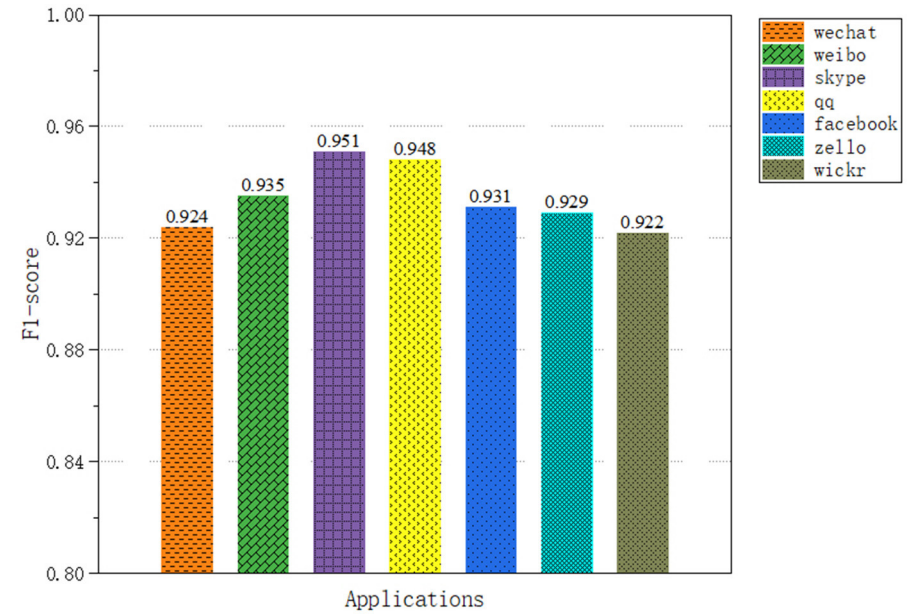


FIGURE 7 | Application classification results of LSTM neural network.

hard. The value of α_i is inversely proportional to the number of samples of each type. Use the parameter γ to automatically adjust the ratio of the loss function. This not only considers the imbalance of sample categories in traffic identification but also solves the difference of recognization cost in different samples. **Figure 5B** shows the comparison with the cross-entropy loss function and the weighted cross-entropy loss function.

Different traffic forms and network layer divisions contain different data content. We have studied the impact of different traffic forms and network layers on the classifier, as shown in **Figure 6A**. A total of eight types of samples were obtained for the two traffic forms and the four network layers. On the data set ISCX (Draper-Gil et al., 2016), we compared the recognition effects of the eight forms. As can be seen from **Figure 6A**, Session + All Layer performs best. The accuracy rate is 0.942, the recall rate is 0.973, and the F1-score is 0.955, because it contains more traffic characteristics.

In application classification, we selected seven applications, a total of 52,155 encrypted network traffic samples, with an average of 7,000 sample data for each application. The results of the capsule convolutional neural network are shown in **Figure 6B** and the results of Long short-term memory network are shown in **Figure 7**. The results show that these two methods have excellent performance for traffic classification.

Many scholars have performed classification method evaluation on the data set ISCX (Moore and Zuev, 2005; Alberto et al., 2006; Huang et al., 2014). Here we compare the Inception-CapsNet classifier with them. As shown in **Table 6**, it can be seen that the accuracy and recall rate of the other four types of methods have been improved. For the decision tree C4.5

algorithm, the accuracy rate has increased by 4.3%, and the recall rate has increased by 7.0%.

The CNN-LSTM joint model experimental results are shown in **Figure 8**. For Hangout and Bittorrent, the convolution method has a high recognition accuracy of 96%. For Facebook, Skype has a low recognition accuracy rate of only 87%, while for ATM, Hangout, and Bittorrent, The recall rate is high, reaching 98.

The results show that the recognition accuracy of the time series method for Facebook, Skype, Hangout is as high as 97%, and the recognition accuracy for AIM and uTorrent is low, only 91%. Facebook's recognition recall rate is even higher, reaching 99%.

TABLE 5 | Comparison of classification performance with other traffic classification algorithms on Datasets ISCX.

Algorithm	Accuracy	Recall
CNN-LSTM	0.981	0.995
C4.5	0.901	0.903
SVM	0.943	0.929
1dCNN	0.933	0.951
2dCNN	0.936	0.955
Apriori	0.931	0.911
Naïve Bayes	0.911	0.927
Hmm-crf	0.955	0.967

TABLE 6 | Comparison of classification performance with other traffic classification algorithms on datasets UNIBS.

Algorithm	Accuracy	Recall
LSTM	0.945	0.973
SVM	0.959	0.953
Multi-classifier	0.924	0.971
Random forest	0.936	0.992
Xgboost	0.931	0.961
CNN-LSTM	0.986	0.987

For the combined model, the recognition accuracy has been significantly improved. Among them, Facebook, Hangout, and Bittorrent have a high recognition accuracy of 98.5%; for ATM, FTPS, Skype, and uTorrent, the recognition accuracy is high. Reached 96%; and for the recall rate, the recall rate of encrypted traffic of AIM, Facebook, Hangout, Bittorrent, and Skype reached 99%. For the traffic FTPS, the recall rate of the two encrypted traffic of uTorrent reached 96%.

From the above results, it can be seen that different network structures extract features of different dimensions, and have different preferences for the quasi-curvature and recall rate of encrypted traffic recognition of the same application. For some traffic convolutional neural networks, the recognition accuracy rate is higher. But the recall rate may be relatively low, such as FTPS. For the encrypted traffic of this kind of application, although the recognition quasi-curvature of the time series neural network is moderate, the recall rate is high. In the experimental results of, this conclusion has also been proved, the accuracy rate of its encrypted traffic has reached more than 96%, the recall rate is 99%, only two kinds are lower, and it is also more than 96%. The effect is compared with the previous neural network. The organization has very distinctive features. This proves that extracting features from different dimensions and different feature spaces helps to capture more recognizable features and enhance the model effect.

Table 5 shows the comparison results on ISCX, where the number of features used by the machine learning method is different, reflecting the complexity of the manual design, such as SVM using 21 manual design Features. The decision tree uses 18 artificially designed features, but the two methods have only ten common features, and it is difficult to generalize to other data sets. **Table 6** shows the comparison results on the UNIBS data (Gringoli et al., 2009). In this data set, the results obtained by the relevant literature are given by the SVM model, but the accuracy of the combined model exceeds the model, and the recall rate is far higher. For SVM, it is 0.05 less than the random forest model with the best recall rate, ranking second, and the overall effect is

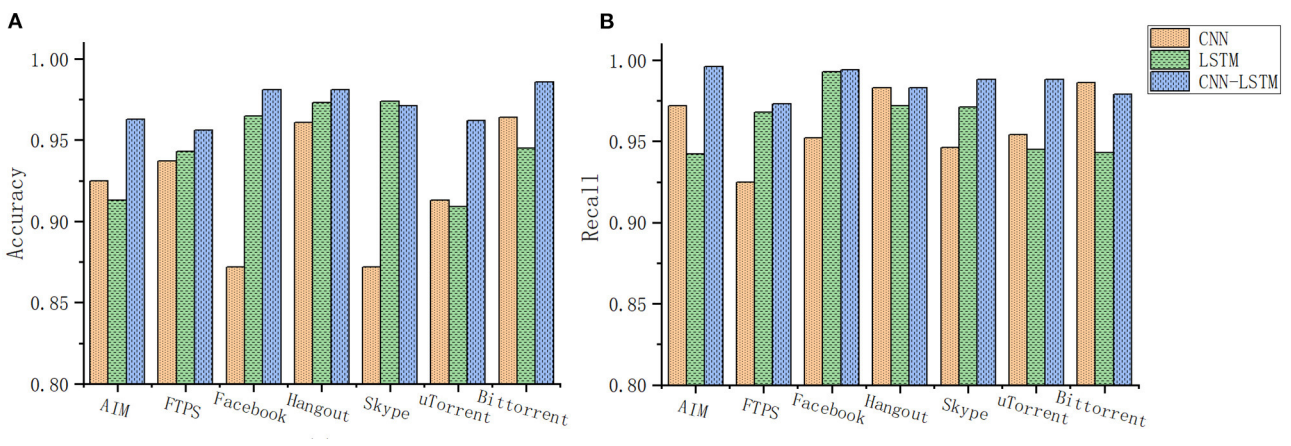


FIGURE 8 | (A) CNN-LSTM joint network application classification accuracy rate. **(B)** CNN-LSTM joint network application classification recall rate.

very good. The results show that the accuracy of the CNN-LSTM network is 8% higher than other methods.

CONCLUSION

In this literature, we propose a capsule convolutional neural network joint the long short-term memory network traffic classification framework. For the problem of imbalance of sample categories in flow recognition, an objective function related to weight and sample recognition accuracy is designed to reduce the classification impact caused by sample imbalance. Besides, the inception structure is added to allow the model to learn diverse features, and the capsulenet structure is added to allow the model to learn the correlation of high and low dimensional features. This model can automatically identify a variety of encrypted traffic and seek the global optimal classification result. The experimental results show that this method can effectively classify the encrypted traffic and is better than previous research work. At the same time, our work proves that the optimized neural network structure can achieve better recognition results.

As future work, We believe that we should try to pay attention to the characteristics of network traffic of different behaviors, so as to more comprehensively describe the communication process between users and robots.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author/s.

AUTHOR CONTRIBUTIONS

MG contributed the idea of the paper, verified the idea through experiments, and wrote the paper at the same time. XY guided the idea of the paper and LL assisted the experiment. All authors contributed to the article and approved the submitted version.

FUNDING

This work was supported by National Key R&D Program of China (2016QY05X1000) and National Natural Science Foundation of China (201561402137).

REFERENCES

Alberto, D., Antonio, P., and Pierluigi, S. R. (2006). "An HMM approach to internet traffic modeling," in *Proceeding of IEEE GLOBECOM* (San Francisco, CA). doi: 10.1109/GLOCOM.2006.453

Alizadeh, H., Khoshrou, A., and Zuquete, A. (2015). "Traffic classification and verification using unsupervised learning of Gaussian Mixture Models," in *2015 IEEE International Workshop on Measurements & Networking* (Coimbra), 1–6. doi: 10.1109/IWMN.2015.7322980

Deng, F., Pu, S., Chen, X., Shi, Y., Yuan, T., and Pu, S. (2018). Hyperspectral image classification with capsule network using limited training samples. *Sensors*. 18:3153. doi: 10.3390/s18093153

Draper-Gil, G., Lashkari, A. H., Mamun, M. S. I., and Ghorbani, A. A. (2016). "Characterization of encrypted and vpn traffic using time-related," in *Proceedings of the 2nd International Conference on Information Systems Security and Privacy (ICISSP)* (Bologna), 407–414.

Ebrahimi, M. A., Khoshtaghaza, M. H., Minaei, S., and Jamshidi, B. (2017). Vision-based pest detection based on SVM classification method. *Comput. Electron. Agric.* 137, 52–58. doi: 10.1016/j.compag.2017.03.016

Este, A., Gringoli, F., and Salgarelli, L. (2009). Support vector machines for TCP traffic classification. *Comput. Netw.* 53, 2476–2490. doi: 10.1016/j.comnet.2009.05.003

Fan, Z., and Liu, R. (2017). "Investigation of machine learning based network traffic classification," in *2017 International Symposium on Wireless Communication Systems* (Bologna), 1–6. doi: 10.1109/ISWCS.2017.8108090

Finsterbusch, M., Richter, C., Rocha, E., Muller, J. A., and Hanssgen, K. (2013). A survey of payload-based traffic classification approaches. *IEEE Commun. Surv. Tutor.* 16, 1135–1156. doi: 10.1109/SURV.2013.100613.00161

Fu, R., Zhang, Z., and Li, L. (2016). "Using LSTM and GRU neural network methods for traffic flow prediction," in *2016 31st Youth Academic Annual Conference of Chinese Association of Automation* (Wuhan), 324–328. doi: 10.1109/YAC.2016.7804912

Gleeson, B., MacLean, K., Haddadi, A., Croft, E., and Alcazar, J. (2013). "Gestures for industry intuitive human-robot communication from human observation," in *2013 8th ACM/IEEE International Conference on Human-Robot Interaction* (Tokyo), 349–356. doi: 10.1109/HRI.2013.6483609

Gringoli, F., Salgarelli, L., Dusi, M., Cascarano, N., Risso, F., and Claffy, K. C. (2009). Gt: picking up the truth from the ground for internet traffic. *ACM SIGCOMM Comput. Commun. Rev.* 39, 12–18. doi: 10.1145/1629607.1629610

Groleat, T., Arzel, M., and Vaton, S. (2012). "Hardware acceleration of SVM-based traffic classification on FPGA," in *2012 8th International Wireless Communications and Mobile Computing Conference* (Limassol). doi: 10.1109/IWCMC.2012.6314245

Gu, Z., Hu, W., Zhang, C., Lu, H., Yin, L., and Wang, L. (2020). Gradient shielding: towards understanding vulnerability of deep neural networks. *IEEE Trans. Netw. Sci. Eng.* doi: 10.1109/TNSE.2020.2996738

Hjelmvik, E., and John, W. (2009). "Statistical protocol identification with spid: preliminary results," in *Swedish National Computer Networking Workshop* (Uppsala), 4–5.

Huang, C., Gao, D., Hu, S., Geng, H., and Peng, Y. (2014). Association rule analysis of vessel traffic accidents based on Apriori algorithm. *J. Shanghai Maritime Univ.* 35, 18–22.

Kanda, T., Shiomi, M., Miyashita, Z., Ishiguro, H., and Hagita, N. (2010). A communication robot in a shopping mall. *IEEE Trans. Robot.* 26, 897–913. doi: 10.1109/TRO.2010.2062550

Kornycky, J., Abdul-Hameed, O., Kondoz, A., and Barber, B. C. (2016). Radio frequency traffic classification over WLAN. *IEEE/ACM Trans. Netw.* 25, 56–68. doi: 10.1109/TNET.2016.2562259

Li, F., Seddigh, N., Nandy, B., and Matute, D. (2000). "An empirical study of today's Internet traffic for differentiated services IP QoS," in *Proceedings ISCC 2000. Fifth IEEE Symposium on Computers and Communications* (Antibes-Juan Les Pins), 207–213. doi: 10.1109/ISCC.2000.860640

Li, Z., Yuan, R., and Guan, X. (2007). "Accurate classification of the internet traffic based on the SVM method," in *2007 IEEE International Conference on Communications* (Glasgow), 1373–1378. doi: 10.1109/ICC.2007.231

Liu, Y., Li, W., and Li, Y. (2007). "Network traffic classification using k-means clustering," in *Second International Multi-Symposiums on Computer and Computational Sciences* (Iowa City, IA), 360–365. doi: 10.1109/IMSCCS.2007.52

Madhukar, A., and Williamson, C. (2006). "A longitudinal study of P2P traffic classification," in *14th IEEE International Symposium on Modeling, Analysis, and Simulation* (Monterey, CA), 179–188. doi: 10.1109/MASCOTS.2006.6

Mavridis, N. (2015). A review of verbal and non-verbal human–robot interactive communication. *Robot. Auton. Syst.* 63, 22–35. doi: 10.1016/j.robot.2014.09.031

Moore, A. W., and Papagiannaki, K. (2005). "Toward the accurate identification of network applications," in *International Workshop on Passive and Active Network Measurement* (Berlin; Heidelberg), 41–54. doi: 10.1007/978-3-540-31966-5_4

Moore, A. W., and Zuev, D. (2005). "Internet traffic classification using bayesian analysis techniques," in *Proceedings of the 2005 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems* (Banff, AB), 50–60. doi: 10.1145/1064212.1064220

Pacheco, F., Exposito, E., Gineste, M., Baudoin, C., and Aguilar, J. (2018). Towards the deployment of machine learning solutions in network traffic classification: a systematic survey. *IEEE Commun. Surv. Tutor.* 21, 1988–2014. doi: 10.1109/COMST.2018.2883147

Rezaei, S., and Liu, X. (2019). Deep learning for encrypted traffic classification: an overview. *IEEE Commun. Mag.* 57, 76–81. doi: 10.1109/MCOM.2019.1800819

Shafiq, M., Tian, Z., Bashir, A. K., Du, X., and Guizani, M. (2020). Corrauc: a malicious bot-iot traffic detection method in iot network using machine learning techniques. *IEEE Internet Things J.* 8, 3242–3254. doi: 10.1109/JIOT.2020.3002255

Shafiq, M., Yu, X., Bashir, A. K., Chaudhry, H. N., and Wang, D. (2018). A machine learning approach for feature selection traffic classification using security analysis. *J. Supercomput.* 74, 4867–4892. doi: 10.1007/s11227-018-2263-3

Si, C., Chen, W., Wang, W., Wang, L., and Tan, T. (2019). "An attention enhanced graph convolutional LSTM network for skeleton-based action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Long Beach, CA), 1227–1236. doi: 10.1109/CVPR.2019.00132

Siahaan, H., Mawengkang, H., Efendi, S., Wanto, A., and Windarto, A. P. (2019). Application of classification method C4. 5 on selection of exemplary teachers. *J. Phys. Conf. Ser.* 1235:1. doi: 10.1088/1742-6596/1235/1/012005

Su, S., Tian, Z., Liang, S., Li, S., Du, S., and Guizani, N. (2020). A reputation management scheme for efficient malicious vehicle identification over 5G networks. *IEEE Wireless Commun.* 27, 46–52. doi: 10.1109/MWC.001.1900456

Sun, G., Chen, T., Su, Y., and Li, C. (2018). Internet traffic classification based on incremental support vector machines. *Mobile Netw. Appl.* 23, 789–796. doi: 10.1007/s11036-018-0999-x

Tao, W., Li, C., Song, R., Cheng, J., Liu, Y., Wan, F., et al. (2020). EEG-based emotion recognition via channel-wise attention and self attention. *IEEE Trans. Affect. Comput.* doi: 10.1109/TAFFC.2020.3025777

Tian, Z., Luo, C., Qiu, J., Du, X., and Guizani, M. (2020). A distributed deep learning system for web attack detection on edge devices. *IEEE Trans. Ind. Inform.* 16, 1963–1971. doi: 10.1109/TII.2019.2938778

Vinayakumar, R., Soman, K. P., and Poornachandran, P. (2017). "Applying convolutional neural network for network intrusion detection," in *2017 International Conference on Advances in Computing, Communications and Informatics* (Udupi), 1222–1228. doi: 10.1109/ICACCI.2017.8126009

Wang, P., Chen, X., Ye, F., and Sun, Z. (2019). A survey of techniques for mobile service encrypted traffic classification using deep learning. *IEEE Access* 7, 54024–54033. doi: 10.1109/ACCESS.2019.2912896

Wang, W., Zhu, M., Wang, J., Zeng, X., and Yang, Z. (2017). "End-to-end encrypted traffic classification with one-dimensional convolution neural networks," in *2017 IEEE International Conference on Intelligence and Security Informatics* (Beijing), 43–48. doi: 10.1109/ISI.2017.8004872

Wu, D., Chen, X., Chen, C., Zhang, J., Xiang, Y., and Zhou, W. (2015). "On addressing the imbalance problem: a correlated KNN approach for network traffic classification," in *International Conference on Network and System Security* (New York, NY), 138–151. doi: 10.1007/978-3-319-11698-3_11

Xiang, C., Zhang, L., Tang, Y., Zou, W., and Xu, C. (2018). MS-CapsNet: a novel multi-scale capsule network. *IEEE Signal Process. Lett.* 25, 1850–1854. doi: 10.1109/LSP.2018.2873892

Yin, C., Li, S., and Li, Q. (2012). Network traffic classification via HMM under the guidance of syntactic structure. *Comput. Netw.* 56, 1814–1825. doi: 10.1016/j.comnet.2012.01.021

Yuan, R., Li, Z., Guan, X., and Xu, L. (2010). An SVM-based machine learning method for accurate internet traffic classification. *Inform. Syst. Front.* 12, 149–156. doi: 10.1007/s10796-008-9131-2

Zhu, Z., Peng, G., Chen, Y., and Gao, H. (2019). A convolutional neural network based on a capsule network with strong generalization for bearing fault diagnosis. *Neurocomputing* 323, 62–75. doi: 10.1016/j.neucom.2018.09.050

Zuev, D., and Moore, A. W. (2005). "Traffic classification using a statistical approach," in *International Workshop on Passive and Active Network Measurement* (New York, NY), 321–324. doi: 10.1007/978-3-540-31966-5_25

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

frontiers in Neurorobotics

ORIGINAL RESEARCH
published: 13 April 2021
doi: 10.3389/fnbot.2021.652562

Geometrical Consistency Modeling on B-Spline Parameter Domain for 3D Face Reconstruction From Limited Number of Wild Images

Weilong Peng[1], Yong Su[2], Keke Tang[3], Chao Xu[4], Zhiyong Feng[4]* and Meie Fang[1]*

[1] School of Computer Science and Cyber Engineering, Guangzhou University, Guangzhou, China, [2] Tianjin Key Laboratory of Wireless Mobile Communications and Power Transmission, Tianjin Normal University, Tianjin, China, [3] Cyberspace Institute of Advanced Technology, Guangzhou University, Guangzhou, China, [4] College of Intelligence and Computing, Tianjin University, Tianjin, China

A number of methods have been proposed for face reconstruction from single/multiple image(s). However, it is still a challenge to do reconstruction for limited number of wild images, in which there exists complex different imaging conditions, various face appearance, and limited number of high-quality images. And most current mesh model based methods cannot generate high-quality face model because of the local mapping deviation in geometric optics and distortion error brought by discrete differential operation. In this paper, accurate geometrical consistency modeling on B-spline parameter domain is proposed to reconstruct high-quality face surface from the various images. The modeling is completely consistent with the law of geometric optics, and B-spline reduces the distortion during surface deformation. In our method, 0th- and 1st-order consistency of stereo are formulated based on low-rank texture structures and local normals, respectively, to approach the pinpoint geometric modeling for face reconstruction. A practical solution combining the two consistency as well as an iterative algorithm is proposed to optimize high-detailed B-spline face effectively. Extensive empirical evaluations on synthetic data and unconstrained data are conducted, and the experimental results demonstrate the effectiveness of our method on challenging scenario, e.g., limited number of images with different head poses, illuminations, and expressions.

Keywords: 3D face modeling, B-spline, face reconstruction, geometrical consistency, parametric domain

OPEN ACCESS

Edited by:
Wei Shi,
Carleton University, Canada

Reviewed by:
Xiaofei Xie,
Nanyang Technological University, Singapore
Weisheng Li,
Chongqing University of Posts and Telecommunications, China
Min Meng,
Guangdong University of Technology, China

****Correspondence:***
Keke Tang
tangbohutbh@gmail.com
Zhiyong Feng
zyfeng@tju.edu.cn

Received: *12 January 2021*
Accepted: *15 March 2021*
Published: *13 April 2021*

Citation:
Peng W, Su Y, Tang K, Xu C, Feng Z and Fang M (2021) Geometrical Consistency Modeling on B-Spline Parameter Domain for 3D Face Reconstruction From Limited Number of Wild Images. Front. Neurorobot. 15:652562. doi: 10.3389/fnbot.2021.652562

1. INTRODUCTION

3D face has been extensively applied in the areas of face recognition (Artificial and Aryananda, 2002; Mian et al., 2006), expression recognition (Zhang et al., 2015). These face analysis technologies are of significance for human-robot cooperative tasks in a safe and intelligent state (Maejima et al., 2012). So 3D face reconstruction is a import topic, and it is meaningful to reconstruct specific 3D face from person-of-interest images under many challenge scenes. The images under challenge scene are also referred as images in the wild, having following characteristics: (1) significant changes in illuminations across time periods; (2) various face poses caused by different camera sensors and view points; (3) different appearances among different

environment; (4) occlusions or redundant backgrounds. More seriously, only limited number of identity images are available under human-robot interaction, surveillance, and mobile shooting scenario as listed in **Figure 1**, sometimes.

As a whole, reconstruction technologies include single-image method, multiple images, and even unconstrained images based methods. Recent researches (Kemelmacher and Seitz, 2011; Roth et al., 2015, 2016) prove that good reconstruction depends on two aspects of efforts: (1) enough rich local information, e.g., normal, and (2) a good face prior, e.g., face template. Particularly, the latter is to find an embedding representation with good characteristic to register local information finely.

According to the template representation, these methods can be categorized into three classes: (i) methods without using template, e.g., integration (Kemelmacher and Seitz, 2011) and structure from motion (Koo and Lam, 2008), (ii) methods using a single discrete template, e.g., a reference face mesh (Roth et al., 2015), and (iii) methods using a statistic continuous template, e.g, T-splineMMs (Peng et al., 2017), or discrete template, e.g., 3DMMs (Piotraschke and Blanz, 2016; Roth et al., 2016). The methods with template always generate good global shape compared with those without template, and a statistic template contributes to a better personalization. Therefore, it is very significant to find a excellent template representation for face reconstruction. Mesh model is widely used due to its rapid computation and popularity in computer vision, but it is not well-compatible with geometric optics in vertex level, resulting in local mapping deviation of rays, seen in **Figure 1**. This makes local information not strictly registered physically. Additional, discretization of Laplace-Beltrami operation (LBO), i.e., cotangent scheme (Meyer et al., 2003), may bring a deformation distortion at local, which often happens when images are not enough for high-quality normal estimation. This distortion irregularly occurs at the edge and the location with large curvature changing, e.g., nose and mouth. Lastly the topology-fixed mesh also restricts an extended refinement. All above problem limits reconstruction precision of mesh.

To solve the existing issue in mesh template, we adopt classic B-spline embedding function (Piegl and Tiller, 1997) to register local information and reconstruct face. Firstly, B-spline surface is a parametric surface that can approximate the true shape of an object with fewer parameters (control points) than mesh. It contributes to correct rays in geometric optics, that makes local information, i.e., texture, feature points and normals, accurately registered. Secondly, we use 2nd-order partial derivative operator w.r.t. parameters as the local deformation constraint to reduce the deformation distortion. Lastly, B-spline surface also can be used to generate mesh in any precision or be extended for further refinement. The three characteristics of B-spline face show great advantages over a mesh template based method. Given a collection of images, we use B-spline embedding function as 3D face representation and model 0th- and 1st-order consistency of reconstruction in the parameter domain, which makes BP imaging rays completely compatible with geometric optics. The 0th-order consistency model guarantees that the images are well-registered to surface even if the face images has occlusion or expression; And the 1st-order consistency model guarantees that the surface normals is consistent to the normals estimated from images. Both qualitative and quantitative experiments are conducted and compared with other methods.

In a nutshell, there are two primary contributions:

1. Pinpoint geometrical consistency is modeled on B-spline embedding function for face reconstruction from multiple images, completely consistent with the law of geometric optics.
2. 0th- and 1st-order consistency conditions and its a practical solution is proposed to optimize B-spline face effectively, which is able to handle variations such as different poses, illuminations, and expressions with limited of number images.

In the following, we will first review related work in section 2. Section 3 provides a geometric modeling of multiple BP imaging in image-based stereo for our problem. We introduce the B-spline embedding and its brief representations in section 4 and present consistency modeling for B-spline face reconstruction in

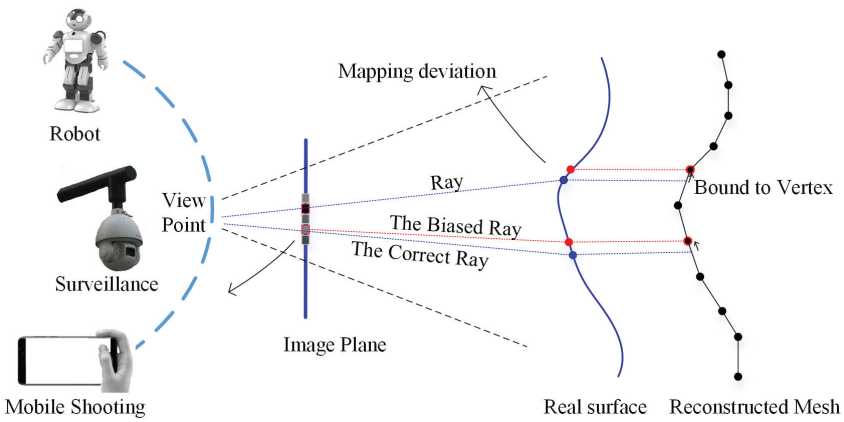


FIGURE 1 | Geometric optics of BP (i.e., back projection) imaging on two types of surfaces: the correct ray lines go through the blue points on the true shape, while the biased ones go through red points on the mesh shape because the cross point between a ray and mesh is bounded to vertex. The difference between red point and the blue point is referred to local mapping deviation.

section 5. In addition, a practical solution is proposed in section 6. We conduct experiment in section 7 and conclude in section 9.

2. RELATED WORK

2.1. 3D Face Required Scenes

With the development of robots and AIoT (Qiu et al., 2020), vision will play an very important role in safety (Khraisat et al., 2019; Li et al., 2019), scene and human understanding (Zhang et al., 2015; Meng et al., 2020). As a base technology, 3D face contributes to the scenes greatly. For example, to build humanoid robots that interact in a human-understanding manner, automatic face, and expression recognition is very import (Zhang et al., 2015). The recognition during real-life human robot interaction could still be challenging as a result of subject variations, illumination changes, various pose, background clutter, and occlusions (Mian et al., 2006). However, humanoid robot API of original version cannot always be able to handling such challenges. Optimal, robust, and accurate automatic face analysis is thus meaningful for the real-life applications since the performance of facial action and emotion recognition relies heavily on it. Many parametric approaches like 3DMMs (Blanz and Vetter, 1999; Blanz et al., 2004) and face alignment with 3D solution (Zhu et al., 2016) in the computer vision field have been proposed to estimate head pose, recognition identity, and expression from real-life images to benefit subsequent automatic facial behavior perception to address the above issues. Therefore, 3d face modeling in a humanoid robot view is of great significant to handling the challenging face analysis during interaction.

2.2. 2D Images Based Face Reconstruction

2D methods generally cover several kinds of fundamental methods including Structure from Motion (SFM) (Tomasi and Kanade, 1992), Shape from Shading (SFM) (Zhang et al., 1999), 3D Morphable Model (3DMM) (Blanz and Vetter, 1999; Blanz et al., 2004), and Deep learnings (Richardson et al., 2017; Deng et al., 2019). SFM methods compute the positions of surface points based on an assumption that there exists a coordinate transformation between the image coordinate system and the camera coordinate system. And SFS methods compute surface normals with an assumption that the subject surface is of Lambertian and under a relatively distant illumination. And the idea of 3DMM is that human faces are within a linear subspace, and that any novel face shape can be represented by a linear combination of shape eigenvectors deduced by PCA. SFS and SFM give the geometrical and physical descriptions of face shape and imaging, and 3DMM concentrates on the statistical explanation of 3D meshes or skeletons. Deep learning methods infer 3D face shape or texture (Lin et al., 2020) by statistically learning mapping between face images and their 3D shapes (Zhou et al., 2019). Being limited to data size, most of them relies 3DMM or PCA for synthesizing supplementary ground truths (Richardson et al., 2016) or as a priori (Tran et al., 2017; Gecer et al., 2019; Wu et al., 2019), resulting absence of shape detail. It's believed that face reconstruction is rather a geometrical optimization problem than a statistical problem, as 3DMM is more suitable to be an assistant of the geometrical method when building detailed shape, e.g., that by Yang et al. (2014).

2.3. Shape in Shading and Structure in Motion

SFS has been widely used for reconstruction, e.g., single-view reconstruction (Kemelmacher Shlizerman and Basri, 2011), multiple frontal images based reconstruction (Wang et al., 2003), and unconstrained image based reconstruction (Kemelmacher and Seitz, 2011; Roth et al., 2015). As single-view is ill posed (Prados and Faugeras, 2005), a reference is always needed (Kemelmacher Shlizerman and Basri, 2011). For unconstrained images, photometric stereo is applied to obtain accurate normals locally (Kemelmacher and Seitz, 2011; Roth et al., 2015). SFM uses multiple frame or images to recover sparse 3D structure of feature points of an object (Tomasi and Kanade, 1992). Spatial-transformation approach (Sun et al., 2013) only estimates the depth of facial points. Bundle adjustment (Agarwal et al., 2011) fits the large scale rigid object reconstruction, but it cannot generate the dense model of non-rigid face. Incremental SFM (Gonzalez-Mora et al., 2010) is proposed to build a generic 3D face model for non-rigid face. The work by Roth et al. (2015) optimizes the local information with normals from shading, based on a 3D feature points-driven global warping. Therefore, shading and motion are important and very distinct geometric information of face, and they enhance the reconstruction when being combined. In our method, 0th- and 1st-order consistency of stereo is modeled to integrate the advantages of both shading and motion information.

2.4. Facial Surface Modeling

Surface modeling is dependent on the data input (point cloud, noise, outlier, etc), output (point cloud, mesh, skeleton), and types of shape (man-made shape, organic shape). Point cloud, skeleton, and mesh grid are the widely used man-made shape type for face reconstruction. Lu et al. (2016) present an a stepwise tracking method approach to reconstruct 3D B-spline space curves from planar orthogonal views through minimizing the energy function with weight values. Spatial transformation method (Sun et al., 2013) estimates positions of sparse facial feature points. Bundle adjustment builds the dense point cloud for large scale rigid object with a great number of images (Agarwal et al., 2011). Heo and Savvides (2009) reconstruct face dense mesh based on skeleton and 3DMM. Kemelmacher and Seitz (2011) apply integration of normals to get discrete surface points, which may produce incredible depth when the recovered normals are unreliable. Roth et al. (2015) reconstruct face mesh based on Laplace mesh editing, which may produce local mesh distortion after several iterations of local optimization. In work of mesh reconstruction, surface-smoothness priors is also needed to guarantee the smoothness of discrete mesh based on point cloud, e.g., radial basis function (Carr et al., 2001) and Poisson surface reconstruction (Kazhdan et al., 2006). Due to the fact that the point cloud and 3D mesh are discontinuous geometric shape, they cannot approximate the true shape of a face of arbitrary precision. There have been works of fitting B-splines to noisy 3D data, like Hoch et al. (1998). B-spline face model is

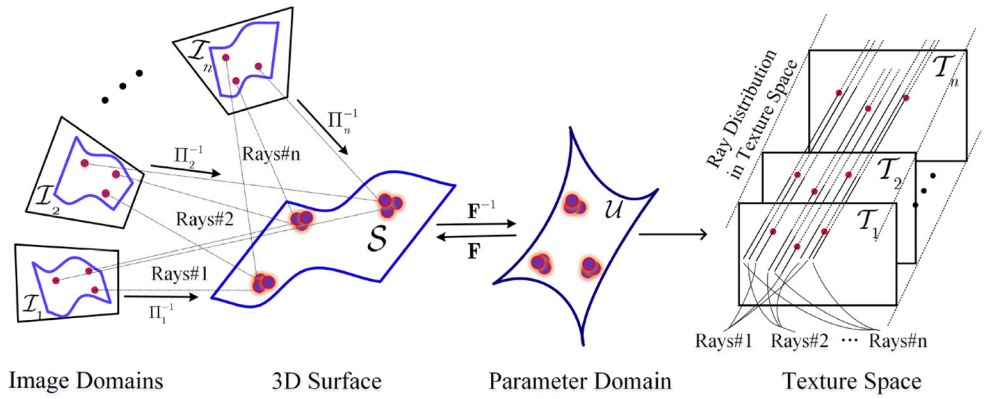


FIGURE 2 | Geometric optics of multiple BP imaging.

a continuous free-form surface that can be reconstructed from images directly, instead of intermediate point data, but it is not a detailed model by only using structure optimization (Peng et al., 2016). Because B-spline surface is a special case of NURBS (Non-Uniform Rational B-Spline) (Piegl and Tiller, 1997), it can also be imported to 3D modeling software like Rhino3D for further editing, analysis, and transformation conveniently by adjusting the B-spline control points. It can also be converted into mesh model with any precision according to appropriate parameter interval, conveniently, which is meaningful for a system with limited memory.

3. GEOMETRIC MODELING

Our problem modeling is illustrated in **Figure 2**. The domain of input image I_i from a camera is $\mathcal{I}_i \subset \mathbb{R}^2, i = 1, 2, \ldots, n$. Π^{-1} denotes the inverse operator of Π. The camera operator $\Pi_i \in C^\infty(\mathbb{R}^3, \mathbb{R}^2)$ map a point $P \in \mathcal{S}$ to $p = \Pi_i(P) \in \mathcal{I}_i$ using weak perspective projection, $i = 1, 2, \ldots, n$. And Π_i^{-1} determines the ray cluster Rays#i of BP imaging from $\mathcal{I}_i, i = 1, 2, \ldots, n$. Let s_i, R_i, and t_i denote scale, rotation, and translation parameter in projection Π_i. The ith projection operation is simply

$$\Pi_i(P) \triangleq s_i \cdot R_{i,[1,2]} \cdot P + t_i. \tag{1}$$

$R_{i,[1,2]}$ expresses the first two rows of R_i.

Let $\mathcal{U} \subset \mathbb{R}^2$ denote the parameter domain of human face surface. A certain embedding $\mathbf{F} \in C^1(\mathcal{U}, \mathbb{R}^3)$ maps a point $\mathbf{u} \in \mathcal{U}$ to the 3D point $P \in \mathcal{S}$. $\mathbf{F}^{-1}$ denote the inverse operator of $\mathbf{F}$. It is thus clear that different embedding F determine different face shapes. According to the geometric optics of BP imaging, a image point $p \in \mathcal{I}_i$ is back projected onto a point $\mathbf{u} = \tau_i(p) \in \mathcal{U}$ *via* the operator

$$\tau_i \triangleq \mathbf{F}^{-1} \circ \Pi_i^{-1}. \tag{2}$$

Therefore, an image I_i in the i-th view is mapped to surface S, and then is mapped to texture space by

$$\mathcal{T}_i \triangleq I_i \circ \tau_i^{-1}, \tag{3}$$

where we define

$$(I \circ \tau^{-1})(\mathbf{u}) \triangleq I(\Pi(\mathbf{F}(\mathbf{u}))), \quad for \ \mathbf{u} \in \mathcal{U}. \tag{4}$$

In fact, $\tau_i, i = 1, 2, \ldots, n$ generate discrete and inconsistent rays mapping in texture space because of the discrete and different images domains, as well as the noises, seen in **Figure 2**.

3.1. 0th- and 1st-Order Consistency

Generally, the problem is how to determine $\mathbf{F}$ according to from multiple images. If all images are the captures of a same $\mathcal{S}$, all $\{\mathcal{T}_i\}_{i=1:n}$ in texture space are hoped to be highly consistent in the geometry.

First, that satisfies

$$< \hat{\mathbf{F}}, \{\hat{\Pi}_i\} >= \arg\min_{\mathbf{F}, \{\Pi_i\}} \text{rank}([\text{vec}(\mathcal{T}_1), \text{vec}(\mathcal{T}_2), \ldots, \text{vec}(\mathcal{T}_n)]), \tag{5}$$

with $\mathcal{T}_i = (I_i \circ \tau_i^{-1})^\#, i = 1, 2, \ldots, n$. And $(\cdot)^\#$ is a composition operator of fitting and sampling, to handle the inconsistency. It firstly fits a texture function based on the discrete texture and parameters mapped from one image, and then samples texture intensity values at unified parameter points $\{\mathbf{u}_j\}_{j=1:N_p}$.

Second, it satisfies

$$\begin{cases} \frac{\frac{\partial \mathbf{F}}{\partial u} \times \frac{\partial \mathbf{F}}{\partial v}}{\left\| \frac{\partial \mathbf{F}}{\partial u} \times \frac{\partial \mathbf{F}}{\partial v} \right\|} = \boldsymbol{n}, \\ \rho_j \boldsymbol{n}_j \cdot \boldsymbol{l}_i = \mathcal{T}_i|_{\mathbf{u}_j}. \end{cases} \tag{6}$$

which describes the equivalence relation between normal $\boldsymbol{n}$ and 1st-order partial derivative in the first formulation, and the equivalence relation among albedo ρ, normal $\boldsymbol{n}$, light direction $\boldsymbol{l}$, and image intensity $\mathcal{T}$ in the second. This follows a linear photometric model, as seen in **Figure 3**.

We refer to Equations (5) and (6) as 0th- and 1st- order consistence equations in 3D surface reconstruction respectively. Generally, researchers solve any one of the two consistence problem to reconstruct 3D surface, classically, by multi-view stereo (Seitz et al., 2006) for 0th-order consistence problem, or by photometric stereo (Barsky and Petrou, 2003) for the 1st-order one.

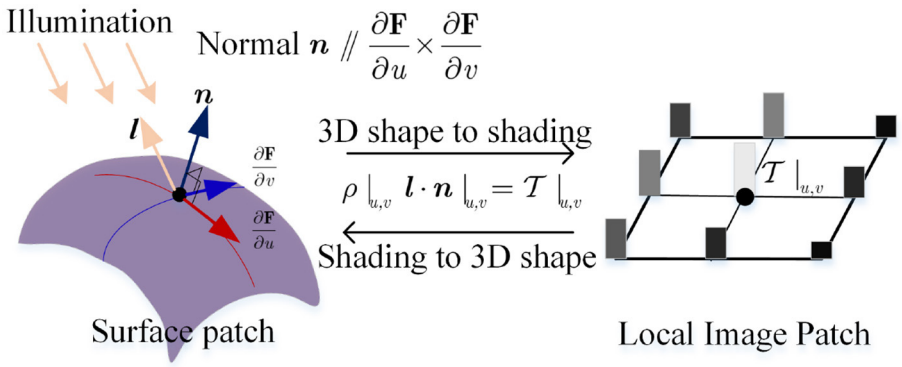


FIGURE 3 | The consistency mapping equivalence between embedding **F** and the image intensity.

3.2. Embedding F

There are several types of representation for embedding **F**, such as discrete mesh and C^2 parametric surface. In fact the representation type of **F** also affects the reconstruction effect. Intuitively for mesh, on one hand there exists mapping deviation of rays from image points to vertices of mesh, which contributes to inaccurate texture charts $\{\mathcal{T}_i\}_{i=1:n}$ and affects the accuracy of reconstruction. On the other, discrete differential operator, i.e., LBO (Meyer et al., 2003), brings potential distortion error when there exists obtuse triangles in the mesh caused by error local normal. Additionally, the precision of mesh also limit the detail of reconstruction.

We consider to apply C^2 parametric surface as the representation of face. Generally, B-spline surface is recommended because of its advantages of good locality over other types of surfaces such as polynomial surface and Bessel surface. By B-spline surface, it doesn't exist mapping deviation in geometric optics, and it avoids the potential distortion brought by discrete differential operator. Therefore, accurate and continuous back projection texture charts $\{\mathcal{T}_i\}_{i=1:n}$ can be generated based on Equations (2), (3), and (5). Then accurate reconstruction can be implemented based on Equation (6). What's more, the precision can be enhanced for high-detailed reconstruction by inserting control points.

4. B-SPLINE FACE EMBEDDING F, AND THE 0TH-, 1ST-, 2ND–ORDER REPRESENTATION

The human face is assumed to be a uniform B-spline surface $\mathcal{S}$ of degree 4×4, with $\mathbf{B} = \{\boldsymbol{b}_{mn}\}_{M\times N}$ as its control points. In parameter domain $\mathcal{U}$, knots $U = \{u_m\}_{m=1}^{M+4}$ and $V = \{v_n\}_{n=1}^{N+4}$ split uv parameter plane into uniform grid. Let $\mathbf{u}$ denote parameter point (u, v). The surface function is

$$\mathbf{F}(\mathbf{u}) = \sum_{m=1}^{M}\sum_{n=1}^{N} R_{m,n}(\mathbf{u})\boldsymbol{b}_{mn},$$

with $R_{m,n}(\mathbf{u}) = N_{m,4}(u) \cdot N_{n,4}(v)$ and

$$\begin{cases} N_{i,1}(w) = \begin{cases} 1 & u_i \le w < u_{i+1}, \\ 0 & otherwise, \end{cases} \\ N_{i,j}(w) = \frac{(w-u_i)\cdot N_{i,j-1}(w)}{u_{i+j-1}-u_i} + \frac{(u_{i+j}-w)\cdot N_{i+1,j-1}(w)}{u_{i+j}-u_{i+1}}, (j = 4, 3, 2). \end{cases}$$

F is C^2, meaning that it can approximate the true shape in arbitrary uv precision with deterministic k-ordered partial derivative $\frac{\partial^k \mathbf{F}}{\partial u^k}$ and $\frac{\partial^k \mathbf{F}}{\partial v^k}$, $k = 1, 2$, and $\frac{\partial^2 \mathbf{F}}{\partial u \partial v}$.

4.1. 0th-Order Representation

We give a more brief formulation of 0th-order representation as follows:

$$\mathbf{F}|_{\mathbf{u}} = \mathbf{T}|_{\mathbf{u}} \cdot \mathbf{b}, \tag{7}$$

where $\mathbf{b}$ denotes a $3MN \times 1$ vector storing B-spline control points, and $\mathbf{T}|_{\mathbf{u}}$ denotes a sparse $3 \times 3MN$ matrix stacking the 0th-order coefficients at parameter $\mathbf{u} \in \mathcal{U}$.

In fact, we needn't consider all 3D points mapping to 2D images when estimating a operator Π. Instead, we only consider f landmark points on human face as shown in **Figure 4**, and their brief formulation is

$$\mathbf{F}|_{\mathbf{u}(l_i)} = \mathbf{T}|_{\mathbf{u}(l_i)} \cdot \boldsymbol{b}, i = 1, 2, \ldots, f, \tag{8}$$

where $\mathbf{u}(l_i)$ is the parameter point of the i-th feature point, $i = 1, 2, \ldots, f$. The landmarks cover a sparse structure of face.

4.2. 1st-Order Representation

The 1st-order partial derivatives of **F** w.r.t u and v are

$$\begin{aligned} \mathbf{F}'_u(\mathbf{u}) &= \sum_{m=1}^{M}\sum_{n=1}^{N} N'_{m,4}(u) \cdot N_{n,4}(v)\boldsymbol{b}_{mn} \\ &= \sum_{m=1}^{M}\sum_{n=1}^{N} \left(\frac{4}{u_{m+4}-u_i}N'_{m,3}(u) - \frac{4}{u_{m+5}-u_{m+1}}N'_{m+1,3}(u)\right) \cdot N_{n,4}(v)\boldsymbol{b}_{mn} \end{aligned}$$

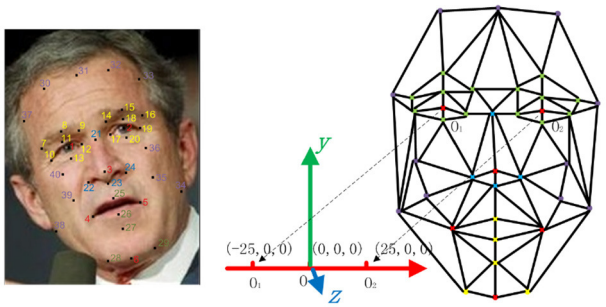


FIGURE 4 | Face structure defined by 40 feature points: the left side shows the point positions in a face image; the right side shows the structure topology with eye center points of O_1 (−25, 0, 0) and O_2 (25, 0, 0) in 3*D* space, which looks like a frontal 2D face structure from the direction of normal (0, 0, 1). (The face image used in the figure comes from LFW database[1]).

and

$$\mathbf{F}'_v(\mathbf{u}) = \sum_{m=1}^{M}\sum_{n=1}^{N} N_{m,4}(u)\cdot N'_{n,4}(v)\boldsymbol{b}_{mn}$$
$$= \sum_{m=1}^{M}\sum_{n=1}^{N} N_{m,4}(u)\cdot\left(\frac{4}{u_{n+4}-u_n}N'_{n,3}(v) - \frac{4}{u_{n+5}-u_{n+1}}N'_{n+1,3}(v)\right)\boldsymbol{b}_{mn}$$

respectively.

Similarly, we give a more brief formulation of 1st-order partial derivative as follows:

$$\begin{cases} \mathbf{F}'_u|_{\mathbf{u}} = \mathbf{T}_1|_{\mathbf{u}}\cdot\mathbf{b} \\ \mathbf{F}'_v|_{\mathbf{u}} = \mathbf{T}_2|_{\mathbf{u}}\cdot\mathbf{b} \end{cases}, \tag{9}$$

where $\mathbf{T}_1|_{\mathbf{u}}$ and $\mathbf{T}_2|_{\mathbf{u}}$ denote the matrixes stacking the 1st-order coefficients w.r.t u and v, respectively.

Therefore, the surface normal vector at $\mathbf{u}$ can be computed by the cross product

$$\boldsymbol{n}|_{\mathbf{u}} = \frac{\boldsymbol{F}'_u|_{\mathbf{u}}\times\boldsymbol{F}'_v|_{\mathbf{u}}}{\left\|\boldsymbol{F}'_u|_{\mathbf{u}}\times\boldsymbol{F}'_v|_{\mathbf{u}}\right\|} = s|_{\mathbf{u}}\cdot\boldsymbol{F}'_u|_{\mathbf{u}}\times\boldsymbol{F}'_v|_{\mathbf{u}}, \tag{10}$$

which is the key information for detailed reconstruction using photometric stereo method.

4.3. 2nd-Order Representation

And similarly, the 2nd-order partial derivatives w.r.t. u and v, respectively are

$$\begin{cases} \mathbf{F}''_{uu}|_{\mathbf{u}} = \mathbf{T}_{11}|_{\mathbf{u}}\cdot\mathbf{b} \\ \mathbf{F}''_{vv}|_{\mathbf{u}} = \mathbf{T}_{22}|_{\mathbf{u}}\cdot\mathbf{b} \end{cases}, \tag{11}$$

where $\mathbf{T}_{11}|_{\mathbf{u}}$ and $\mathbf{T}_{22}|_{\mathbf{u}}$ denote the matrixes stacking the 2nd-order coefficients w.r.t u and v, respectively. The 2nd-order information can be used for smooth control during optimization.

Based on face surface embedded with B-spline function, we present the pinpoint 0th- and 1st-order geometric consistency conditions in the following section.

5. CONSISTENCY MODELING IN B-SPLINE FACE RECONSTRUCTION

Reconstruction problem is to compute $\mathbf{F}$ by solving 0th-order consistence of Equation (5) or 1st-order consistence of Equation (6). Generally, two consistency conditions are combined for face reconstruction considering that estimating abundant consistent points in images is limited and that the estimated normals are unfaithful. Furthermore, how to obtain the accurate registration of 0th- and 1st-order information is the most important to high-detailed B-spline reconstruction.

The well-registered textures are low-rank structures of the back projection texture charts. But in practice, they can be easily violated due to the presence of partial occlusions or expressions in the images captured. Since these errors typically affect only a small fraction of all pixels in an chart, they can be modeled as sparse errors whose nonzero entries can have arbitrarily large magnitude.

5.1. Modeling Occlusion and Expression Corruptions in 0th-Order Consistence

Let e_i represent the error corresponding to image I_i such that the back projection texture charts $\mathcal{T}_i = (I_i\circ\tau_i^{-1})^{\#} - e_i = \mathcal{T}_i^e - e_i, i = 1, 2, \ldots, n$ are well registered to the surface $\mathbf{F}$, and free of any corruptions or expressions. Also combining with 0th-order representation of B-spline face in Equation (7), the formulation (5) can be modified as follows:

$$\begin{aligned} <\hat{\mathbf{b}}, \{\hat{\Pi}_i\}, \hat{\mathbf{D}}, \hat{\mathbf{E}}> = \arg\lim_{\mathbf{b},\{\Pi_i\},\mathbf{D},\mathbf{E}} \|\mathbf{D}\|_* + \eta\|\mathbf{E}\|_1, \\ s.t.\ \left\|\mathbf{D}^e - \mathbf{D} - \mathbf{E}\right\|_F \le \varepsilon. \end{aligned} \tag{12}$$

where $\mathbf{D}^e = \left[\mathrm{vec}(\mathcal{T}_1^e), \mathrm{vec}(\mathcal{T}_2^e), \ldots, \mathrm{vec}(\mathcal{T}_n^e)\right]$ and $\mathbf{E} = [\mathrm{vec}(e_1), \mathrm{vec}(e_2), \ldots, \mathrm{vec}(e_n)]$.

However, the solution $\hat{\mathbf{b}}$ of face surface $\mathcal{S}$ is not unique if all images are in similar views. And the reconstruction is not high-detailed even if we can make a unique solution by applying a prior face template. So we also need to model high details in 1st-order consistence.

5.2. Modeling High Details in 1st-Order Consistence

The resolution of reconstruction is determined by the density of correctly estimated normals. To enhance the resolution of B-spline surface, we use operator $(\cdot)^{\#}$ to sample N_p dense parameter points $\{\mathbf{u}_j\}_{j=1:N_p}$ on the domain $\mathcal{U}$ for the problem of Equation (6).

Then the well-registered and dense texture are obtained by

$$\mathcal{T}_i|_{\mathbf{u}_j} = \hat{\mathbf{D}}_{ji}, \tag{13}$$

for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, N_p$.

According to Lambertian illumination model seen in Equation (6), dense normals $\boldsymbol{n}_j$ as well as light $\boldsymbol{l}_i$ can be computed from the shading (intensity) of charts $\mathcal{T}_i$ by SVD method.

Finally, the high detailed reconstruction must satisfy

$$\min_{\mathbf{F}}\sum_{j=1}^{N_p}\left\|\boldsymbol{n}_j - s|_{\mathbf{u}_j}\mathbf{F}'_u|_{\mathbf{u}_j}\times\mathbf{F}'_v|_{\mathbf{u}_j}\right\|_2^2. \tag{14}$$

By putting Equation (9) into Equation (14), we get

$$\min_{\mathbf{b}} \sum_{j=1}^{N_p} \left\| \boldsymbol{n}_j - s|_{\mathbf{u}_j}(\mathbf{T}_1|_{\mathbf{u}_j} \cdot \mathbf{b}) \times (\mathbf{T}_2|_{\mathbf{u}_j} \cdot \mathbf{b}) \right\|_2^2. \quad (15)$$

Conditions of both Equations (6) and (15) have to be considered for a good reconstruction, which is very difficult. Therefore, we propose a practical solution that combining both 0th- and 1st-order consistence.

6. PRACTICAL SOLUTION COMBINING 0TH- AND 1ST-ORDER CONSISTENCE

The problems of both 0th-order consistence and 1st-order consistence are difficult to solve. For , Jacobian matrices w.r.t. $\{\tau_i^{-1}\}_{i=1:n}$ have to be computed, which is computing-expensive. And the solution of Equation (15) is not unique, either. Therefore, we aim to find a practical solution to handle both two consistence conditions in this section. We first define the subproblem for each condition, and then provide a iterative algorithm.

6.1. 0th-Order Solution

In Equation (6), three kind of parameters including camera parameters $\{\Pi_i\}_{i=1:n}$, surface parameters $\mathbf{F}$ (or $\mathbf{b}$), and texture parameters $\{\mathcal{T}_i\}_{i=1:n}$ (or $\mathbf{D}$) need to be computed, but they are difficult to be solved simultaneously. We adopt to optimize them by turns, instead.

6.1.1. Estimating Π_i

According to linear transformation from 3D to 2D in Equation (1), we can estimate scale s_i, rotation R_i and translationt_i of landmarks for each image $I_i, i = 1, 2, \ldots, n$ based on the and SVD method (Kemelmacher and Seitz, 2011). The image landmarks are detected by a state-of-art detector (Burgos-Artizzu et al., 2013) that has a similar high performance to human. And the 3D landmarks are defined on a B-spline face template with control point parameter $\mathbf{b}_0$, according to Equation (8).

6.1.2. Estimating b

Let $\mathbf{f}$ denote a $2nf \times 1$ vector stacking f landmarks of n images, and $\mathbf{P}$ denote a $2nf \times 3f$ projection matrix stacking n views of parameters $s_i R_{i,[1,2]}$, and $\mathbf{t}$ denote a $2nf \times 1$ vector stacking f translation. The update of $\mathbf{b}$ can be implemented by solving:

$$\min_{\boldsymbol{b}} \left\| \mathbf{f} - \mathbf{t} - \mathbf{P} \cdot \mathbf{T}^{\#l}\mathbf{b} \right\|_2^2 + \zeta \left\| (\mathbf{T}_{11}^{\#} + \mathbf{T}_{22}^{\#})(\mathbf{b} - \mathbf{b}_0) \right\|_2^2 \quad (16)$$

where the first and the second are 0th- and 2nd-order item, respectively, and ζ is used to balance them. Operator $(\cdot)^{\#l}$ is a sampling operator that selects B-spline coefficients of landmarks at parameters $\{\mathbf{u}(l_i)\}_{i=1:f}$, and $(\cdot)^{\#}$ selects B-spline coefficients at $\{\mathbf{u}_j\}_{i=1:N_p}$. In fact, $\mathbf{T}^{\#l}$ is a $3f \times 3MN$ matrix that stacks $\mathbf{T}|_{\mathbf{u}(l_i)}, i = 1, 2, \ldots, f$, and $\mathbf{T}_{11}^{\#}$ (or $\mathbf{T}_{22}^{\#}$) is a $3f \times 3MN$ matrix that stacks $\mathbf{T}_{11}|_{\mathbf{u}_j}$ (or $\mathbf{T}_{22}|_{\mathbf{u}_j}$), $j = 1, 2, \ldots, N_p$.

The second item also work as a regularization measuring the distance of local information between faces $\mathbf{b}$ and $\mathbf{b}_0$. It helps eliminate affect of geometric rotation brought by 0st-order warping, and guarantee a smoothness changing during optimization. Particularly, ζ cannot be too small, otherwise a fast changing may bring a local optimal.

6.1.3. Estimating $\mathcal{T}_i$

τ_i^{-1} and τ_i is determined by Equation (2) when Π_i and $\mathbf{b}$ is known. Then texture chart with noise is obtained by applying consistent parameter sampling $\mathcal{T}_i^e = (I_i \circ \tau_i^{-1})^{\#}$. Let $\mathbf{D}^e = [\mathrm{vec}(\mathcal{T}_1^e), \mathrm{vec}(\mathcal{T}_2^e), \ldots, \mathrm{vec}(\mathcal{T}_n^e)]$. The update of texture charts is to minimize the following formulation

$$\begin{aligned} &< \hat{\mathbf{D}}, \hat{\mathbf{E}} >= \arg\lim_{\mathbf{D},\mathbf{E}} \|\mathbf{D}\|_* + \eta \|\mathbf{E}\|_1, \\ &s.t. \|\mathbf{D}^e - \mathbf{D} - \mathbf{E}\|_F \le \varepsilon. \end{aligned} \quad (17)$$

which can be solved by Robust PCA (Bhardwaj and Raman, 2016). And let $\mathcal{T}_i|_{\mathbf{u}_j} = \hat{\mathbf{D}}_{ji}$, for $i = 1, 2, \ldots, n$, and $j = 1, 2, \ldots, N_p$.

6.2. 1st-Order Solution

Firstly, texture charts based photometric stereo method is used to estimate the local normals. Secondly, a normals driven optimization strategy is proposed to optimize the B-spline face.

6.2.1. Estimating n_j

According to Photometric stereo, the shape of each point can be solved by the observed variation in shading of the images. Data of n texture charts are input into $M_{n \times N_p}$ for estimating the initial shape $\tilde{S}$ and lighting $\tilde{L}$ by factorizing $M = LS$ *via* SVD (Yuille et al., 1999). $\tilde{L} = U\sqrt{\Sigma}$ and $\tilde{S} = \sqrt{\Sigma}V^T$, where $M = U\Sigma V^T$. To approach the true normal information, we estimate the shape S and ambiguity A by following the work of Kemelmacher and Seitz (2011). Lastly, the normal at j-th point is $\boldsymbol{n}_j = S_j^T$, where S_j is the j-th row of S.

6.2.2. Estimating b

We normalize $\boldsymbol{n}_j$ and stack them into a $3N_p \times 1$ vector $\mathbf{h}$. Equation (15) can be rewritten as

$$\mathbf{O}_1 = \min_{\mathbf{b}} \left\| \mathbf{h} - \Lambda|_{\mathbf{b}} \cdot ((\mathbf{T}_1^{\#}\mathbf{b}) \otimes ((\mathbf{T}_2^{\#}\mathbf{b})) \right\|_2^2,$$

where Λ is a $3N_p \times 3N_p$ diagonal matrix that stores $3N_p$ reciprocals of lengths of the normals $\{\boldsymbol{n}_j\}_{j=1:N_p}$; and $(\cdot)^{\#}$ is a selection operator that selects $3N_p$ rows of 1st-order coefficients at parameter $\{\mathbf{u}_j\}_{j=1:N_p}$; and $\mathbf{b}_0$ represent the control points of a B-spline template face. Particularly, symbol $\otimes$ denotes a composite operator of cross product, which makes $\boldsymbol{w} \otimes \boldsymbol{v} = [\boldsymbol{w}_1 \times \boldsymbol{v}_1; \boldsymbol{w}_2 \times \boldsymbol{v}_2; \ldots; \boldsymbol{w}_{N_p} \times \boldsymbol{v}_{N_p}]$, where $\boldsymbol{w}$ and $\boldsymbol{v}$ are $3N_p \times 1$ vectors containing N_p normals.

However, there exists two issues: (1) the low-dimension $\boldsymbol{h}$ may not guarantee an unique solution of high-dimension $\boldsymbol{b}$; and (2) the system is not simply linear, which is difficult to be solved. Therefore, a frontal constraint based on template $\mathbf{b}_0$ is applied to make a unique solution; And a strategy of approximating to linearization is also proposed to make a linear solution.

6.2.2.1. Frontal Constraint

The frontal constraint is a distance measurement condition between surface $\mathcal{S}$ and template w.r.t. x- and y-component:

$$\mathbf{O}_2 = \left\| \mathbf{T}^{\#xy}(\mathbf{b} - \mathbf{b}_0) \right\|_2^2 < \epsilon,$$

where the matrix $\mathbf{T}^{\#xy}$ stacks 0th-order coefficients at parameter $\{\mathbf{u}_j\}_{j=1:N_p}$ corresponding to x- and y- components. Operator $(\cdot)^{\#sxy}$ also sets the coefficients corresponding to z- components to zeros.

Particularly, the first item $\boldsymbol{O}_1$ is not a simple linear form, for which an approximating to linearization is proposed.

6.2.2.2. Approximating to Linearization

According to the characteristics of the cross-product $\otimes$, the first item in $\boldsymbol{O}_1$ can be rewritten as a linear-like formulation:

$$\|\mathbf{h} - \boldsymbol{L}|_{\mathbf{b}} \cdot \mathbf{b}\|_2^2 \quad \text{or} \quad \|\mathbf{h} - \boldsymbol{R}|_{\mathbf{b}} \cdot \mathbf{b}\|_2^2,$$

where $\begin{cases} \boldsymbol{L}|_{\mathbf{b}} = \Lambda|_{\mathbf{b}} \cdot \left[\mathbf{T}_1^{\#}\mathbf{b}\right]_{\otimes} \cdot \mathbf{T}_2^{\#} \\ \boldsymbol{R}|_{\mathbf{b}} = -\Lambda|_{\mathbf{b}} \cdot \left[\mathbf{T}_2^{\#}\mathbf{b}\right]_{\otimes} \cdot \mathbf{T}_1^{(sn)}. \end{cases}$

Particularly, the operation $[\cdot]_{\otimes}$ makes a $3N_p \times 1$ vector $\boldsymbol{w} = \left[\boldsymbol{w}_1^T, \boldsymbol{w}_2^T, \ldots, \boldsymbol{w}_{N_p}^T\right]^T$ become a $3N_p \times 3N_p$ sparse matrix $[\boldsymbol{w}]_{\otimes} = diag([\boldsymbol{w}_1]_{\times}, [\boldsymbol{w}_2]_{\times}, \ldots, \left[\boldsymbol{w}_{N_p}\right]_{\times})$, where $[\boldsymbol{w}_i]_{\times} = [0, -\boldsymbol{w}_i^z, \boldsymbol{w}_i^y; \boldsymbol{w}_i^z, 0, -\boldsymbol{w}_i^x; -\boldsymbol{w}_i^y, \boldsymbol{w}_i^x, 0], i = 1, 2, \ldots, N_p$.

If $\mathbf{b}$ is a known parameter, e.g., as $\mathbf{b}_0$, for $\boldsymbol{L}|_{\mathbf{b}}$, the minimization of $\left\|\mathbf{h} - \boldsymbol{L}|_{\mathbf{b}0} \cdot \mathbf{b}\right\|$ will be a linear system. That is also true for $\boldsymbol{R}|_{\mathbf{b}}$.

In fact, we can use formulation $\left\|\mathbf{h} - \boldsymbol{L}|_{\mathbf{b}0} \cdot \mathbf{b}\right\|$ to optimize the control points in parameter space of v by fixing u, and use $\left\|\mathbf{h} - \boldsymbol{R}|_{\mathbf{b}0} \cdot \mathbf{b}\right\|$ to optimize in parameter space of u by fixing v.

Algorithm 1: Iterative Algorithm for B-spline Face Optimization

Input: Face images $\{I_i\}_{i=1:n}$, B-spline template face $\mathbf{b}_0$, and landmark parameters $\{\mathbf{u}(l_i)\}_{i=1:f}$ in domain $\mathcal{U}$.

```
Detect facial landmark points of images
while b is not converged do
    // LOOP1: 0th-order consistence
    Estimate camera parameter {Π_i}_{i=1:n} according to landmarks.
    Estimate b via Equ(16), and update b_0 with b.
    // Obtain well-registered texture
    Register images to texture space by {I_i ∘ τ_i^{-1}}_{i=1:n}, and build D^e based on unified parameter {u_j}_{j=1:N_p}.
    Solve Equ(17) to obtain D̂.
    while ||D||_* + η ||E||_1 is not converged // LOOP1 END
    Extract texture charts {T_i}_{i=1:n} from D̂.
    while b_0 is not converged // LOOP2: 1st-order consistence do
        Estimate normals {n_j} from {T_i}.
        Estimate b via Equation (18.a), and update b_0 with b.
        Estimate b via Equation (18.b), and update b_0 with b.
    end while
end while
```

Output: Solution of B-spline objective face $\mathbf{b}$.

A practical skill is to optimize the control points on u and v parameter spaces by turns. The two iteration items are rewritten as

$$\begin{cases} \left\|\mathbf{h} - \boldsymbol{L}|_{\mathbf{b}_0} \cdot \mathbf{b}\right\|_2^2 + \lambda \left\|\Lambda_1|_{\mathbf{b}_0} \cdot \mathbf{T}_1^{\#} \cdot (\mathbf{b} - \mathbf{b}_0)\right\|_2^2, \\ \left\|\mathbf{h} - \boldsymbol{R}|_{\mathbf{b}_0} \cdot \mathbf{b}\right\|_2^2 + \lambda \left\|\Lambda_2|_{\mathbf{b}_0} \cdot \mathbf{T}_2^{\#} \cdot (\mathbf{b} - \mathbf{b}_0)\right\|_2^2. \end{cases}$$

where the second term for each formulation is unit tangent vector constraint on the fixed the directions. $\Lambda_1|_{\mathbf{b}_0}$ (or $\Lambda_2|_{\mathbf{b}_0}$) is a $3N_p \times 3N_p$ diagonal matrix that stores $3N_p$ reciprocals of lengths of tangent vector $\frac{\partial \mathbf{F}}{\partial u}$ (or $\frac{\partial \mathbf{F}}{\partial v}$) at $\{\mathbf{u}_j\}_{j=1:N_p}$. During this procedure $\mathbf{b}_0$ is updated step-by-step. As shown in **Figure 5**, two partial derivatives $\frac{\partial \mathbf{F}}{\partial v}$ and $\frac{\partial \mathbf{F}}{\partial u}$ at (u, v) are updated until $\frac{\partial \mathbf{F}}{\partial v} \times \frac{\partial \mathbf{F}}{\partial u}$ converges to $\boldsymbol{n}$.

By integrating with $\mathbf{O}_2$, the final formulation of optimization consists of two items as follows:

$$\begin{cases} \min\limits_{\mathbf{b}} \left\| \begin{bmatrix} \mathbf{h} \\ \mathbf{T}^{\#xy}\mathbf{b}_0 \end{bmatrix} - \begin{bmatrix} \boldsymbol{L}|_{\mathbf{b}_0} \\ \mathbf{T}^{\#xy} \end{bmatrix} \mathbf{b} \right\|_2^2 + \lambda \left\|\Lambda_1|_{\mathbf{b}_0} \cdot \mathbf{T}_1^{\#}(\mathbf{b} - \mathbf{b}_0)\right\|_2^2, \ (a) \\ \min\limits_{\mathbf{b}} \left\| \begin{bmatrix} \mathbf{h} \\ \mathbf{T}^{\#xy}\mathbf{b}_0 \end{bmatrix} - \begin{bmatrix} \boldsymbol{R}|_{b_0} \\ \mathbf{T}^{\#xy} \end{bmatrix} \mathbf{b} \right\|_2^2 + \lambda \left\|\Lambda_2|_{\mathbf{b}_0} \cdot \mathbf{T}_2^{\#}(\mathbf{b} - \mathbf{b}_0)\right\|_2^2. \ (b) \end{cases} \tag{18}$$

The $\mathbf{b}_0$ is initialized by value of $\mathbf{b}_0$. Then we can solve $\mathbf{b}$ and update $\mathbf{b}_0$ orderly by minimizing (a) and (b) in Equation (18) iteratively until convergence.

6.3. Algorithm

An iterative algorithm is presented for this practical solution in Algorithm 1. Processes of 0th-order consistence and 1st-order consistence are separately conducted in the inner loop. And the outer loop guarantees a global convergence on two consistence problem.

6.3.1. Computational Complexity

The computation in above Algorithm 1 involves linear least square for solving Equations (16), (18.a), and (18.b), SVD for estimating camera parameter, and Robust PCA for Equation (17). In detail, the computational complexity for solving Equation (16) is $O(n^2f^2MN)$, and that of both Equations (18.a) and (18.b) are $O(N_p^2MN)$. The computational complexity of robust PCA comes to be $O(N_p^2k)$, where k is the rank constraint. By assuming $N_p > M > N >> f > n$, computational complexity of the other parts can be negligible. In addition, we need considering the number of iteration for total computation of Algorithm 1.

7. EXPERIMENT

In this section experiments are presented to verify our automatic free-form surface modeling method. We first describe the pipeline to prepare a collection of face images of a person for B-spline face reconstruction. And then we demonstrate the quantitative and qualitative comparisons with recent baseline methods on projected standard images from ground truth 3D data (Zhang et al., 2017) with various expressions, illuminations and poses. Finally, we conduct challenging reconstructions and comparison based on real unconstrained data taken from the challenging Labeled Faces in Wild (LFW) database[1] (Huang et al., 2007).

[1] http://vis-www.cs.umass.edu/lfw/

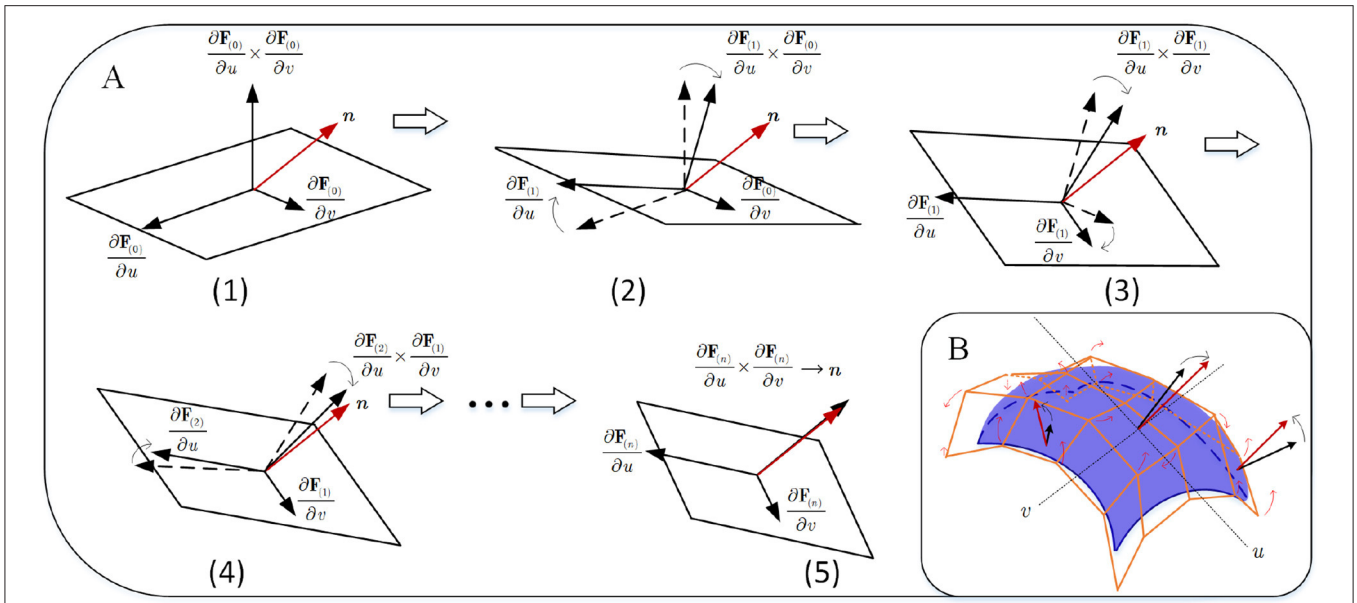


FIGURE 5 | Iterative adjustment on two partial derivatives: Process (1) to (2) adjusts $\frac{\partial \boldsymbol{F}}{\partial u}$ by fixing $\frac{\partial \boldsymbol{F}}{\partial v}$, and process (3) to (4) adjusts $\frac{\partial \boldsymbol{F}}{\partial v}$ by fixing $\frac{\partial \boldsymbol{F}}{\partial u}$, … until that $\frac{\partial \boldsymbol{F}}{\partial u} \times \frac{\partial \boldsymbol{F}}{\partial v}$ is infinitely close to objective $\boldsymbol{n}$; Process A implements a practically and iteratively linear handle for B-spline surface adjustment in B.

7.1. Data Pipeline and Evaluation

7.1.1. Synthesized Data With Expression

The ground truth data are from the space-times faces (Zhang et al., 2017) which contains 3D face models with different expressions. We use the data because it is convenient to evaluate our method with ground truth. And different poses and illuminations can also be simulated by the space-times faces, seen in **Figure 6**. Images with various poses and illuminations are collected, and feature points manually labeled. The reconstruction is evaluated by the error to the ground truth model.

7.1.2. Real Data in the Wild

The wild data (Huang et al., 2007) has characteristics of subject variations, illumination changes, various pose, background clutter and occlusions. Images of each person are collected and input into a facial point detector (Burgos-Artizzu et al., 2013) that has a similar high performance to human, to find the 40 facial points shown in **Figure 4**. The initial B-spline template face is computed from a neutral model of space-time faces.

7.1.3. Comparison

To verify the accuracy of automatic surface reconstruction, discrete points are sampled from the generated continuous free-form shape, and are compared to the traditional discrete reconstructions, e.g., work by Kemelmacher and Seitz (2011) and Roth et al. (2015). For a memory-limited capture system, it is not available to collect thousands of images as what Kemelmacher and Seitz (2011) and Roth et al. (2015) have done, so we limit all the reconstructions to less than forty images. We also compare them with an end-to-end deep learning method by Sela et al. (2017) qualitatively. Deep learning methods rely training on a large amount of unconstrained data, so we just use the model provided by Sela et al. (2017) that have been training on unconstrained images, and test it on the images in the wild.

7.2. Synthesized Standard Images

We conduct five sessions of reconstructions: the first four are used to reconstruct expression *S1*, *S2*, *S3*, and *S4* by using their corresponding images, and the fifth session *S5* is based on images with different expressions. Each session contains 40 images with various illumination and different poses. Reconstruction results are compared with the re-implemented method Kemel_meth by Kemelmacher and Seitz (2011) and Roth_meth by Roth et al. (2015). Kemel_meth generates frontal face surface based on integration in image domain of size 120 × 110. We clip it according to the peripheral facial points and interpolate points to get more vertices. Roth_meth generates a face mesh based on a template with 23,725 vertices. In our method, control point grid of 102 × 77 is optimized for a B-spline face surface.

7.2.1. Quantitative Comparison

To compare the approaches numerically, we compute the shortest point-to-point distance from ground truth to reconstruction. Point clouds are sampled from B-spline face and aligned according to absolute orientation problem. As done in work of Roth et al. (2015), mean Euclidean distance (MED), and the root mean square (RMS) of the distances, after normalized by the eye-to-eye distance, are reported in **Table 1**. Particularly, evaluation of Roth_meth is based on surface clipped with same facial points like the other two methods by considering a fair comparison. In the table, the best results are highlighted in boldface, and the underlined result has no significant difference with the best. To our knowledge, Roth_meth is the state-of-art

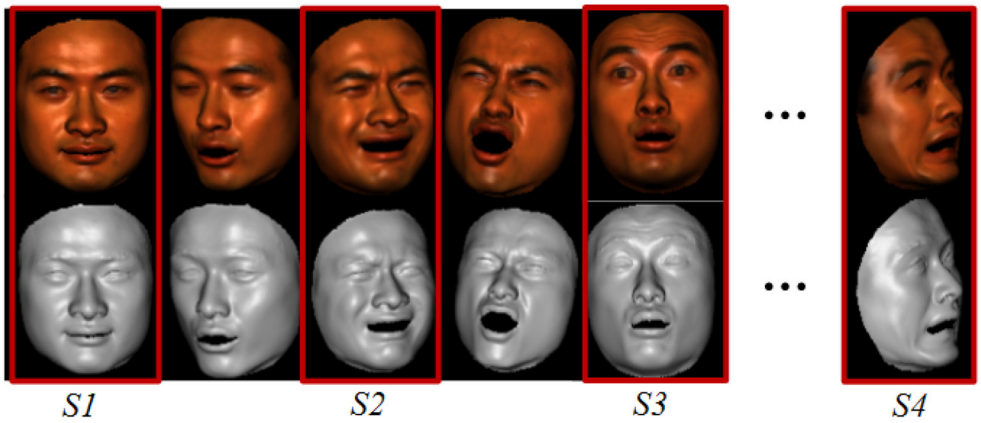


FIGURE 6 | Sample data simulated by the spaces-times faces (Zhang et al., 2017) : images and 3D model with various poses and illuminations are available; data of sample *S1*, *S2*, *S3*, and *S4* are used for evaluation.

TABLE 1 | Distances of the reconstruction to the ground truth.

Meth.	Index	S1	S2	S3	S4	S5
Kemel_meth	MED (%)	8.08	8.18	8.18	10.75	8.65
	RMS (%)	6.64	6.93	4.29	7.11	6.90
Roth_meth	MED (%)	**5.25**	7.06	5.43	6.63	**6.96**
	RMS (%)	**4.36**	5.79	4.54	4.42	**4.62**
Ours	MED (%)	6.31	**6.49**	**4.43**	**6.46**	6.98
	RMS (%)	4.10	**4.66**	**2.91**	**4.06**	4.34

The bold means the best value of MED and RMS, while the underline indicates the values next to the best.

method for face reconstruction from unconstrained images. Its re-implementation version is affected by the noisy normal estimation because of limited number images, showing results that are not good like as in its original paper. But it still performs good on all sessions. As a whole, results by both Roth_meth and our method have lower errors than Kemel_meth. On session S1 and S5, Roth_meth obtains the lowest mean error 5.21 and 6.96%, respectively. However, we obtains lower RMS 4.10 and 4.34% while its errors is quite close to the best especially on session S5. And on session S2, S3, and S4, our method obtains the best results, 6.49 $\pm$ 4.66, 4.43 $\pm$ 2.91, and 6.46 $\pm$ 4.06%. In contrast, the errors by Kemel_meth exceed 8%, and the RMS is also very large on every session. These numerical comparisons supply highly persuasive evidence that our B-spline method can build promising reconstructions based on face images.

7.2.2. Visual Comparison

The visual results in **Figure 7**. We show 3D models in mesh format for three methods on different sessions, and vertex numbers of models are also presented. It also demonstrates that our method has a promise performance by comparisons in the figure. An important fact is that Kemel (Kemelmacher and Seitz, 2011) cannot make a credible depth information and global shape, e.g., the global shape of reconstruction S2 and the mouse and nose of S3 are obviously incorrect, but our method solves global and local problem by optimization of 0th- and 1st-order consistency. And while Roth (Roth et al., 2015) generates more detailed information of an individual, it also produces distortion at the detailed shape, e.g., the eye of reconstruction S2 and the nose of reconstruction S3 and S4. In contrast, our method obtains realistic shape both globally and locally.

7.2.3. Characteristic Comparison

We give statistics of characteristics of the results generated by the three methods in **Table 2**, covering the global shape, local detail, credible depth, smoothness, distortion, and derivability. Depending on the quantitative and qualitative comparisons, we also give a rough rating. One star, two stars, and three stars represents bad, general, and good reconstruction respectively in the rating system. Both Roth_meth and our method obtain good scores on global shape, local detail, and credible depth. And both Kemel_meth and our method obtain a good score on smoothness. Because of the bad depth, Kemel_meth also gets bad score on global shape and distortion, and gets general scores on local detail. In addition, B-spline face model has better smoothness than the models by Kemel_meth and Roth_meth, because it is C^2 differentiable parametric surface while the other two are discrete model. Conclusively, 0th- and 1st-order consistency modeling using B-spline surface is efficient to reconstruct parametric surface of individual face.

7.3. Real Unconstrained Images

Our method is also tested based on real unconstrained data. Unconstrained data mean that the images are captured under uncertain condition, and the faces in the images are different in expression, pose and illumination condition. It is difficult to build the geometrical consistency for reconstruction using such data. Unlike the experiments in the work by Kemelmacher and Seitz (2011) using hundreds of images, we conduct reconstruction with limited number of images, because a large mount of face images for one person are not always available for small sample size tasks such as criminal investigation. In the experiment, uniformly 35 images are collected for each person from LFW database[1] covering different poses, illuminations and expressions.

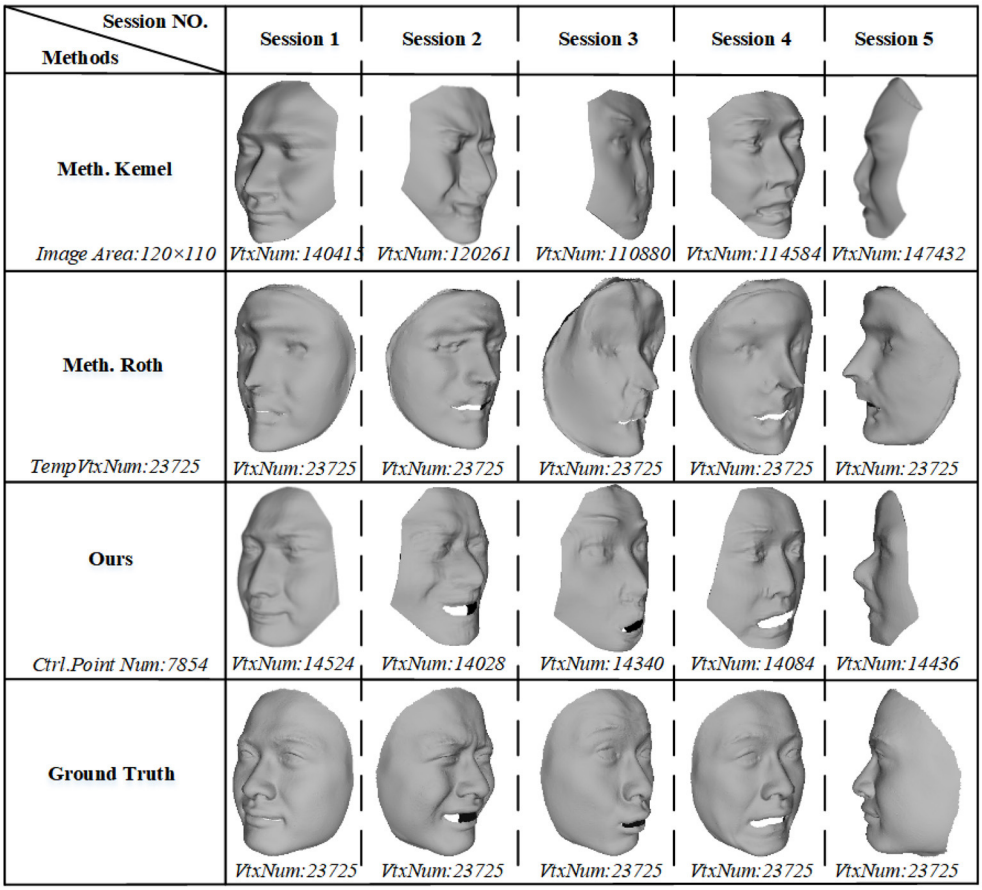


FIGURE 7 | Visual reconstructions and comparisons for session *S1*, *S2*, *S3*, *S4*, and *S5*: for each session of reconstructions, a column lists the 3D results of Kemelmacher and Seitz (2011), Roth et al. (2015), and us, as well as ground truth. (VrxNum means vertex number; TempVtxNum means vertex number of template; and Ctrl.Point Num means the control point number of B-spline face surface. Particularly, the vertices of B-spline face are points sampled from the reconstructed parametric surface).

TABLE 2 | A characteristics summarization of three methods by rough rating with number of ☆.

Characteristics	Kemel_meth	Roth_meth	Ours
Global shape	☆×1	☆×3	☆×3
Local detail	☆×2	☆×3	☆×3
Credible depth	☆×1	☆×3	☆×3
Smoothness	☆×3	☆×2	☆×3
No distortion	☆×1	☆×2	☆×3
C^2 differentiable	NO	NO	YES

Visual face reconstructions for Colin Powell, Donald Rumsfeld, George W. Bush, Hugo Chavez, and Gloria Macapagal Arroyo are compared with other two methods, as shown in **Figure 8**. Let *A* label the results generated by the reimplemented Kemel_meth, and let *B* label the results generated by the reimplemented Roth_meth, and let *C* label the method Seta_meth of deep learning by Sela et al. (2017) and let *D* label our results. Particularly, the input for Seta_meth is one image selected from the 35 images. Images in column 1, 5, and 8 are corresponding mean textures and two views of images respectively. By comparing these results, we observe some phenomena as follows:

(1) In frontal viewpoint, *A* and *D* show more vivid details than *B*, e.g., eyes and nose of Colin Powell. But in an other viewpoint, *D* shows more credible shape than *A*, e.g., the eyes and the forehead of Colin Powell, and the forehead and the mouth of Donald Rumsfeld.
(2) When the normals are incorrectly estimated from a limited number of images, e.g., for Gloria Macapagal Arroyo, *A* loses the local information completely, but *B*, *C*, and *D* still maintain general geometrical shape of face. For all methods, reconstructing nose is a challenge because the geometric curvature of the nose varies greatly. When the images are not enough, the noise could be amplified. So *B* shows bad results at nose being limited by number of input images.
(3) The input of *C* is a approximately frontal face image selected. As the model of *C* is learning on a set of 3D face data, it may not handle the uncertain noise and identity of inputs. So the

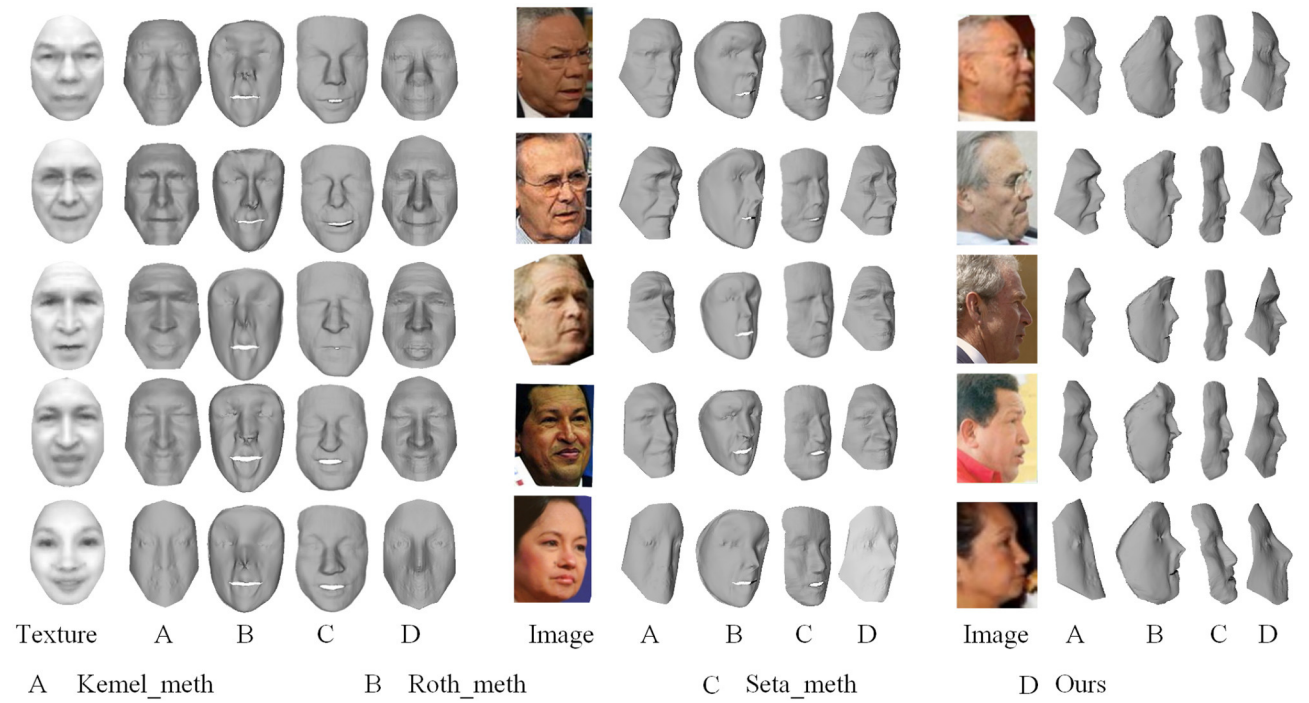


FIGURE 8 | Visual reconstructions and comparisons for Colin Powell, Donald Rumsfeld, George W. Bush, Hugo Chavez, and Gloria Macapagal Arroyo: (1) Images in column 1, 5, and 9 are corresponding mean textures and two views of images, respectively; (2) Columns labeled by *A* show the results generated by the reimplemented Kemel_meth, and columns labeled by *B* show the results generated by the reimplemented method Roth_meth, and columns labeled by *C* show the results generated by the Seta_meth, and columns labeled by *C* show our results. (The face images used in the figure come from LFW database[1]).

details in reconstruction by *C* don't look real, although their global shapes are stable and like human faces.

(4) By comparison, our method steadily produces better looking results than others from different viewpoints in the dataset. Clear and vivid details can be seen at key components such as eyes, nose and mouth, forehead, and cheek.

8. DISCUSSION

All the above experiments prove that our method can build pinpoint geometrical consistency on the limited number of real unconstrained data. Our method may not be best method in area of 3D reconstruction from multiple images, as the results in the original work by *B* looks better. It could deal with 3D reconstruction with limited number of images. Because we may not obtain large amount of images for reconstruction as done by Roth et al. (2015), for some condition restricted system. The shortcomings of *A* are mainly resulted from the inauthentic depth generated by integration method. And the bad results of *B* are caused by that the mesh template cannot build correct geometric consistency of number limited of unconstrained images and that the discrete differential operating on estimated noisy normal brings distortion errors. In contrast, we build pinpoint geometric consistency using B-spline surface. B-spline can smooth the noise in estimated normal better. So *D* can reconstruct correct face shape with little distortion, showing better result as a whole.

In the comparison, we don't consider other deep learning methods based methods appeared in recent years (Dou et al., 2017; Richardson et al., 2017; Lin et al., 2020; Sengupta et al., 2020; Shang et al., 2020). Because almost all recent works are focused on deep learning methods for single image based 3D face reconstruction (Dou et al., 2017; Richardson et al., 2017; Lin et al., 2020; Sengupta et al., 2020), as well as using a 3DMM model as prior. And the multi-view deep learning method only handle constrained face images (Shang et al., 2020). It means the deep learning methods can use a large amount of training data, and also a good prior. The input are different between these learning based methods and our method. So we conduct comparison with the classic optimization-based approaches for the sake of fairness. Nevertheless, we also select one representative method by Sela et al. (2017) to show result by deep learning as a reference in the comparison. It proves that if the test are not satisfactory to the prior and distribution of training data, it may obtain bad result.

9. CONCLUSIONS

This study set out to present high-detailed face reconstruction from multiple images based on pinpoint 0th- and 1st-order

geometric consistence using B-spline embedding. Based on the good consistence modeling in geometric optics, the method works well for data with different poses and expressions in the wild. The key contribution of this study is that surface modeling adapts the correct rays in geometric optics by using B-spline embedding. This makes the high-detailed B-spline modeling from a number limited of face images captured under wild condition become reality. The method could also be applied to expression tracking and assisting face recognition in a monitoring or robot system.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author/s.

AUTHOR CONTRIBUTIONS

WP and ZF has contributed equally to the core idea as well as the experiment design and results analysis. YS, KT, and CX has provided assistance in experiments and analysis, under ZF's supervision. Besides, KT and MF provided the research group with financial support and experimental equipments. KT and ZF are supportive corresponding authors. All authors contributed to the article and approved the submitted version.

FUNDING

This research was partly supported by Science and Technology Program of Guangzhou, China (No. 202002030263), Shenzhen Science and Technology Foundation (JCYJ20170816093943197), Guangdong Basic and Applied Basic Research Foundation (2020A1515110997), National Natural Science Foundation of China (Nos. 61772164 and 62072126), and National Key R&D Program of China (No. 2019YFB1706003).

REFERENCES

Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S. M., et al. (2011). Building Rome in a day. *Commun. ACM* 54, 105–112. doi: 10.1145/2001269.2001293

Artificial, L. A., and Aryananda, L. (2002). "Recognizing and remembering individuals: Online and unsupervised face recognition for humanoid robot," in *Proc. of IROS* (Lausanne), 1202–1207.

Barsky, S., and Petrou, M. (2003). The 4-source photometric stereo technique for three-dimensional surfaces in the presence of highlights and shadows. *IEEE Trans. Pattern Anal. Mach. Intell.* 25, 1239–1252. doi: 10.1109/TPAMI.2003.1233898

Bhardwaj, A., and Raman, S. (2016). Robust PCA-based solution to image composition using augmented lagrange multiplier (ALM). *Visual Comput.* 32, 591–600. doi: 10.1007/s00371-015-1075-1

Blanz, V., Mehl, A., Vetter, T., and Seidel, H. P. (2004). "A statistical method for robust 3D surface reconstruction from sparse data," in *International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)* (Thessaloniki), 293–300. doi: 10.1109/TDPVT.2004.1335212

Blanz, V., and Vetter, T. (1999). "A morphable model for the synthesis of 3D faces," in *Proceedings of Conference on Computer Graphics and Interactive Techniques* (New York, NY), 187–194. doi: 10.1145/311535.311556

Burgos-Artizzu, X. P., Perona, P., and Dollár, P. (2013). "Robust face landmark estimation under occlusion," in *IEEE International Conference on Computer Vision (ICCV)*, (Sydney, VIC), 1513–1520. doi: 10.1109/ICCV.2013.191

Carr, J. C., Beatson, R. K., Cherrie, J. B., Mitchell, T. J., Fright, W. R., Mccallum, B. C., et al. (2001). Reconstruction and representation of 3D objects with radial basis functions. *ACM Siggraph* 67–76. doi: 10.1145/383259.383266

Deng, Y., Yang, J., Xu, S., Chen, D., Jia, Y., and Tong, X. (2019). "Accurate 3D face reconstruction with weakly-supervised learning: From single image to image set," in *IEEE Computer Vision and Pattern Recognition Workshops*. Long Beach, CA. doi: 10.1109/CVPRW.2019.00038

Dou, P., Shah, S. K., and Kakadiaris, I. A. (2017). "End-to-end 3D face reconstruction with deep neural networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Honolulu, HI), 1503–1512. doi: 10.1109/CVPR.2017.164

Gecer, B., Ploumpis, S., Kotsia, I., and Zafeiriou, S. (2019). "Ganfit: generative adversarial network fitting for high fidelity 3D face reconstruction," in *IEEE Computer Vision and Pattern Recognition (CVPR)* (Long Beach, CA), 1155–C1164. doi: 10.1109/CVPR.2019.00125

Gonzalez-Mora, J., De la Torre, F., Guil, N., and Zapata, E. L. (2010). Learning a generic 3D face model from 2D image databases using incremental structure-from-motion. *Image Vis. Comput.* 28, 1117–1129. doi: 10.1016/j.imavis.2010.01.005

Heo, J., and Savvides, M. (2009). "In between 3D active appearance models and 3D morphable models," in *Computer Vision and Pattern Recognition* (Miami Beach, FL). doi: 10.1109/CVPRW.2009.5204300

Hoch, M., Fleischmann, G., and Girod, B. (1998). Modeling and animation of facial expressions based on b-splines. *Vis. Comput.* 11, 87–95. doi: 10.1007/BF01889979

Huang, G. B., Ramesh, M., Berg, T., and Learned-Miller, E. (2007). *Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments*. Technical Report 07-49, University of Massachusetts, Amherst.

Kazhdan, M., Bolitho, M., and Hoppe, H. (2006). "Poisson surface reconstruction," in *Proceedings Symposium on Geometry Processing (SGP) 06* (Goslar), 32.

Kemelmacher Shlizerman, I. and Basri, R. (2011). 3d face reconstruction from a single image using a single reference face shape. *IEEE Trans. Pattern Anal. Mach. Intell.* 33, 394–405. doi: 10.1109/TPAMI.2010.63

Kemelmacher Shlizerman, I. and Seitz, S. M. (2011). "Face reconstruction in the wild," in *IEEE International Conference on Computer Vision (ICCV)* (Barcelona), 1746–1753. doi: 10.1109/ICCV.2011.6126439

Khraisat, A., Gondal, I., Vamplew, P., Kamruzzaman, J., and Alazab, A. (2019). A novel ensemble of hybrid intrusion detection system for detecting internet of things attacks. *Electronics* 8:1210. doi: 10.3390/electronics8111210

Koo, H.-S., and Lam, K.-M. (2008). Recovering the 3D shape and poses of face images based on the similarity transform. *Pattern Recogn. Lett.* 29, 712–723. doi: 10.1016/j.patrec.2007.11.018

Li, M., Sun, Y., Lu, H., Maharjan, S., and Tian, Z. (2019). Deep reinforcement learning for partially observable data poisoning attack in crowdsensing systems. *IEEE Intern. Things J.* 7, 6266–6278. doi: 10.1109/JIOT.2019.2962914

Lin, J., Yuan, Y., Shao, T., and Zhou, K. (2020). "Towards high-fidelity 3D face reconstruction from in-the-wild images using graph convolutional networks," in *2020 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5891–5900. doi: 10.1109/CVPR42600.2020.00593

Lu, Y., Yong, J. H., Shi, K. L., Song, H. C., and Ye, T. Y. (2016). 3D b-spline curve construction from orthogonal views with self-overlapping projection segments. *Comput. Graph.* 54, 18–27. doi: 10.1016/j.cag.2015.07.010

Maejima, A., Kuratate, T., Pierce, B., Morishima, S., and Cheng, G. (2012). "Automatic face replacement for humanoid robot with 3D face shaped display," in *2012 12th IEEE-RAS International Conference on Humanoid Robots* (Osaka), 469–474. doi: 10.1109/HUMANOIDS.2012.6651561

Meng, M., Lan, M., Yu, J., Wu, J., and Tao, D. (2020). Constrained discriminative projection learning for image classification. *IEEE Trans. Image Process.* 29, 186–198. doi: 10.1109/TIP.2019.2926774

Meyer, M., Desbrun, M., Schroder, P., and Barr, A. H. (2003). *Discrete Differential-Geometry Operators for Triangulated 2-Manifolds*. Berlin; Heidelberg: Springer. doi: 10.1007/978-3-662-05105-4_2

Mian, A., Bennamoun, M., and Owens, R. (2006). "Automatic 3D face detection, normalization and recognition," in *Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'06)* (Chapel Hill, NC) 735–742. doi: 10.1109/3DPVT.2006.32

Peng, W., Feng, Z., Xu, C., and Su, Y. (2017). "Parametric t-spline face morphable model for detailed fitting in shape subspace," in *IEEE Computer Vision and Pattern Recognition (CVPR)* (Honolulu, HI), 5515–5523. doi: 10.1109/CVPR.2017.585

Peng, W., Xu, C., and Feng, Z. (2016). 3D face modeling based on structure optimization and surface reconstruction with b-spline. *Neurocomputing* 179, 228–237. doi: 10.1016/j.neucom.2015.11.090

Piegl, L., and Tiller, W. (1997). *The Nurbs Book*. Monographs in Visual Communication. doi: 10.1007/978-3-642-59223-2

Piotraschke, M., and Blanz, V. (2016). "Automated 3D face reconstruction from multiple images using quality measures," in *Proc. IEEE Computer Vision and Pattern Recognition* (Las Vegas, NV), doi: 10.1109/CVPR.2016.372

Prados, E., and Faugeras, O. (2005). "Shape from shading: a well-posed problem?," in *IEEE Computer Vision and Pattern Recognition (CVPR)*, *Vol. 2*, (San Diego, CA), 870–877.

Qiu, J., Tian, Z., Du, C., Zuo, Q., Su, S., and Fang, B. (2020). A survey on access control in the age of internet of things. *IEEE Intern. Things J.* 7, 4682–4696. doi: 10.1109/JIOT.2020.2969326

Richardson, E., Sela, M., and Kimmel, R. (2016). "3D face reconstruction by learning from synthetic data," in *International Conference on 3D Vision (3DV)* (Stanford, CA), 460–469. doi: 10.1109/3DV.2016.56

Richardson, E., Sela, M., Or-El, R., and Kimmel, R. (2017). "Learning detailed face reconstruction from a single image," in *IEEE Computer Vision and Pattern Recognition (CVPR)* (Honolulu, HI), 5553–5562. doi: 10.1109/CVPR.2017.589

Roth, J., Tong, Y., and Liu, X. (2015). "Unconstrained 3D face reconstruction," in *IEEE Computer Vision and Pattern Recognition (CVPR)* (Boston, MA). doi: 10.1109/CVPR.2015.7298876

Roth, J., Tong, Y., and Liu, X. (2016). "Adaptive 3D face reconstruction from unconstrained photo collections," in *Proc. IEEE Computer Vision and Pattern Recognition*. doi: 10.1109/CVPR.2016.455

Seitz, S. M., Curless, B., Diebel, J., Scharstein, D., and Szeliski, R. (2006). "A comparison and evaluation of multi-view stereo reconstruction algorithms," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)* (New York, NY), 519–528.

Sela, M., Richardson, E., and Kimmel, R. (2017). "Unrestricted facial geometry reconstruction using image-to-image translation," in *2017 IEEE International Conference on Computer Vision (ICCV)* (Venice). doi: 10.1109/ICCV.2017.175

Sengupta, S., Lichy, D., Kanazawa, A., Castillo, C. D., and Jacobs, D. W. (2020). SfSNet: Learning shape, reflectance and illuminance of faces in the wild. *IEEE Trans. Pattern Anal. Mach. Intell.* doi: 10.1109/TPAMI.2020.3046915

Shang, J., Shen, T., Li, S., Zhou, L., Zhen, M., Fang, T., et al. (2020). "Self-supervised monocular 3D face reconstruction by occlusionaware multi-view geometry consistency," in *Proceedings of the European Conference on Computer Vision (ECCV)* (Glasgow). doi: 10.1007/978-3-030-58555-6_4

Sun, Z.-L., Lam, K.-M., and Gao, Q.-W. (2013). Depth estimation of face images using the nonlinear least-squares model. *IEEE Trans. Image Process.* 22, 17–30. doi: 10.1109/TIP.2012.2204269

Tomasi, C., and Kanade, T. (1992). Shape and motion from image streams under orthography: a factorization method. *Int. J. Comput. Vis.* 9, 137–154. doi: 10.1007/BF00129684

Tran, A. T., Hassner, T., Masi, I., and Medioni, G. (2017). "Regressing robust and discriminative 3D morphable models with a very deep neural network," in *IEEE Computer Vision and Pattern Recognition (CVPR)*, (Honolulu, HI). doi: 10.1109/CVPR.2017.163

Wang, H., Wei, H., and Wang, Y. (2003). "Face representation under different illumination conditions," in *International Conference on Multimedia and Expo (ICME)* (Baltimore, MD), 285–288.

Wu, F., Bao, L., Chen, Y., Ling, Y., Song, Y., Li, S., et al. (2019). "MVF-net: multi-view 3D face morphable model regression," in *IEEE Computer Vision and Pattern Recognition (CVPR)* (Long Beach, CA), 959–968. doi: 10.1109/CVPR.2019.00105

Yang, C., Chen, J., Su, N., and Su, G. (2014). "Improving 3D face details based on normal map of hetero-source images," in *IEEE Computer Vision and Pattern Recognition Workshops (CVPRW)* (Columbus, OH), 9–14. doi: 10.1109/CVPRW.2014.7

Yuille, A. L., Snow, D., Epstein, R., and Belhumeur, P. N. (1999). Determining generative models of objects under varying illumination: shape and albedo from multiple images using SVD and integrability. *Int. J. Comput. Vis.* 35, 203–222. doi: 10.1023/A:1008180726317

Zhang, L., Mistry, K., Jiang, M., Chin Neoh, S., and Hossain, M. A. (2015). Adaptive facial point detection and emotion recognition for a humanoid robot. *Comput. Vis. Image Understand.* 140, 93–114. doi: 10.1016/j.cviu.2015.07.007

Zhang, L., Snavely, N., Curless, B., and Seitz, S. M. (2017). "Spacetime faces: high-resolution capture for modeling and animation," in *Data-Driven 3D Facial Animation* eds Deng, Z., and Neumann, U. (Los Angeles, CA: Springer), 248–276. doi: 10.1007/978-1-84628-907-1_13

Zhang, R., Tsai, P.-S., Cryer, J. E., and Shah, M. (1999). Shape-from-shading: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.* 21, 690–706. doi: 10.1109/34.784284

Zhou, Y., Deng, J., Kotsia, I., and Zafeiriou, S. (2019). "Dense 3D face decoding over 2500 fps: joint texture shape convolutional mesh decoders," in *IEEE Computer Vision and Pattern Recognition (CVPR)* (Long Beach, CA), 1097–1106. doi: 10.1109/CVPR.2019.00119

Zhu, X., Lei, Z., Liu, X., Shi, H., and Li, S. Z. (2016). "Face alignment across large poses: a 3D solution," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (Las Vegas, NV), 146–155. doi: 10.1109/CVPR.2016.23

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

frontiers in Neurorobotics

ORIGINAL RESEARCH
published: 11 May 2021
doi: 10.3389/fnbot.2021.674428

A Novel Encoder-Decoder Knowledge Graph Completion Model for Robot Brain

Yichen Song, Aiping Li, Hongkui Tu, Kai Chen and Chenchen Li*

College of Computer, National University of Defence Technology, Changsha, China

With the rapid development of artificial intelligence, Cybernetics, and other High-tech subject technology, robots have been made and used in increasing fields. And studies on robots have attracted growing research interests from different communities. The knowledge graph can act as the brain of a robot and provide intelligence, to support the interaction between the robot and the human beings. Although the large-scale knowledge graphs contain a large amount of information, they are still incomplete compared with real-world knowledge. Most existing methods for knowledge graph completion focus on entity representation learning. However, the importance of relation representation learning is ignored, as well as the cross-interaction between entities and relations. In this paper, we propose an encoder-decoder model which embeds the interaction between entities and relations, and adds a gate mechanism to control the attention mechanism. Experimental results show that our method achieves better link prediction performance than state-of-the-art embedding models on two benchmark datasets, WN18RR and FB15k-237.

Keywords: learning-based artificial intelligence, robot intelligence, human-robot interaction, knowledge graph reasoning and completion, knowledge graph embedding

OPEN ACCESS

Edited by:
Zhaoquan Gu,
Guangzhou University, China

Reviewed by:
Xiang Lin,
Shanghai Jiao Tong University, China
Li Xiaoyong,
Beijing University of Posts and Telecommunications (BUPT), China
Jinqiao Shi,
Beijing University of Posts and Telecommunications (BUPT), China

****Correspondence:***
Aiping Li
13017395458@163.com

Received: *01 March 2021*
Accepted: *12 April 2021*
Published: *11 May 2021*

Citation:
Song Y, Li A, Tu H, Chen K and Li C (2021) A Novel Encoder-Decoder Knowledge Graph Completion Model for Robot Brain. Front. Neurorobot. 15:674428. doi: 10.3389/fnbot.2021.674428

1. INTRODUCTION

With the development of science and technology, significant progress has been achieved in robotics that the types and application fields of robots are constantly enriched. These robots have played key roles in reducing tedious work. They provide optimal user service and improve the convenience of life. The popularity of various kinds of robots is an inevitable trend.

The emergence of learning intelligent social robots means that robots have truly begun to play roles in people's daily lives, such as pepper and buddy. There are some typical applications, such as greeting conversations, question responses, interest recommendations, and risk management (Gu et al., 2021). Huge information is need at the backend of these services. However, the traditional search engine will be affected by the combination of information, resulting in an increase in search volume and a decrease in accuracy. Knowledge with unique meaning and with the goal of solving practical problems can avoid this problem well.

The "Robot Brian" is taken the same as the brain for humans, which stores and infers knowledge to support other behaviors. Knowledge base is usually used to work as the brain of an intelligent robot. Unlike general applications that implicitly encode information in programs, it can explicitly express the corresponding knowledge of actual problems. Providing continuous knowledge support for robots through the knowledge base is equivalent to injecting "thought" into the robots to realize real intelligence true intelligence. Knowledge base construction is a core configuration for

intelligent robots. Without a knowledge base, a robot cannot answer any questions. The richer the knowledge base, the more intelligent the robot will have when interacting with users.

A large amount of research work in knowledge representing, web data mining, natural language processing and other fields are dedicated to acquiring large-scale knowledge (Jia et al., 2021), providing rich knowledge bases for building the intelligent brain of robots. In order to facilitate computer processing and understanding, we express the knowledge base in a more formal and concise way, that is, a highly structured knowledge graph composed of triples (e_h, r_k, e_t). The Knowledge Graphs (KG) not only provides robots with a more human-like representation of the world, but also provides a better way to organize, manage and utilize massive amounts of information.

Although the large-scale knowledge graphs already contain a large amount of entity and relation information, they are still incomplete compared with existing knowledge and newly added knowledge (Zhao et al., 2020). Through knowledge graphs and knowledge self-learning, problems in the knowledge system can be found, and knowledge can be supplemented and enhanced so that the robot's knowledge base can be continuously improved and evolved. There is no end to the optimization and completion of the knowledge base, just as there is no end to human learning, this is research work that needs continuous improvement and development.

In order to alleviate the above problems, researchers have proposed a knowledge graph embedding method, which predicts missing links based on existing facts so as to expand the knowledge base. Its purpose is to learn low-dimensional vector representations of all entities and relationships, so as to simplify operations while the original structured information of the knowledge graph is retained. These knowledge graph embedding methods are widely divided into translation models (Bordes et al., 2013; Ji et al., 2015; Lin et al., 2015), semantic matching model (Nickel et al., 2011, 2016; Yang et al., 2015; Trouillon et al., 2016), and neural network models (Dettmers et al., 2018; Shang et al., 2019; Vashishth et al., 2020). The related work will be introduced in detail in section 2.

Compared with neural network models, the other types of models are all shallow models, which leads to problems with poor expressiveness. Therefore, more and more complex and deeper models, which have better expressive performance and have achieved competitive success in modeling knowledge graphs, have been proposed in recent years. But these existing models, such as Dettmers et al. (2018), Nguyen et al. (2018), Shang et al. (2019), and Vashishth et al. (2020), are more focused on entity representation learning, and the importance of relation representation learning are ignored, let alone the cross-interaction between entities and relations. The interaction between entities and relations plays an important role in knowledge graph representation learning that the entities and relations in the knowledge graph will influence each other and influence the prediction of new triples as they do in the real world.

In this paper, a method of knowledge reasoning and completion based on neural networks on the knowledge graph is designed for robots to simulate the reaction and learning process of human brains. Our model adopts the encoder-decoder model. The encoder model improved the KBGAT model with a gate mechanism to control the attention mechanism and use entity embeddings to update relation embeddings. The decoder model uses Conv-TransE and Conv-TransR to achieve state-of-the-art efforts. This method can enable the robot to quickly search for information, predict answers, and complete knowledge from the knowledge base, to better understand user intent and interact with users more intelligently.

2. RELATED WORK

In this section, we mainly introduce the work related to our Large-scale Knowledge Graph reasoning and completion methods for robots. As one of the research hotspots, Large-scale Knowledge Graph reasoning and completion has attracted extensive attention from academia and industry. Thus, many different types of methods are born, such as the translation model, the bilinear model, the hyperbolic geometry model, the neural network model, the rotate model, and so on. Among these different kinds of methods, the knowledge graph Embedding method is the closest to human expression, which can be regarded as languages for computers and machines like robots. The knowledge graph Embedding method generally includes the following types of models: (i) translation models; (ii) models based on semantic matching; (iii) models based on neural networks; (iv) models with additional information. We will mainly introduce the work related to translation models and neural network based models related to our work in the following.

The translation model represented by TransE (Bordes et al., 2013) uses a simple vector form to represent the entities and relations in the knowledge graph. TransE (Bordes et al., 2013) regards relation as the conversion from the head entity e_h to the tail entity e_t, and uses $e_h + r_k = e_t$ to determine whether the given triplet is correct. In order to make up for the defect that TransE can only handle the 1–1 relation, TransH (Wang et al., 2014), TransD (Ji et al., 2015), TransR (Lin et al., 2015), and other models have increased the ability to handle multiple relations and semantics and enhanced the knowledge embedding model. It shows that entities and relations can also be embedded in other spaces besides real number space. TransG (Xiao et al., 2015) introduces Gaussian distribution to solve the problem of multi-relational semantics to capture the uncertainty of entities and relations. TorusE (Ebisu and Ichise, 2018) is the first model to embed objects outside the space of real or complex numbers and select a torus (compact Lie group) as the embedding space.

Neural network-based embedding models have received extensive attention in recent years. These methods include embedding models based on convolutional neural networks (CNN) and graph convolution networks (GCN). For example, convE (Dettmers et al., 2018), ConvKB (Nguyen et al., 2018), and InteractE (Vashishth et al., 2020) are both relational prediction models based on convolutional neural networks. ConvE (Dettmers et al., 2018) stacks the embeddings of head entity and relation into a 2-dimensional matrix, and performs convolution operation to extract features with fewer parameters and faster calculations. IntercatE (Vashishth et al., 2020) increases the expressive power of ConvE and expands the interaction between entities and relations through three

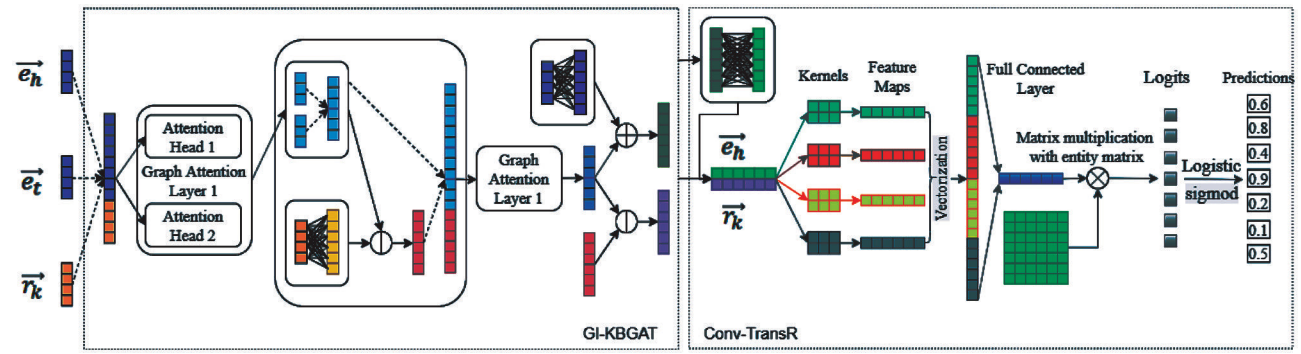


FIGURE 1 | Model Architecture. The encoder in the left uses is our GI-KBGAT to obtain the embeddings of entity and relation. The right decoder model feeds the embeddings of e_h and r_k and use logits activation function to calculate scores. Dashed arrows in the figure represent concatenation operation. The dimension of vector $\vec{e_h}$ and $\vec{r_k}$ in the middle is the same with the left in fact. We here use 4 dimension to simply the representation of the model.

key ideas—feature permutation, feature reshaping, and circular convolution. ConvKB (Nguyen et al., 2018) represents each triple as a 3-column matrix where each column vector represents a triple element and feeds this matrix to a 1D convolution layer to generalize transitional characteristics in transition-based embedding models.

Graph convolution networks have made great progress in improving the efficiency of node representation in the graph, and it is also applied in the knowledge graph by researchers. Graph convolution network (GCN) (Kipf and Welling, 2017) gathers information for node(entity) from its neighbors with equal importance. Velickovic et al. (2018) introduce a graph attention network (GAT) to learn to assign varying levels of importance to node(entity) in every neighbor. However, these models are unsuitable for KGs, since they ignore that edges (triples) play different roles depending on the relation they are associated with in KGs. SACN (Shang et al., 2019) extends the classic GCN to a weighted graph convolutional network (WGCN) as an encoder, and uses a convolution model Conv-TransE as a decoder to construct an end-to-end model. WGCN weighs the different types of relations differently when aggregating multiple single-relation graphs into a multi-relation graph and the weights are adaptively learned during the training of the network. But WGCN inherits GCN's shortcomings in that it treats the same relation type for different entities of the same weight. Nathani et al. (2019) extends classic GAT to KBGAT by incorporating relation and neighboring node features in the attention mechanism and uses KBGAT as encoder and ConvKB (Nguyen et al., 2018) as a decoder.

The above-mentioned models have achieved good performance in knowledge graph embedding for knowledge graph reasoning and completion. However, as far as we know, few works consider the cross-interaction of entities and relations when designing models. Our proposed model uses a variant of the graph attention network (GAT) as the encoder and uses variants of ConvE [Conv-TransE (Shang et al., 2019), Conv-TransR] as decoder, to achieve the simultaneous capture of entity-to-relation and relation-to-entity cross-interaction.

3. MODEL

This section begins by introducing some notations and definitions used in the rest of this article. This is followed by an introduction of our encoder model GI-KBGAT, an improved Graph Attention Network for KG, which considers gate mechanism on multi-head attention and interaction between entities and relations to generate embeddings. Finally, we describe our decoder network based on Conv-TransE (Conv-TransR). The architecture of our model is as shown in **Figure 1**.

3.1. Notations and Definitions

The knowledge graph is defined as $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T})$, where $\mathcal{E} = \{e_1, e_2, ...e_N\}$ and $\mathcal{R} = \{r_1, r_2, ...r_K\}$ represent the set of entities (nodes) and relations. N is number of entities and K is number of relations. $\mathcal{T}$ denotes the triples (edges) of the form $t_{htk} = (e_h, r_k, e_t) \in \mathcal{E} \times \mathcal{R} \times \mathcal{E}$, where e_h is head entity, e_t is tail entity, and r_k is the relation between head and tail entity. In particular, entity e_h and e_t in this paper refer to the head entity and the tail entity, respectively, while other entities with subscripts, such as entity e_i are not specified. **Table 1** explains the notations that will be used in the rest of this article.

3.2. Encoder: GI-KBGAT

As shown in section 2, most existing models ignore the cross-interaction of entities and relations. They only use relations to updating entities, but ignore the effects of entities on relations. We improve the KBGAT (Nathani et al., 2019) by modifying the update process of embeddings of entities and relations to consider the interaction between entities and relations in the update process, and adding a gate mechanism to the attention mechanism for control.

Our model uses the initial embeddings of entities and relations as input, and the following layers use the embeddings obtained from its previous layer as input. The same as Nathani et al. (2019)'s GAT model, in order to learn the embeddings of entity e_i, we aggregate features of triples associated with it. The triple's embedding is learned by performing a linear transformation over the concatenation of entity and relation vectors corresponding to

TABLE 1 | Some of the notations and explanations used in this paper.

Notation	Explanation	Notation	Explanation
$\mathbf{E}, \mathbf{R}$	Embedding matrices for entities and relations	$[\|\|]$	Concatenation of vectors
$\vec{e}_i, \vec{r}_k$	Embedding vectors of entity e_i and relation r_k	$\mid \mathcal{S} \mid$	Number of elements in set $\mathcal{S}$
$\mathcal{N}_i$	Neighbor sets of entity e_i	ϕ	LeakyReLU function
$\mathcal{R}_{ij}$	Relation sets connecting entities e_i and e_j	σ	Activate function
$\mathcal{P}_k$	Entity pair (e_h, e_t) sets with relation r_k	ψ	ReLU function

it, as shown in Equation (1).

$$\vec{t_{htk}} = \mathbf{W}_t[\vec{e_h} \parallel \vec{e_t} \parallel \vec{r_k}] \tag{1}$$

Where $\vec{t_{htk}}, \vec{e_h}, \vec{e_t}$, and $\vec{r_k}$ represent the embeddings of triple t_{htk}, entity e_h, e_t, and relation r_k, respectively, $\mathbf{W}_t$ denotes the linear transformation matrix. To measure the importance of each triple t_{htk} for entity e_h, the LeakyRelu non-linearity activation function ϕ is used to get the absolute attention value, the activate vector is defined as $\vec{b}$, and the softmax function is used to obtain the relative attention values α_{htk}, as shown in Equation (2).

$$\alpha_{htk} = \frac{exp(\phi(\vec{b}\vec{t_{htk}}))}{\sum_{n\in N_h}\sum_{r\in\mathcal{R}_{hn}} exp(\phi(\vec{b}\vec{t_{hnr}}))} \tag{2}$$

Then, the new embedding of entity e_h is obtained by aggregating the features of the triples associated with e_h through weighted by their attention values. As shown in Equation (3), attention values are used to calculate the linear combination of triples(neighbor) features, and the embedding is obtained with a activate function σ.

$$\vec{e_h} = \sigma(\sum_{t\in N_h}\sum_{k\in\mathcal{R}_{ht}} \alpha_{htk}\vec{t_{htk}}) \tag{3}$$

To stabilize the learning process and encapsulate more information, our encoder also uses a gated multi-head attention mechanism inspired by Vaswani et al. (2017), Velickovic et al. (2018), and Zhang et al. (2018). Considering M independent attention heads, M embeddings for an entity are obtained. For example, the embedding of entity e_h calculated by the $m-th$ attention head is represented as $\vec{e_h}^m$. These embeddings of an entity are concatenated with independent gate value g_h^m except the last layer (for which we use the mean pooling). The final entity embedding update equation is as follows:

$$\vec{e_h} = \sigma([\|_{m=1}^{M}(g_h^m \cdot \sigma(\sum_{t\in N_h}\sum_{k\in\mathcal{R}_{ht}} \alpha_{htk}\vec{t_{htk}})^m)]) \tag{4}$$

For the relation update, we propose an update mechanism that uses the same projection operation as the TransR model (Lin et al., 2015). Similar to GAT, TransR model holds that an entity is a complex of various attributes, and different relations focus on different attributes of the entity. TransR uses the projection matrix $\mathbf{M_r}$ to project the head entity e_h and tail entity e_t into the corresponding relation space, and defines the score function as $f_r(e_h, e_t) = \| \vec{e_h}\mathbf{M_r} + \vec{r} - \vec{e_t}\mathbf{M_r} \|_2^2$. Inspired by TransR, we project the head entity e_h and the tail entity e_t of a triple into the relation space with a projection matrix $\mathbf{W}_r$, and update their relation r_k as Equation (5).

$$\vec{r_k} = \frac{1}{\mid \mathcal{P}_k \mid}\sum_{(h,t)\in\mathcal{P}_k}(\vec{e_t} - \vec{e_h})\mathbf{W}_r \tag{5}$$

In order not to lose the initial embeddings information during training, our model design a gated mechanism to aggregate the initial embeddings and the updated embeddings with learnable gate values. The equation is as shown in Equation (6).

$$\begin{aligned}\mathbf{E} &= g_{ei}\mathbf{E}_{initial}\mathbf{W}_{te} + g_{eu}\mathbf{E}_{update}\\ \mathbf{R} &= g_{ri}\mathbf{R}_{initial}\mathbf{W}_{tr} + g_{ru}\mathbf{R}_{update}\end{aligned} \tag{6}$$

Where $\mathbf{W}_{te}$ and $\mathbf{W}_{tr}$ are the linearly transform matrices for the initial embeddings of entity $\mathbf{E}_{initial}$ and relation $\mathbf{R}_{initial}$, g_{ei}, and g_{ri} are the memory gate for the initial embeddings of entity $\mathbf{E}_{initial}$ and relation $\mathbf{R}_{initial}$, g_{eu}, and g_{ru} are the update gate for the updated embeddings of entity $\mathbf{E}_{update}$ and relation $\mathbf{R}_{update}$ obtained by Equations (4) and (5), respectively. The scoring function for the GI-KBGAT method is defined as follows:

$$f(\Omega) = \sum_{t_{htk}\in\mathcal{S}}\sum_{t'_{htk}\in\mathcal{S}'}(\| \vec{e_h} + \vec{r_k} - \vec{e_t} \|_1 - \| \vec{e'_h} + \vec{r'_k} - \vec{e'_t} \|_1) \tag{7}$$

Where $\mathcal{S}$ and $\mathcal{S}'$ denotes the set of valid triples $[t_{htk} = (e_h, r_k, e_t)]$ and invalid triples $[t'_{htk} = (e'_h, r'_k, e'_t)]$, respectively, $\| \cdot \|_1$ means L1-norm dissimilarity.

3.3. Decoder

The convolutional structure is used as the base model of our decoder, which transforms the embedding vector to another space and possesses powerful feature extraction ability and good parameter efficiency. The decoder takes the embeddings of entity and relation trained from the encoder as input. We test both Conv-TransE (Shang et al., 2019), and Conv-TransR, which keeps the translational property of TransR ($\vec{e_h}\mathbf{W} + \vec{r_k} \approx \vec{e_t}\mathbf{W}$) with 1D convolution inspired by Conv-TransE to be consistent with encoder, as decoder as shown in **Figure 1**.

The only difference between Conv-TransR and Conv-TransE is that the Conv-TransR model has one more project matrix for entities than Conv-TransE. Following shows the model of Conv-TransR: Conv-TransR uses matrix $\mathbf{W}$ to project the entity e_h into its corresponding relation r_k's space, the result is $\vec{e_h}\mathbf{W}$. This result is then stacked with its corresponding relation embedding $\vec{r_k}$ to get $[\vec{e_h}\mathbf{W}, \vec{r_k}]$ as the input of convolutional network. The convolutional network uses different filters(kernels) $\omega \in \mathbb{R}^{2\times F}(F \in \{1, 2, 3...\})$ to generate different feature maps as Conv-TransE. The scoring function for the Conv-TransR method is defined as below:

$$g(e_h, r_k, e_t) = \tau(\psi([\| \psi([\vec{e_h}\mathbf{W}, \vec{r_k}] \circledast \omega)]\mathbf{W}_c)\vec{e_t}) \tag{8}$$

Where $\circledast$ represents a 1D convolution operation, [‖] denotes vector concatenation which concatenates features output from convolution with different filters ω, $\mathbf{W}_c$ is a learnable weight matrix for linear transformation to projected the concatenation embedding into the tail entity e_t space, ψ is chosen to be a ReLU non-linear function, then the calculated embedding is matched to tail entity e_t by an appropriate distance metric, and the logistic sigmoid function τ is used for scoring finally.

4. EXPERIMENTS AND RESULTS

4.1. Datasets

Through continuous learning, the data scale of intelligent robots will only increase. Therefore, when evaluating our proposed method, we ignore the small datasets and chose two large datasets WN18RR (Dettmers et al., 2018) and FB15k-237 (Toutanova et al., 2015) as the benchmark datasets. WN18RR and FB15k-237 are improved versions of two common datasets WN18 and FB15k (Bordes et al., 2013) derived from WordNet and freebase, respectively, in which all inverse relations have been deleted to prevent direct inference of test triples by reversing training triples. **Table 2** provides statistics of them.

4.2. Training Settings

We follow a two-step training procedure that, we first train our GI-KBGAT to encode information about the graph entities and relations and then train decoder model Conv-TransR to perform the link prediction task. For encoder training, we use the margin ranking loss, use Adam to optimize all the parameters with the initial learning rate set at 0.001, set the entity and relation embedding dimension of the last layer to 200, and set the other hyper-parameters for each dataset to be the same as KBGAT (Nathani et al., 2019). For decoder training, we use the standard binary cross-entropy loss with label smoothing, set the size and number of the kernel to 9 and 200, respectively, and set the other hyper-parameters for each dataset to be the same as InteractE (Vashishth et al., 2020).

4.3. Evaluation Protocol

Following the previous work, we use the filtered setting (Bordes et al., 2013) that all valid triples are filtered out from the candidate set while evaluating test triples. The performance is reported on the standard evaluation metrics: Mean Reciprocal Rank (MRR) and the proportion of correct entities ranked in the top 1, 3, and 10 (Hits@1, Hits@3, Hits@10).

4.4. Results and Analysis

Table 3 presents the experimental results of our methods and several baseline methods on FB15K-237 and WN18RR test sets. In which all values are presented in percentage. Among these baseline methods, the methods in the first box, namely TransE (Bordes et al., 2013), ConvE (Dettmers et al., 2018), ConvKB (Nguyen et al., 2018), Conv-TransE (Shang et al., 2019), and InteractE (Vashishth et al., 2020), have their results taken from the original paper and can be resumed to acceptable results. We compared our methods with these methods in **Table 3** to label the best score in bold.

TABLE 2 | Statistics of the experimental datasets.

Dataset	# Entities | $\mathcal{E}$ |	# Relations | $\mathcal{R}$ |	# Edges | $\mathcal{T}$ |				Mean in-degree	Median in-degree
			Training	Validation	Testing	Total		
WN18RR	40,943	11	86,835	3,034	3,134	93,003	2.12	1
FB15k-237	14,541	237	272,115	17,535	20,466	310,116	18.71	8

TABLE 3 | Experimental results on FB15K-237 and WN18RR test sets.

Models	FB15k-237				WN18RR			
	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10
TransE Bordes et al., 2013	29.4	–	–	46.5	22.6	–	–	50.1
ConvE Dettmers et al., 2018	32.5	23.7	35.6	50.1	43	40	44	52
ConvKB Nguyen et al., 2018	**39.6**	–	–	51.7	24.8	–	–	52.5
Conv-TransE Shang et al., 2019	33	24	37	51	46	43	47	52
InteractE Vashishth et al., 2020	35.4	**26.3**	–	53.5	46.3	43.0	–	52.8
SACN Shang et al., 2019	35	26	39	54	47	43	48	54
KBGAT Nathani et al., 2019	20.5	11.4	22.8	39.6	40.4	32.2	44.8	55.4
Our encoder model (+ ConvTransE)	35.5	**26.3**	**39.1**	**53.8**	45.9	42.8	46.7	52.5
Our encoder model (+ ConvTransR)	33.9	24.4	37.6	52.8	**46.6**	**43.4**	**47.9**	**52.9**
Our encoder model (+ InteractE)	35.5	26.2	**39.2**	**54.1**	**46.7**	**43.5**	**48.1**	**52.9**

All values are in percentage and the best scores of our model is in bold regardless of the second box (SACN and KBGAT).

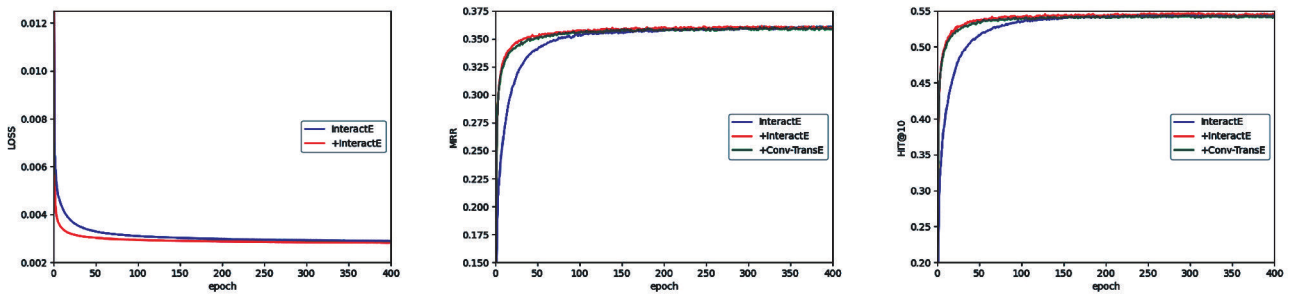


FIGURE 2 | The convergence study of InteractE and our encoder model with InteractE as decoder (represented by "+InteractE") and with Conv-TransE as decoder (represented by "+Conv-TransE") in FB15k-237 using the validation set. Here we only report the results of loss, MRR and Hit@10.

TABLE 4 | Ablation experimental results on WN18RR test sets.

Models	WN18RR			
	MRR	**H@1**	**H@3**	**H@10**
KBGAT encoder (+ KBGAT decoder)	40.4	32.2	44.8	55.4
Our encoder model (+ KBGAT decoder)	41.1 (+0.7)	33.1 (+0.9)	45.5 (+0.7)	55.9 (+0.5)
Our encoder model (+ ConvTransR)	46.6	43.4	47.9	52.9
−gate	46.1 (−0.5)	43.0 (−0.4)	47.3 (−0.6)	52.4 (−0.5)
−rel	46.2 (−0.4)	43.0 (−0.4)	47.5 (−0.4)	52.7 (−0.2)
−gate − rel	46.1 (−0.5)	42.9 (−0.5)	47.3 (−0.6)	52.2 (−0.7)

Since our method is inspired by methods SACN (Shang et al., 2019) and KBGAT (Nathani et al., 2019) that we present their results in the second box. The results of the SACN model are obtained from its corresponding paper, but this model requires a large GPU to train and these results can not be reproduced with the authors' code. KBGAT has test data leakage in its original implementation that the results in its paper are not credible. In our experiment results table, we fix the problem and show the correct results of the model.

We first compare our model use Conv-TransE as the decoder with the Conv-TransE model. Our model performs better than Conv-TransE on both datasets. Especially in the FB15K-237 dataset, our model improves upon Conv-TransE's MRR by a margin of 7.6%, Hits@1 of 9.6%, Hits@3 of 5.7%, and Hits@10 of 5.5%. In the WN18RR dataset, our model improves upon Conv-TransE's Hits@10 by a margin of 1.0%. Under the same accuracy, our model achieves the same performance on the other metrics compared with Conv-TransE.

Second, we compare our model use InteractE as the decoder with the InteractE model to better prove the effectiveness of our encoder. As shown in the **Table 3**, compared with the original model, most metrics of InteractE have been improved after our encoder model is added. For example, our model with InteractE improves upon InteractE's MRR by a margin of 0.3% and Hits@10 of 1.1% in the FB15K-237 dataset.

Third, we compare our model with the other baseline models. In the FB15K-237 dataset, our model with Conv-TransE as decoder achieves the best performance in Hit@3 and Hit@10, and tied for the best in Hit@1. In the WN18RR dataset, our model with Conv-TransR as decoder achieves the best performance in all metrics. Meanwhile, these two models both can achieve the top three effects on the other datasets. In conclusion, our model can achieve the best results on both datasets FB15K-237 and WN18RR.

Figure 2 shows the convergence of the three models: InteractE, our encoder model with InteractE as the decoder, and Conv-TransE as the decoder. Because Conv-TransE uses a different loss function, we do not put its loss result for comparison. We can see that our models (the red line and green line) are always better than InteractE (the blue line) under MRR and Hit@10. And our models converge faster than InteractE.

4.5. Ablation Experiment

In order to prove the validity of our model, we do some ablation experiments on WN18RR dataset to show the influences of different parts of our model. The results of the ablation experiments are shown in **Table 4**. Compared with the KBGAT model, our improved encoder with the KBGAT decoder (convKB) can achieve better performance as shown in the first box. After changing the decoder methods, our method performs more superior as in the second box. Combined with **Table 3**, the decoders we use are both better than the ConvKB which is used by KBGAT, which shows the effectiveness of our decoder chosen.

To better show the influences of our innovation in KBGAT, we also test the influence of different parts in our encoder. We separately remove the gate mechanism, the interaction mechanism, and both of them to see the impact on results. The results are shown in the second box of **Table 4**. The gate

mechanism and the interaction mechanism both perform a similar influence on the encoder model. And the best result can be achieved by combining them in the KBGAT model as the encoder model.

For these results, we conclude that our encoder-decoder model can better the expressive performance of entity embeddings and relation embeddings, and can achieve competitive success in modeling knowledge graphs. Since the hyper-parameters of our model for each dataset are set to the same as the existing methods and no parameter tuning is performed to obtain the best performance, we believe that the performance of our model can still be improved with parameter tuning.

5. CONCLUSION

In this paper, we propose a novel approach for knowledge graph relation prediction, which can be used in intent understanding in human-robot interaction and in robots' knowledge graph completion. Our methods can work well in large-scale knowledge graphs and can be extended to learn embeddings for various applications of robots, such as dialog generation and question answering.

In the future, we intend to extend our method to work as an end-to-end work and consider the attribute information and temporal information into our model to improve the ability to handle complex knowledge graphs. And we also intend to test our work in a real robot's "brain" to test the ability of our model in actual work.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found at: two datasets used: WN18RR and FB15k-237. Download from https://github.com/TimDettmers/ConvE.

AUTHOR CONTRIBUTIONS

YS, KC, and CL contributed to the conception and design of the study. KC wrote the sections of the code. CL wrote the parts of the first draft of the manuscript. YS organized the experiments and the manuscript. AL and HT directed the whole work and revised the manuscript. All authors approved the submitted version.

FUNDING

This work described in this paper was partially supported by the National Key R&D Program of China (Nos. 2016QY03D0603, 2017YFB0802204, 2019QY1406, and 2017YFB0803303), the Key R&D Program of Guangdong Province (No. 2019B010136003), and the National Natural Science Foundation of China (Nos. 61732004, 61732022, 62072131, and 61672020).

REFERENCES

Bordes, A., Usunier, N., García-Durán, A., Weston, J., and Yakhnenko, O. (2013). "Translating embeddings for modeling multi-relational data," in *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a Meeting Held December 5–8, 2013*, eds C. J. C. Burges, L. Bottou, Z. Ghahramani, and K. Q. Weinberger (Lake Tahoe, NV), United States, 2787–2795.

Dettmers, T., Minervini, P., Stenetorp, P., and Riedel, S. (2018). "Convolutional 2D knowledge graph embeddings," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18), The 30th Innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), February 2–7, 2018*, eds S. A. McIlraith and K. Q. Weinberger (New Orleans, LA: AAAI Press), 1811–1818.

Ebisu, T., and Ichise, R. (2018). "Toruse: knowledge graph embedding on a lie group," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18), The 30th Innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), February 2–7, 2018*, eds S. A. McIlraith and K. Q. Weinberger (New Orleans, LA: AAAI Press), 1819–1826.

Gu, Z., Wang, L., Chen, X., Tang, Y., Wang, X., Du, X., et al. (2021). Epidemic risk assessment by a novel communication station based method. *IEEE Trans. Netw. Sci. Eng.* 1. doi: 10.1109/TNSE.2021.3058762. [Epub ahead of print].

Ji, G., He, S., Xu, L., Liu, K., and Zhao, J. (2015). "Knowledge graph embedding via dynamic mapping matrix," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26–31, 2015, Volume 1: Long Papers* (Beijing: The Association for Computer Linguistics), 687–696. doi: 10.3115/v1/P15-1067

Jia, Y., Gu, Z., and Li, A. (2021). *MDATA: A New Knowledge Representation Model*. Springer International Publishing. doi: 10.1007/978-3-030-71590-8

Kipf, T. N., and Welling, M. (2017). "Semi-supervised classification with graph convolutional networks," in *5th International Conference on Learning Representations, ICLR 2017, April 24–26, 2017, Conference Track Proceedings* (Toulon: OpenReview.net).

Lin, Y., Liu, Z., Sun, M., Liu, Y., and Zhu, X. (2015). "Learning entity and relation embeddings for knowledge graph completion," in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25–30, 2015*, eds B. Bonet and S. Koenig (Austin, TX: AAAI Press), 2181–2187.

Nathani, D., Chauhan, J., Sharma, C., and Kaul, M. (2019). "Learning attention-based embeddings for relation prediction in knowledge graphs," in *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, July 28–August 2, 2019, Volume 1: Long Papers*, eds A. Korhonen, D. R. Traum, and L. Màrquez (Florence: Association for Computational Linguistics), 4710–4723. doi: 10.18653/v1/P19-1466

Nguyen, D. Q., Nguyen, T. D., Nguyen, D. Q., and Phung, D. Q. (2018). "A novel embedding model for knowledge base completion based on convolutional neural network," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, June 1–6, 2018, Volume 2 (Short Papers)*, eds M. A. Walker, H. Ji, and A. Stent (New Orleans, LA: Association for Computational Linguistics), 327–333. doi: 10.18653/v1/N18-2053

Nickel, M., Rosasco, L., and Poggio, T. A. (2016). "Holographic embeddings of knowledge graphs," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12–17, 2016*, eds D. Schuurmans and M. P. Wellman (Phoenix, AZ: AAAI Press), 1955–1961.

Nickel, M., Tresp, V., and Kriegel, H. (2011). "A three-way model for collective learning on multi-relational data," in *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, June 28–July 2, 2011*, eds L. Getoor and T. Scheffer (Bellevue, WA: Omnipress), 809–816.

Shang, C., Tang, Y., Huang, J., Bi, J., He, X., and Zhou, B. (2019). "End-to-end structure-aware convolutional networks for knowledge base completion," in *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, January 27–February 1, 2019* (Honolulu, HI: AAAI Press), 3060–3067. doi: 10.1609/aaai.v33i01.33013060

Toutanova, K., Chen, D., Pantel, P., Poon, H., Choudhury, P., and Gamon, M. (2015). "Representing text for joint embedding of text and knowledge bases," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, September 17–21, 2015*, eds L. Màrquez, C. Callison-Burch, J. Su, D. Pighin, and Y. Marton (Lisbon: The Association for Computational Linguistics), 1499–1509. doi: 10.18653/v1/D15-1174

Trouillon, T., Welbl, J., Riedel, S., Gaussier, É., and Bouchard, G. (2016). "Complex embeddings for simple link prediction," in *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, June 19–24, 2016, Volume 48 of JMLR Workshop and Conference Proceedings*, eds M. Balcan and K. Q. Weinberger (New York, NY: JMLR.org), 2071–2080.

Vashishth, S., Sanyal, S., Nitin, V., Agrawal, N., and Talukdar, P. P. (2020). "Interacte: improving convolution-based knowledge graph embeddings by increasing feature interactions," in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, February 7–12, 2020* (New York, NY: AAAI Press), 3009–3016. doi: 10.1609/aaai.v34i03.5694

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). "Attention is all you need," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4–9, 2017*, eds I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, et al. (Long Beach, CA), 5998–6008.

Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y. (2018). "Graph attention networks," in *6th International Conference on Learning Representations, ICLR 2018, April 30–May 3, 2018, Conference Track Proceedings* (Vancouver, BC: OpenReview.net).

Wang, Z., Zhang, J., Feng, J., and Chen, Z. (2014). "Knowledge graph embedding by translating on hyperplanes," in *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27–31, 2014*, eds C. E. Brodley and P. Stone (Québec City, QC: AAAI Press), 1112–1119.

Xiao, H., Huang, M., Hao, Y., and Zhu, X. (2015). Transg: a generative mixture model for knowledge graph embedding. *CoRR* abs/1509.05488.

Yang, B., Yih, W., He, X., Gao, J., and Deng, L. (2015). "Embedding entities and relations for learning and inference in knowledge bases," in *3rd International Conference on Learning Representations, ICLR 2015, May 7–9, 2015, Conference Track Proceedings*, eds Y. Bengio and Y. LeCun (San Diego, CA).

Zhang, J., Shi, X., Xie, J., Ma, H., King, I., and Yeung, D. (2018). "Gaan: gated attention networks for learning on large and spatiotemporal graphs," in *Proceedings of the Thirty-Fourth Conference on Uncertainty in Artificial Intelligence, UAI 2018, August 6–10, 2018*, eds A. Globerson and R. Silva (Monterey, CA: AUAI Press), 339–349.

Zhao, X., Jia, Y., Li, A., Jiang, R., and Song, Y. (2020). Multi-source knowledge fusion: a survey. *World Wide Web* 23, 2567–2592. doi: 10.1007/s11280-020-00811-0

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

frontiers in Neurorobotics

ORIGINAL RESEARCH
published: 20 May 2021
doi: 10.3389/fnbot.2021.674949

Dynamic Task Allocation in Multi-Robot System Based on a Team Competition Model

Kai Jin[1]*, *Pingzhong Tang*[2], *Shiteng Chen*[3] and *Jianqing Peng*[4]

[1] Department of Computer Science and Engineering, The Hong Kong University of Science and Technology (HKUST), Hong Kong, China, [2] Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing, China, [3] Institute of Software, Chinese Academy of Sciences, Beijing, China, [4] School of Intelligent Systems Engineering, Sun Yat-sen University, Shenzhen, China

In recent years, it is a trend to integrate the ideas in game theory into the research of multi-robot system. In this paper, a team-competition model is proposed to solve a dynamic multi-robot task allocation problem. The allocation problem asks how to assign tasks to robots such that the most suitable robot is selected to execute the most appropriate task, which arises in many real-life applications. To be specific, we study multi-round team competitions between two teams, where each team selects one of its players simultaneously in each round and each player can play at most once, which defines an extensive-form game with perfect recall. We also study a common variant where one team always selects its player before the other team in each round. Regarding the robots as the players in the first team and the tasks as the players in the second team, the sub-game perfect strategy of the first team computed via solving the team competition gives us a solution for allocating the tasks to the robots—it specifies how to select the robot (according to some probability distribution if the two teams move simultaneously) to execute the upcoming task in each round, based on the results of the matches in the previous rounds. Throughout this paper, many properties of the sub-game perfect equilibria of the team competition game are proved. We first show that uniformly random strategy is a sub-game perfect equilibrium strategy for both teams when there are no redundant players. Secondly, a team can safely abandon its weak players if it has redundant players and the strength of players is transitive. We then focus on the more interesting case where there are redundant players and the strength of players is not transitive. In this case, we obtain several counterintuitive results. For example, a player might help improve the payoff of its team, even if it is dominated by the entire other team. We also study the extent to which the dominated players can increase the payoff. Very similar results hold for the aforementioned variant where the two teams take actions in turn.

Keywords: team competition, task allocation, multi-robot system, dominated players, sub-game perfect equilibrium

OPEN ACCESS

Edited by:
Zheng Wang,
Southern University of Science and Technology, China

Reviewed by:
Xiaodong Li,
The University of Hong Kong, Hong Kong
Arkadi Predtetchinski,
Maastricht University, Netherlands
Vincent Chau,
Chinese Academy of Sciences (CAS), China

***Correspondence:**
Kai Jin
cscjjk@gmail.com

Received: *02 March 2021*
Accepted: *13 April 2021*
Published: *20 May 2021*

Citation:
Jin K, Tang P, Chen S and Peng J (2021) Dynamic Task Allocation in Multi-Robot System Based on a Team Competition Model. Front. Neurorobot. 15:674949. doi: 10.3389/fnbot.2021.674949

1. INTRODUCTION

In the past two decades, intelligent multi-robot systems are more and more widely used in industrial manufacturing, agriculture, hospital, fire rescue, cargo handling, entertainment, and many other places. The efficiency of the systems is crucial to their applications and it highly depends on the collaboration between robots. One of the primary problem that occurs to the designer of multi-robot systems is how to assign tasks to robots such that the most suitable robot is selected to execute the most appropriate task, which is usually referred to as the *task allocation problem* and which arises in all kinds of real-life applications.

It is well-known that game theory lays the mathematical foundation for the research of collaboration in multi-robot system, and it is a trend to integrate the ideas and theoretical results in game theory into the research of multi-robot system. For example, market-based approaches to task allocation are proposed in Botelho and Alami (1999), Gerkey and Mataric (2002), Wang et al. (2004), Dias et al. (2006), Zlot and Stentz (2006), and Wu and Shang (2020). In this paper, a team-competition model, which is of interest by itself in game theory, is proposed to solve a Dynamic Multi-Robot Task Allocation (DMRTA) problem.

In our DMRTA problem, there are m robots and n pre-described tasks, and the tasks are coming in T rounds, where $T \leq \min\{m, n\}$. One task will come in every round, and it would like to be assigned to one robot immediately. Be aware that when $T < n$, only T tasks, but not all the n tasks, will be assigned (as there are only T rounds), and this means the set of tasks to be assigned are not fully determined at the beginning in our problem. For simplicity, assume that each pre-described task comes at most once (this constraint can easily be removed by replicate the task) and that each robot can take at most one task (as well, this constraint can be removed by replicate the robot). Different robots have different performances in solving different tasks, and to describe this diversity we make the following assumption: if the robot with index r (i.e., robot r) is to execute the task with index s (i.e., task s), there is a probability $p_{r,s}$ that robot r succeeds to complete its job—and a probability $1 - p_{r,s}$ that it fails to do its job. The $m \times n$ probability matrix $p_{r,s} \mid 1 \leq r \leq m, 1 \leq s \leq n$ is prior information—it is given to us before the allocation mission starts. Roughly, our objective is that the number of successful robots is as high as possible—in other words, the number of tasks that have been solved successfully is as high as possible. Details of the DMRTA problem will be elaborated right after we introduce our team competition model in what follows.[1]

We investigate a type of *team competitions* where there are two teams, each with a number of players, competing against each other. The competition proceeds in a fixed number of rounds. In each round, each team simultaneously sends out a player to a match (We also consider a variant where one team, say Team 2 without loss of generality, always takes actions before the other team in each round. This variant will mainly be discussed in section 3). The result of the match is then revealed according to a probabilistic strength matrix between players. The selected players cannot compete in the subsequent rounds. The competition proceeds to the next round if there is one; or terminated otherwise. The format of the competition and the strength matrix are common knowledge to both teams. The final payoff of each team is the number of matches it wins. To make it more general, we also investigate another commonly seen form where each team gets payoff 1 if it wins strictly more matches than the other team, 0 if ties, and -1 if it wins less matches. Clearly, this competition between two teams defines a standard *extensive-form game*, or more precisely, a *stacked matrix game* (Lanctot et al., 2014). We are interested in the sub-game perfect equilibria of the game, i.e., a strategy profile that specifies for each team which player to play at each round. A formal description of our team competition model is given in the next section.

Our team competition model is first motivated by the Chinese horse race story described in Tang et al. (2009) (see also Wikipedia, 2015b). It represents one of the most popular forms of horse races where each team ranks its horses to match sequentially. Moreover, the Swaythling Cup, as known as World Table Tennis Championships, follows the same model described in our paper: each team adaptively selects a ranking of three players and brings two additional substitutes. In fact, this has been one of the most popular formats of team competition in table tennis. In addition, the card game Goofspeil (Lanctot et al., 2014; Wikipedia, 2015a) also falls into nearly the same model as described in our paper. Last but not least, many military engagements (like fighting between two groups of drones) may also have this type of structure.

Obviously, we can regard the m robots as the players in the first team and the n tasks as the players in the second team, and the probability matrix $p_{r,s} \mid 1 \leq r \leq m, 1 \leq s \leq n$ can serve as the probabilistic strength matrix in the team competition. Then, the (sub-game perfect) strategy profile of the first team we obtained via solving the team competition gives us a solution for allocating the tasks to the robots—it specifies how to select the robot (randomly according to some probability distribution given by the strategy) to execute the upcoming task in each round, based on the results of the matches in the previous rounds.

At this place, it is necessary to point out a feature of our DMRTA problem inherited from the team competition model: As the two teams shall send out their players simultaneously in each round, in our DMRTA problem we shall select the robot before the task in the corresponding round is revealed to us. Nevertheless, if we would like to handle the case where we can select the robot after the task in the corresponding round is revealed, we only need to deal with the variant where Team 2 reacts before Team 1. Gladly, we will see in section 3 that many results for this non-simultaneous variant are aligned with the results for the original simultaneous case (In particular, most of the results are the same. See **Table 1** for a comparison). We first discuss the simultaneous case because this case is typical and more difficult.

Despite the aforementioned application in the DMRTA problem, our team competition model is interesting by itself in

[1]This team competition model was introduced in the conference version of this paper, which was accepted by the 2016 International Conference on Autonomous Agents and Multiagent Systems (Jin et al., 2016) [Yet the conference version did not discuss (1) its application in DMRTA and (2) the react-in-turn variant].

TABLE 1 | Summarize of the results for simultaneous case and non-simultaneous case.

	Simultaneous case	**Non-simultaneous case**
$n = m = T$	The value of the game equals the average utility (of Team 1) across all matchings. Moreover, the uniformly random strategy is a SPE strategy (Theorem 1).	The value equals the maximal utility that Team 1 can gain over all matchings. The pure strategy ensuring the best matching for Team 1 is a SPE strategy (Theorem 5).
Transitive strength	If $A_m \leq \ldots \leq A_1, A_{T+1}, \ldots, A_m$ can be removed. If $B_n \leq \ldots \leq B_1, B_{T+1}, \ldots, B_n$ can be removed (Theorem 2).	Exactly the same result holds (Theorem 6).
For non-transitive strength: *When other players are weaker than the first T players in a team, can the weaker players be removed?*	For $U = U_E$, the answer is YES as long as $n = T$. For other utility functions U (e.g., U_M), the answer is NO (Theorem 3).	For Team 1, the answer is YES as long as $n = T$ (this holds for any U). However, the answer is NO for Team 2 (Theorem 7).
For non-transitive strength: *How many dominated players shall we recruit to achieve maximum value?*	$\begin{cases} T-1 & \text{for } U = U_E; \\ \lfloor T/2 \rfloor & \text{for } U = U_M. \end{cases}$ (Theorem 4)	Exactly the same result holds (Theorem 8).

game theory. We are particularly interested in a situation where at least one team has more players than the number of rounds in the competition. As a result, some players will never have chance to participate in any match. A main agenda of this paper is to understand to what extent can the presence of additional players affect the payoff of both teams. In particular, we ask the following questions: (1) Can the presence of additional weakest teammate, a teammate whose row in the strength matrix is strictly dominated by any other row, help increase the payoff of the team? (2) Can the presence of additional dominated teammate, a teammate that always loses to any player in the opponent team, help increase the payoff of the team? It might appear intuitive that the answers to both questions are negative. For the first question, it seems that the weakest teammate will never have a chance to participate in any match since one can always replace him by a better teammate and increase payoff. For the second question, it might seem more obvious since the dominated teammate will lose any matches thus must be replaced by a better teammate. To our surprise, we find that the answers to both questions are affirmative.

Our contributions to the team competition model are summarized in the following. We first show that uniformly random strategy is a sub-game perfect equilibrium strategy for both teams when there are no redundant players (i.e., the number of players in each team equals the number of rounds). The uniformly random strategy always picks the unmatched player uniformly at random in each round. Then, we consider the general case where at least one team has redundant players. We first study the case where the strength of players is transitive (see Definition 2), which means that the players can be rearranged in a queue so that each of them is weaker than its successor. We prove that, a team can safely abandon its weak players if it has redundant players and the strength of players is transitive. Therefore, this case reduces to the case where there are no redundant players. Finally, we focus on the case where there are redundant players and the strength of players is not transitive. In this case, we obtain a number of counterintuitive results. Most importantly, a player might help improve its team's payoff, even if it is dominated by the entire opposing team. We give a necessary condition for a dominated player to be useful, which alternatively suggest that a particular utility function (named U_E below) is more reasonable in team competition. Our results imply that a team can increase its utility by recruiting additional dominated players. We further show that, the optimal number of dominated players to recruit can scale with the number of rounds. More precisely, this number can be $\Theta(T)$ if there are T rounds. Last but not least, we study the limitation of dominated players. These results bring insights into playing and designing general team competitions.

Team competition has been studied for years. Tang et al. (2009, 2010) study a team competition setting where the number of players equals the number of rounds and both teams must determine the ordering of players upfront, before the competition starts. They put forward competition rules that are truthful while satisfy other desirable properties. The main difference between their work and ours is that we do not design new mechanisms but study game theoretical properties of commonly used competition rules. The differences also lie in that the strategies are adaptive in our setting and each team can have more players than the rounds. The strategic aspects of team competition have also been under scrutiny of computer scientists due to a recent Olympic scandal in badminton, where several teams deliberately throw matches in order to avoid a strong opponent in the next round. The phenomenon has been discussed in depth in a series of algorithmic game theory blogposts by Kleinberg (2012) and Procaccia (2013). A parallel literature has been concerned with the strategic aspect of tournament seeding (Hwang, 1982; Rosen, 1986; Knuth, 1987; Schwenk, 2000; Altman et al., 2009; Vu et al., 2009). It is well-known that there are cases where by strategic seeding and structuring, any player can be winner in knockout tournament. Various game theoretical questions, such as player-optimal seeding, complexity of manipulation and incentives to guarantee strategyproofness, have been investigated in this literature.

The study of various kinds of multi-robot task allocation problem using game theory dates back to early 2000's. Botelho and Alami (1999), Gerkey and Mataric (2002), Wang et al. (2004), Dias et al. (2006), Zlot and Stentz (2006), and Wu and Shang (2020) make use of the theory of market economies to determine how to allow robots to negotiate on responsibilities in task allocation. In particular, they discuss how to manage bids (robots communicate to bid for tasks according to their expected contribution to the tasks), how to handle bids in parallel and how to handle multiple tasks at once, and so on. Usually, heuristic assignments are made by assigning every task to the robot that can execute it with the highest utility.

2. MATERIALS AND METHODS

Team 1 has a set of m players $\{A_1, \ldots, A_m\}$. Team 2 has a set of n players $\{B_1, \ldots, B_n\}$. A competition between team 1 and 2 is a tuple $G(T, P, U)$ where,

1. T is the number of rounds.
2. In each round, each team **simultaneously** selects one of its players that have not been selected yet.
3. P is a probabilistic matrix that describes the relative strength between players, with $P_{i,j}$ denoting the probability that A_i wins against B_j and $1 - P_{i,j}$ the probability for A_i to lose to B_j.
4. $U : [T] \to R$ denotes the utility function of each team. The utility function only depends on the number of rounds t each team wins, i.e., it can be represented by $U(t)$. This also implies that both teams have the same utility functions.
5. The parameters n, m, T, P, U are common knowledge to both teams, and historical plays are perfectly observable. It is assured that $P_{i,j} \in [0, 1]$ for all i, j and that $m \geq T$ and $n \geq T$ so that there are enough players to complete the competition.

The following utility functions U_E and U_M are two commonly seen ones:

$$U_E(t) = t - T/2. \quad U_M(t) = \begin{cases} 1 & t > T/2 \\ 0 & t = T/2 \\ -1 & t < T/2 \end{cases}. \tag{1}$$

In other words, U_E describes a competition where a team's utility is exactly the number of rounds it wins (minus some constant $T/2$); while $U_M(t)$ describes a competition where a team's utility is whether it wins more than its opponent. Notice that, when $U = U_E$ or $U = U_M$, we have $U(t) + U(T - t) = 0$, hence both utility functions define a zero-sum game. In this paper we always assume that $U(T) + U(T - t) = 0$.

2.1. Example: Simultaneous Card Games

The models above formulates the standard team competitions as commonly seen under the context of sports, but shall not be limited to sports. The following is an instance of card games that fall into our framework. Suppose that Alice and Bob each has a deck of three cards. In each deck one card is in suit ♡ and two cards are in suit ♠. They play three rounds; in each round Alice and Bob select one card and they reveal the cards simultaneously. If they select cards in same suit (both in ♡ or both in ♠), Alice wins this round; otherwise Bob wins this round. The one who wins two or three rounds gets utility 1; the other one wins zero or one rounds and it gets utility −1.

This game can be conveniently represented in our model using the following parameters:

$$m = n = T = 3, P = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}, U = U_M.$$

For this game, applying our first theorem, a (sub-game perfect) equilibrium strategy for both players exists, and it is to just to play uniformly random. It follows that Alice and Bob have utilities $-1/3$ and $+1/3$.

2.2. Extensive-Form Game With Perfect Recall

Any particular instance $G(T, P, U)$ of our team competition is an extensive-form game. In this game, a history can be described by a tuple $(k, \mathbf{a}, \mathbf{b}, \mathbf{c})$ where:

- k indicates the number of rounds that has been played;
- $\mathbf{a}$ is a k-dimensional vector which stores the players selected by Team 1 in the past k rounds;
- $\mathbf{b}$ stores the players selected by Team 2;
- $\mathbf{c}$ is a k-dimensional 0–1 vector which stores the results in the first k rounds, where 0 corresponds to a lose by Team 1 and 1 corresponds to a win by Team 1.

A behavioral strategy in this game is a mapping from every history to a probability distribution over actions. That is, at each history, the strategy of each team is to pick the next player according to a probability distribution. By Kuhn' Theorem (Kuhn, 1953; Osborne and Rubinstein, 1994), there is a sub-game perfect equilibrium (SPE), in which both teams use behavioral strategies. According to the SPE, a value $V(H)$ can be defined for each history H, which indicates the expected utility that Team 1 would get at the end of the game if it is now at history H. Note that each history is the root of a sub-game and so the value of a history is the same as the value of the sub-game.

2.3. Computing SPE

By backward induction, one can easily get

Lemma 1. *Two histories have the same value if they have selected the same players to play (but may be in different orders) and Team 1 won the same number of rounds.*

Based on the above lemma, the histories can be partitioned to equivalence classes, such that each equivalence class corresponds to a four-tuple (k, X, Y, w): k is a number in $[T]$ which denotes the number of past rounds; X is a subset of A of size k; Y is a subset of B of size k; X, Y denote the players that have played; w is a number in $[k]$ which denotes how many rounds Team 1 has won so far.

In the following, we show in detail how to compute the value of each equivalence class via dynamic programming.

Let $V[k, X, Y, w]$ denote the expected utility of Team 1 when the history belongs to class (k, X, Y, w).

Clearly, we have $V[k, X, Y, w] = U(w)$ when $k = T$.

When $k < T$, computing $V[k, X, Y, w]$ reduces to computing the value of the matrix game $M(k, X, Y, w)$ where the matrix $M(k, X, Y, w)$ is defined as follows:

It consists of $m - k$ rows and $n - k$ columns. Each row corresponds to a player in $A - X$, and each column corresponds to a player in $B - Y$. The cell corresponding to A_i, B_j equals to the expect utility of Team 1 when Team 1 and Team 2, respectively make action A_i and B_j on the current state (k, X, Y, w), which equals

$$V[k+1, X+\{A_i\}, Y+\{B_j\}, w+1] \cdot P_{i,j} + V[k+1, X+\{A_i\}, Y+\{B_j\}, w] \cdot (1 - P_{i,j}).$$

The reason behind the above definition of $M(k, X, Y, w)$ is as follows. If the two teams select A_i, B_j in this round, Team 1 has probability $P_{i,j}$ to win this round and hence the history becomes $(k+1, X+\{A_i\}, Y+\{B_j\}, w+1)$; besides, Team 1 has probability $(1 - P_{i,j})$ to lose this round and hence the history becomes $(k+1, X+\{A_i\}, Y+\{B_j\}, w)$.

We can compute the value of all equivalent classes of histories according to the above induction. In fact, by computing these values, we also find a sub-game perfect behavior strategy for both players. To see this, suppose that (k, X, Y, w) is a non-terminal equivalent class of history. On solving the matrix game $M[k, X, Y, w]$ we find the strategies for all the histories in the history class (k, X, Y, w).

2.4. Uniformly Random Strategies

The next theorem states that, if there are no redundant players, uniformly random is an equilibrium strategy for the teams. It holds for arbitrary utility function including U_E and U_M.

Definition 1. *The* uniformly random strategy *is a behavioral strategy, in which a team always selects from the remaining players uniformly at random in each round.*

Theorem 1. *When both teams have no redundant players (i.e., $n = m = T$), then it is a SPE when both teams apply the uniformly random strategy.*[2]

We apply the following lemma for proving Theorem 1.

Lemma 2. *Suppose that there are no redundant players. Let $\mathbb{S}$ denote the set of all perfect matchings between $\{A_1, \ldots, A_T\}$ and $\{B_1, \ldots, B_T\}$. If Team 1* ***or*** *Team 2 applies the uniformly random strategy, then the probability that the competition ends with any fixed matching in $\mathbb{S}$ is exactly $1/(T!)$.*

Proof of Lemma 2*:* We only prove that the statement holds when Team 1 applies the uniformly random strategy. Symmetrically, the statement holds when Team 2 applies the uniformly random strategy.

First, suppose that Team 1 applies the uniformly random strategy while Team 2 applies an arbitrary pure strategy.[3] In this case, we claim that the probability that the competition ends with any fixed matching is exactly $1/(T!)$. This can be proved by induction on the number of remaining τ rounds. In the stage with τ remaining rounds, let M be any fixed matching between the τ unused players in Team 1 and the τ unused players in Team 2. In the next round, it occurs with probability $1/\tau$ that some edge e of M is chosen, because Team 1 will assign the player B_i selected by Team 2 to a random player among the unused players in Team 1. If this occurs, denote by $M' = M - \{e\}$ the matching obtained by deleting e from M, which is chosen with probability $1/((\tau - 1)!)$ by induction hypothesis. Thus, M is chosen with probability $1/(\tau!)$.

Finally, since a mixed strategy is a linear combination of the pure strategies, our job is done. □

Proof of Theorem 1*:* Let $\mathbb{S}$ be the same set as in Lemma 2. As there are no redundant players, a game will always end with some matching in $\mathbb{S}$. For any matching $s \in \mathbb{S}$, let Z_s denote the event that the game ends with this matching. If Team 1 applies the uniformly random strategy, it has expected utility

$$\sum_{s\in\mathbb{S}} E(\text{the utility of Team 1} \mid Z_s) / \Pr_{\text{Team 1 uniformly random}}(Z_s) = \sum_{s\in\mathbb{S}} E(\text{the utility of Team 1} \mid Z_s)/(T!)$$

The second equation is according to Lemma 2, which states $\Pr_{\text{Team 1 uniformly random}}(Z_s) = 1/(T!)$.

Similarly, if Team 2 applies the uniformly random strategy, it will get

$$\sum_{s\in\mathbb{S}} E(\text{the utility of Team 2} \mid Z_s)/(T!) = \sum_{s\in\mathbb{S}} -E(\text{the utility of Team 1} \mid Z_s)/(T!)$$

Therefore, it is a Nash Equilibrium if both teams apply the uniformly random strategy. The argument can be similarly extended to show that it is SPE. □

In the remainder of this paper, we focus on the case where there are redundant players.

Claim 1. *If there are redundant players, then the uniformly random strategy may not be a SPE strategy (For any team, a SPE strategy for this team requires that it is optimal in each subgame).*

This claim is obvious for a team with redundant players; but less obvious for a team without redundant players. Here we give an example in which the uniformly random strategy is not a SPE strategy for a team with no redundant players.

[2]Note that there could be other SPEs. For example, when players in Team 1 always lose, any strategies for the two teams form a SPE.

[3]A pure strategy does NOT mean it determines the entire order of players at the beginning. Instead, it means that at each possible history, some unmatched player will be selected deterministically in the upcoming round. In this way, any mixed strategy is a linear combination of the pure strategies.

Example 1. *Let* $m = T = 2, n = 3$, $U = U_E = U_M$ ($U_E = U_M$ *when* $T = 2$), $P = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$.

According to method given in subsection 2.3, we can compute that $M(0, \varnothing, \varnothing, 0) = \begin{pmatrix} -1 & -1 & 1 \\ 0 & 0 & -1 \end{pmatrix}$. Therefore, in the SPE, the behavior for Team 1 on the initial state should be select A_1 with probability 1/3 and select A_2 with probability 2/3. This guarantees (expected) utility $-1/3$. If to the contrary that Team 1 adopts the uniformly random strategy, it should select A_1, A_2 with probability 1/2, which would only guarantees (expected) utility $-1/2$.

2.5. Transitive Strength

Definition 2. *Player* A_i *is* weaker *than its teammate* A_j*, denoted by* $A_i \leq A_j$*, if for any opponent* B_k*, the probability of "*A_i *wins against* B_k*" is less than or equal to the probability of "*A_j *wins against* B_k*." Similar for Team 2. Team 1* $\{A_1, \ldots, A_m\}$ *are* transitive *if there is a permutation* π *of* $1, \ldots, m$*, such that* $A_{\pi(1)} \leq \ldots \leq A_{\pi(m)}$*. Similar for Team 2.*

Definition 3. *A utility function* U *is* monotone *if* $U(t+1) \geq U(t)$ *for* $t \in 0, \ldots T-1$.

Theorem 2. *Assume monotone utility function. Then, (1) If* $A_m \leq \ldots \leq A_1$*, Team 1 has a SPE strategy which only selects, in each round, one of the players in* $A_1, \ldots, A_T$*. (2) Symmetrically, if* $B_n \leq \ldots \leq B_1$*, Team 2 has a SPE strategy which only selects, in each round, one of the players in* $B_1, \ldots, B_T$.

By combining Theorem 1 and Theorem 2, we can immediately get the following

Corollary 1. *When players in each team are transitive and* U *is monotone, there is a simple SPE strategy for both teams as follows. Assume that* $A_m \leq \ldots \leq A_1$ *and* $B_n \leq \ldots \leq B_1$*. Then, the SPE strategy for Team 1 is to select an unused player in* $A_1, \ldots, A_T$ *uniformly random in each round; a SPE strategy for Team 2 is to select an unused player in* $B_1, \ldots, B_T$ *uniformly random in each round.*

Theorem 2 and Corollary 1 have many applications. In the real word, the utility function is monotone, and in many situations, such as in board or sport games, it is indeed the case that the players are transitive.

We prove Theorem 2 (1) in the next; the claim (2) is symmetric.

We first provide two basic terminologies which are necessary for understanding the subsequent proof. Suppose that $A_m \leq \ldots \leq A_1$ and that A' is a subset of A and $A' = (A_{i[1]}, \ldots A_{i[|A'|]})$, where $i[1] < \ldots < i[|A'|]$. Then, for any $0 \leq C \leq |A'|$, the top C players of A' refers to $\{A_{i[1]}, \ldots, A_{i[C]}\}$, and the rank C player of A' refers to $A_{i[C]}$.

Lemma 3. *Suppose that* $U()$ *is monotone.*

1. *Consider a pair of history classes* $H_1 = (k, X_1, Y, w_1)$ *and* $H_2 = (k, X_2, Y, w_2)$*. We claim that, if the top* $T-k$ *players of* $A - X_1$ *and the top* $T-k$ *players of* $A - X_2$ *are the same and* $w_1 \geq w_2$*, then* $V(H_1) \geq V(H_2)$.
2. *Let* $H = (k, X, Y, w)$ *be a non-terminal history class. Let* A_u *be the rank* $T-k$ *player in* $A - X$ *and* A_v *be any player in* $A - X$ *that is not a top* $T-k$ *player. Then, the row in* $M(k, X, Y, w)$ *that corresponds to* A_u *dominates the row that corresponds to* A_v*. As a result, there is an equilibrium strategy at history* H *(for Team 1) which only selects the top* $T-k$ *unmatched players to play.*

Proof: We prove it by backward induction. When $k = T$, Claim 1 holds according to the monotone property of $U()$; and Claim 2 naturally holds since it is a terminal history.

Now, we argue that, for $0 \leq k < T$, if the lemma holds for $k+1$, it also holds for k.

First, we prove Claim 2. Let us compare the two rows corresponding to A_u and A_v. Let us fix a column, say the one corresponding to B_r. The cell corresponding to (A_u, B_r) is

$$M[u,r] = \underbrace{V(k+1, X+\{A_u\}, Y+\{B_r\}, w+1)}_{a} \cdot P_{u,r} + \underbrace{V(k+1, X+\{A_u\}, Y+\{B_r\}, w)}_{b} \cdot (1 - P_{u,r})$$

The cell corresponding to (A_v, B_r) is

$$M[v,r] = \underbrace{V(k+1, X+\{A_v\}, Y+\{B_r\}, w+1)}_{a'} \cdot P_{v,r} + \underbrace{V(k+1, X+\{A_v\}, Y+\{B_r\}, w)}_{b'} \cdot (1 - P_{v,r})$$

Notice that the top $T-k-1$ players in $A - X - \{A_u\}$ and $A - X - \{A_v\}$ are the same. So, from the induction hypothesis, $a' \geq a \geq a' \geq b \geq b' \geq b$, i.e., $a = a' \geq b = b'$.

Since that A_u is the top $T-k$ player while A_v is not, player A_v is weaker than A_u, which means that $P_{u,r} \geq P_{v,r}$.

Combining the above arguments, we get that

$$M[u,r] - M[v,r] = (a-b) \cdot (P_{u,r} - P_{v,r}) \geq 0.$$

Therefore, $M[u,r] \geq M[v,r]$, and thus Claim 2 holds.

Then, we prove Claim 1. Let M_1 denote $M(k, X_1, Y, w_1)$ and M_2 denote $M(k, X_2, Y, w_2)$ for short. Suppose that A_u is a top $T-k$ player in $A - X_1$ (which is also a top $T-k$ player in $A - X_2$) and that B_r is any player in $B - Y$.

We know

$$\begin{aligned} M_1[u,r] =& V(k+1, X_1+\{A_u\}, Y+\{B_r\}, w_1+1) \cdot P_{u,r} + \\ & V(k+1, X_1+\{A_u\}, Y+\{B_r\}, w_1) \cdot (1 - P_{u,r}) \\ M_2[u,r] =& V(k+1, X_2+\{A_u\}, Y+\{B_r\}, w_2+1) \cdot P_{u,r} + \\ & V(k+1, X_2+\{A_u\}, Y+\{B_r\}, w_2) \cdot (1 - P_{u,r}) \end{aligned}$$

By induction hypothesis, it follows that $M_1[u,r] \geq M_2[u,r]$.

Now, let σ denote the equilibrium strategy at H_2 that only selects the top $T-k$ unmatched players to play (Such a strategy exists according to Claim 2). Note that σ is also a legal strategy at

H_1. Let $\mu(H_1, \sigma)$ and $\mu(H_2, \sigma)$, respectively denote the utility of Team 1 when it applies strategy σ on H_1 and H_2. Then,

$$\mu(H_1, \sigma) = \min_{r : B_r \in Y} \sum_{u : A_u \in A - X_1} \sigma(A_u) \cdot M_1(u, r),$$
$$\mu(H_2, \sigma) = \min_{r : B_r \in Y} \sum_{u : A_u \in A - X_2} \sigma(A_u) \cdot M_2(u, r).$$

From the inequality $M_1(u, r) \geq M_2(u, r)$, we get $\mu(H_1, \sigma) \geq \mu(H_2, \sigma)$. Moreover, we also have $V(H_1) \geq \mu(H_1, \sigma)$ and $V(H_2) = \mu(H_2, \sigma)$ (the equality is since that σ is the equilibrium strategy on H_2). Together, $V(H_1) \geq V(H_2)$. □

Finally, Claim 2 of Lemma 3 implies Theorem 2 (1).

2.6. Non-transitive Strength

Definition 4. *A player is said to be* weakest*, if it is weaker than all its teammates; and is said to be* dominated*, if it has 0 probability to win against any player in the opponent team.*

Assume that the utility function is monotone. In the previous section, we show that if there are redundant players in Team 1 and if the strength of players of Team 1 are transitive, then there is a SPE strategy for Team 1 which does not select the weakest player. In other words, Team 1 can abandon the weakest one without decreasing its utility. In this section, we show that the transitivity is essential for this to hold. We start by the following claim.

Claim 2. *Suppose that Team 1 has redundant players, and some player A_u in Team 1 is weaker than all its teammate, and yet the players in Team 1 are not transitive. Then, Team 1 might decrease its utility by abandoning A_u.*

This is somewhat counterintuitive; it might be intuitive that the weakest player has no chance to participate in any match since one can always replace him by a better teammate and increase utility.

Perhaps even more surprisingly, we have the following claim:

Claim 3. *Suppose that Team 1 has redundant players, and some player A_u in Team 1 is dominated by the other team (i.e., has no chance to win at all), and the players in Team 1 are not transitive. Then, Team 1 might decrease its utility by abandoning the dominated player A_u.*

The above claims confirm that, the weakest player or even dominated player could help its team.

We would now like to state the organization of the remainder of the section. In subsection 2.6.1, we give examples that verify Claim 3, and we briefly explain the reason why we need dominated players. In subsection 2.6.2, we identify a special case where the weakest player can be abandoned without changing the utility. In subsection 2.6.3, we consider the optimal number of dominated players that we may need to achieve maximum utility. In subsection 2.6.4, we discuss the limitations of the dominated players.

2.6.1. Dominated Teammates Can Be Helpful

Let $V(T, P, U)$ denote the value of game $G(T, P, U)$. Let P^* denote the sub-matrix of P by deleting the last row (thus $G(T, P^*, U)$ is the game where Team 1 has abandoned A_m).

Example 2. *Let $n = m = 3, T = 2$, $U = U_E$ (recall that $U_E(t) = t - T/2$). $P = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$.*

In Example 2, there are redundant players and the players in each team are not transitive. Besides, A_3 is a dominated player. We argue the follows: (I) If A_3 is abandoned, Team 2 can win both rounds and hence $V(T, P^*, U) = -1$. (II) If A_3 is in the team, Team 2 cannot win both rounds with certainty and that means $V(T, P, U) > -1$. Combining (I) and (II), we get $V(T, P, U) > V(T, P^*, U)$, which implies Claim 3.

Proof of (I): If A_3 is abandoned, Team 2 can play as follows. It chooses B_3 to win the first round. If B_3 defeated A_1, it chooses B_1 in the second round to beat A_2; otherwise, it chooses B_2 in the second round to beat A_1. □

Proof of (II): If Team 2 wants to win with certainty in both rounds, it must select B_3 to play the first round. However, if Team 1 selects the dominated player A_3 to play the first round, Team 2 cannot win the second round with certainty anymore. □

From this example, we see why a dominated player might be helpful for its team. The reason behind is similar to the horse race story described at the beginning of Tang et al. (2010).

In the next, we give one more example. It gives, to our best knowledge, the largest decrease of the value of the game by abandoning a dominated player.

Example 3. *Let $m = 4, n = T = 3$. Let $U = U_E$ or $U = U_M$. Let $P = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$.*

According the method shown in subsection 2.3, we can compute that[4]

$$V(T, P, U_M) = 0;\ V(T, P^*, U_M) = -2/3;$$
$$V(T, P, U_E) = -1/2;\ V(T, P^*, U_E) = -1/2.$$

So, for the game $G(T, P, U_M)$, we will lose utility as much as 2/3 if we abandon the dominated player.

In the following we explicitly state a SPE strategy for Team 1. In the first round it selects the dominated player A_4. Without loss of generality, assume that it loses to B_1. In the second round, it selects A_2, A_3 uniformly random. So, there is 1/2 chance that Team 1 wins this round. Furthermore, if Team 1 wins the second round (say A_2 beats B_2) it can also wins the next (let A_3 beat B_3)

[4]The value of $G(T, P^*, U_M)$ and $G(T, P^*, U_M)$ can be simply computed according to Theorem 1 since there are no redundant players in these games.

and thus gets utility 1. By this strategy, there is 1/2 chance to get utility 1 and 1/2 chance to get utility −1, so the expected utility is 0.

However, for the game (T, P, U_E), we do not lose any utility by abandoning the dominated player. This is not a coincidence. In fact, this example belongs to a special case where the weakest player can indeed be abandoned. We show this in the next theorem.

2.6.2. A Case Where the Weakest Player Can Be Abandoned

We have seen that the weakest redundant player is useless when the players are transitive (as proved in Theorem 2) but might be useful when the players are not transitive (as shown in the previous subsection). So, the next question is:

Question 1. *If the players are not transitive, in what cases can the weakest player be abandoned?*

We get the following result.

Theorem 3. *Suppose that Team 1 has redundant players but Team 2 does not. So, $m > T$ and $n = T$. Moreover, suppose that each player in $A_{T+1}, \ldots, A_m$ is weaker than each player in $A_1, \ldots, A_T$. Let $U = U_E$. Then, Team 1 can abandon all the players in $A_{T+1}, \ldots, A_m$ without losing its utility.*

Remark 1. *According to Example 3, the claim in Theorem 3 fails when $U = U_M$. As a comparison, by recruiting extra dominated players, a team can gain more utility when $U = U_M$, but cannot when $U = U_E$. This may suggest that U_E is more reasonable than U_M in team competition.*

The condition $m > n = T$ is important. If both team got redundant players, the claim in Theorem 3 fails.

We need the following lemma in proving Theorem 3. It is a technical statement of probability theory.

Lemma 4. *Assume that $n = T$ and A_i, B_j are any pair of players from the two teams. Let $Q^{\sigma}_{i,j}$ denote the probability that A_i meets B_j in the game when Team 2 applies the uniformly random strategy and Team 1 applies some strategy σ. Then, $Q^{\sigma}_{i,j} \leq 1/T$.*

Proof: We prove it by induction on n. The case $n = 1$ is trivial. Suppose that the lemma holds for $n - 1$, and we now argue that it also holds for n. Assume that by applying σ, Team 1 has probability p to selects A_i in the first round. Then, the probability that A_i meats B_j in the game is at most $p\frac{1}{n} + (1-p)(1-\frac{1}{n}) \cdot \frac{1}{n-1}$ (the term $\frac{1}{n-1}$ is due to the induction hypothesis). Therefore, $Q^{\sigma}_{i,j} \leq p\frac{1}{n} + (1-p)\frac{1}{n} = \frac{1}{n} = \frac{1}{T}$. □

Proof of Theorem 3: We call $A_{T+1}, \ldots, A_m$ the weak players. When the weak players are abandoned, there are T remaining players for each team. By Theorem 1, the uniformly random strategy is a SPE strategy for Team 2. To prove Theorem 3, the key idea is to show that even if Team 1 is allowed to select the weak player, it will not gain more utility if Team 2 keep using the uniformly random strategy. On the other hand, it is obvious that Team 2 can't gain more utility (when Team 1 is allowed to select more players). Therefore, the value of game does not change when the weak players are allowed to play.

First, we compute the utility U^* of Team 1 when this team abandons its weak players. As an application of Lemma 4, for any pair of two players A_i, B_j $(1 \leq i, j \leq T)$, they meet with a probability no more than $1/T$. It follows that this probability equals $1/T$, as the sum of all these $T \cdot T$ probabilities equals T. Because A_i meets B_j with probability $\frac{1}{T}$ and A_i wins B_j with probability $P_{i,j}$ when they meet, it follows that the number of rounds t that Team 1 wins equals $\sum_{j=1..T} \sum_{i=1..T} \frac{1}{T} P_{i,j}$ in expectation. Therefore,

$$U^* = (\sum_{j=1..T} \sum_{i=1..T} \frac{1}{T} P_{i,j}) - \frac{T}{2}.$$

We now state a formula of the utility U^{σ} of Team 1 when it does not abandon its weak players and it applies some strategy σ against the uniformly random strategy of Team 2. Let $Q^{\sigma}(i,j)$ be defined as Lemma 4. Similar as above, the number of rounds t that Team 1 wins equals $\sum_{j=1..T} \sum_{i=1..m} Q^{\sigma}_{i,j} P_{i,j}$ in expectation. Therefore,

$$U^{\sigma} = (\sum_{j=1..T} \sum_{i=1..m} Q^{\sigma}_{i,j} P_{i,j}) - \frac{T}{2}$$

We only need to prove that $U^{\sigma} \leq U^*$, and it reduces to showing that for any fixed j in $1..T$,

$$\sum_{i=1..m} P_{i,j} Q^{\sigma}_{i,j} \leq \sum_{i=1..T} \frac{1}{T} P_{i,j} \tag{2}$$

To prove (2), consider the following optimization problem:

$$\begin{cases} \texttt{Variables:} & x = (x_1, \ldots, x_m) \\ \texttt{Parameters:} & c = (c_1, \ldots, c_m) \\ \texttt{Guarantee:} & c_i \geq c_{i'} (\forall (i, i') \text{ such that } i \leq T < i') \\ \texttt{Constraint 1:} & 0 \leq x_i \leq \frac{1}{T} (\forall 1 \leq i \leq m) \\ \texttt{Constraint 2:} & \sum_{i=1}^{m} x_i = 1 \\ \texttt{Objective:} & \max f(x) = \sum_{i=1}^{m} c_i x_i \end{cases}$$

Clearly, $f(x)$ is maximized at x^*, where $x^*_i = \begin{cases} \frac{1}{T} & i \leq T \\ 0 & i > T \end{cases}$.

Noticing the following facts, we see that inequality (2) is just an application of the above problem.

$$\begin{array}{cc} Q^{\sigma}_{i,j} \leq \frac{1}{T} & \text{(Applying Lemma 4)} \\ \sum_{i=1}^{m} Q^{\sigma}_{i,j} = 1 & \text{(According to the definition)} \\ \forall i \leq T < i', P_{i,j} \geq P_{i',j} & \text{(Since } A_{i'} \text{ is weaker than } A_i) \end{array}$$

□

2.6.3. Optimal Number of Dominated Players

Here we study the power of dominated players in another direction. As we see in subsection 2.6.1, by abandoning a redundant dominated player, Team 1 may decrease its utility. In other words, Team 1 may increase its utility by recruiting more dominated players. Note that the utility of Team 1 will not decrease by recruiting more dominated players. However, it

is unclear that the utility will strictly increase by doing so. For example, recruiting T dominated players is the same as recruiting $T-1$ players — in any case, if any team uses T dominated players in a competition, it gets the lowest utility! So, a natural question is:

Question 2. *In order to maximize the expected utility of Team 1 (i.e., the value of the game), how many dominated players should we recruit at least? Is it possible that we need as many as $\Theta(T)$ such players?*

The theorem below answers this question.

Theorem 4.

1. *Suppose $U = U_E$. Recruiting $T-1$ dominated players can be better than $T-2$, but recruiting T dominated players is the same as $T-1$. So, to achieve optimal utility, one may require $T-1$ dominated players. This number is tight.*
2. *Suppose $U = U_M$. Recruiting $\lfloor T/2 \rfloor$ dominated players can be better than $\lfloor T/2 \rfloor - 1$, but recruiting $\lfloor T/2 \rfloor + 1$ dominated players cannot be better than $\lfloor T/2 \rfloor$. So, to achieve optimal utility, one may require $\lfloor T/2 \rfloor$ dominated players, and this number is tight as well.*

One direction in these claims are rather trivial; we should never use T dominated players when $U = U_E$ or $\lfloor T/2 \rfloor + 1$ players when $U = U_M$. To prove the other direction, we need to construct some examples in which recruiting $T-1$ (resp. $\lfloor T/2 \rfloor$) could be better than $T-2$ (resp. $\lfloor T/2 \rfloor - 1$) when $U = U_E$ (resp. $U = U_M$). To construct such examples, an intuition is that we should make the current players in Team 1 as weak as possible. Our construction is as follows:

Example 4. $T \geq 1, m = T, n = T + (T-1), U = U_E, P_{i,j} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$.

Example 5. $T \geq 1, m = T, n = T + \lfloor T/2 \rfloor, U = U_M, P_{i,j} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$.

The following claims together prove Theorem 4.

C1. In Example 4, if Team 1 only recruit $T-2$ dominated players, it can win no rounds and thus can get utility $-T/2$.
C2. In Example 4, if Team 1 recruit $T-1$ dominated players, it can win a positive number of rounds in expected and thus gain utility more than $-T/2$.
C3. In Example 5, if Team 1 only recruit $\lfloor T/2 \rfloor - 1$ dominated players, it will always lose at least $\lfloor T/2 \rfloor + 1$ rounds and thus can only get utility -1.
C4. In Example 5, if Team 1 recruit $\lfloor T/2 \rfloor$ dominated players, it can sometimes win at least $\lceil T/2 \rceil$ rounds and thus can gain utility more than -1.

Proof of C1: In this case Team 2 can win all the rounds by playing as follows: in the first $T-1$ rounds, it selects the players $B_{T+1}, \ldots, B_{2T-1}$ to play; and they all win. Then, since Team 1 only has $T-2$ dominated players, at least one player in $A_1, \ldots, A_T$ has already played, denote it by A_i. In the last round, Team 2 select B_i and it definitely wins. □

Proof of C3: In this case, by applying a strategy similar to C1, Team 2 can win all the first $\lfloor T/2 \rfloor + 1$ rounds.[5] □

Proof of C2: For convenience, we denote the $T-1$ dominated players by $A_{T+1}, \ldots, A_{2T-1}$. We argue that, if Team 1 applies the uniform random strategy (that is, select one unused player in $A_1, \ldots, A_{2T-1}$ uniformly random in each round), then, Team 2 has no strategy to win all rounds all the time. Suppose to the contrary that Team 2 can do it, it must select a player from $B_{T+1} \ldots, B_{2T-1}$ to play in the first round; otherwise there is a chance that it loses the first round. Note that, since Team 1 apply the uniform random strategy, there is a chance that Team 1 select a dominated player in the first round. If this happens, Team 2 must again select a player from $B_{T+1} \ldots, B_{2T-1}$ to play in the second round. Once again, Team 1 might still select a dominated player in the second round. By induction, there is chance that Team 1 select all the dominated players in the first $T-1$ rounds while Team 2 consumes all its $T-1$ invincible players in $B_{T+1} \ldots, B_{2T-1}$. Then, Team 2 cannot win with certainty in the last round. □

The claim C4 is the most non-trivial. To prove it we first state the following lemma.

Definition 5. *For integers a, b, C such that*

$$C \geq 1, 0 \leq a \leq \lceil C/2 \rceil, 0 \leq b \leq \lfloor C/2 \rfloor, \tag{3}$$

let $\Gamma^C_{a,b}$ denote the following instance of team competition:

$$m = n = (C - a) + (\lfloor C/2 \rfloor - b),$$
$$T = C - a - b, P_{i,j} = \begin{cases} 1 & i = j \leq C - a \\ 0 & \text{otherwise} \end{cases}.$$

The utility is as follows[6]: if Team 1 wins at least $\lceil C/2 \rceil - a$ rounds, it gets utility 1 and Team 2 gets -1; otherwise, Team 1 gets utility -1 and Team 2 gets 1.

Lemma 5. *For integers a, b, C satisfying condition (3), Team 1 can win utility larger than -1 in the game $\Gamma^C_{a,b}$.*

Proof: Consider three cases.

Case 1 $a = \lceil C/2 \rceil$. In this case, Team 1 always get utility 1, and so $\Gamma^C_{a,b}$ has value 1, which is larger than -1.

Case 2 $b = \lfloor C/2 \rfloor$. The game $\Gamma^C_{a,b}$ can be restated as follows.

- $m = n = C - a$, $T = \lceil C/2 \rceil - a$.
  Player A_i can only defeat B_i for i in $1..m$.
  Team 1 gets utility 1 if it wins all the rounds; and -1 otherwise.

We argue that the uniformly random strategy guarantees Team 1 an expected utility larger than -1. Equivalently speaking, by applying the uniformly random strategy, Team 1

[5] In this case Team 2 can actually win all the T rounds.

[6] Here the utility functions for two teams are not identical. However, since it is still a zero-sum game, SPE strategies for the teams exists as before. The requirement that the utility functions are identical is not necessary in our model.

has a chance to win all the rounds. The proof is as follows. In the first round, there is a positive chance that A_i meets B_i for some i. Then, in the second round, same thing happens with a positive chance. This could happen for each round. When these coincidences happen, Team 1 wins all the rounds.

Case 3 $a < \lceil C/2 \rceil$ and $b < \lfloor C/2 \rfloor$.

We use induction. Assume that $\Gamma^C_{a+1,b}$ and $\Gamma^C_{a,b+1}$ both have value larger than -1, we argue that so does $\Gamma^C_{a,b}$. The following facts follow from the definition of $\Gamma^C_{a,b}$.

Fact 1. If the two teams select A_i and B_i for $i \leq C - a$ in the first round, it becomes a sub-game that is equivalent to $\Gamma^C_{a+1,b}$.

Fact 2. If the two teams select A_i and B_i for $i > C - a$ in the first round, it becomes a sub-game that is equivalent to $\Gamma^C_{a,b+1}$.

Combining them with the induction hypothesis, we get

Fact 3. If the two teams select players under the same index, it becomes a sub-game whose value is larger than -1.

The value of $\Gamma^C_{a,b}$ is equal to the value of the matrix game M, where $M(i,j)$ indicate the value of the sub-game when Team 1 select A_i and Team 2 select B_j in the first round. Fact 3 implies that all the utilities on the diagonal of matrix M are larger than -1. So, by using uniformly random strategy over its players, Team 1 can win a utility larger than -1. Therefore, $\Gamma^C_{a,b}$ has value larger than -1. □

PROOF OF C4: Let G denote the revised game of Example 5, in which Team 1 has recruited $\lfloor T/2 \rfloor$ dominated players. We could observe that game G is almost the same as $\Gamma^T_{0,0}$. To be more specific, when T is odd, G is exactly $\Gamma^T_{0,0}$; when T is even, the parameters m, n, T, P in G and $\Gamma^T_{0,0}$ are the same; but the utility U is slightly different.

Suppose that the value of G is -1. Then, Team 2 has a strategy which guarantees a expected utility -1. It means that Team 2 has a strategy which can always win $\lfloor \frac{T}{2} \rfloor + 1$ or more rounds. When Team 2 applies this strategy, Team 1 can never win $\lceil \frac{T}{2} \rceil$ rounds. It further implies that the value of $\Gamma^T_{0,0}$ is also -1. However, this contradicts with Lemma 5. Therefore, the value of G must be larger than -1. □

2.6.4. Limitations of the Dominated Players

Although the presences of dominated players can affect the value of the game, we conjecture that it will not be too much. A question is then,

QUESTION 3. *By abandoning a dominated player, how much value might be lost in the worst case? In other words, how much extra (expected) utility can a team gain by recruiting dominated players?*

According to our simulations, we have the following conjecture that we cannot prove at the moment.

CONJECTURE 1. *If $U = U_M$, we can gain at most 2/3 extra (expected) utility (in other words, the value of the game increases by at most 2/3) by recruiting arbitrary number of dominated players. If $U = U_E$, we can gain at most 1 extra (expected) utility by recruiting arbitrary number of dominated players.*

2.6.5. Throwing a Match and Discarding a Player

Recall the card game between Alice and Bob in subsection 2.1. We shall point out that, recruiting a dominated player in this context can be thought of applying a cheating action, which is to *throw a match* by not placing any card in that round. In the mentioned card game, if Alice and Bob are not allowed to throw a match, Alice can get expected utility $-1/3$; if Alice is allowed to throw a match, she can get expected utility $1/3$. This can be computed according to the method shown in subsection 2.3. Therefore, throwing a match is profitable if permitted.

It may seem unnatural to let a team throw a match like this. The following alternative cheating action called *discarding*, which may seem more natural, is still profitable for the team.

Discarding is defined as follows. Alice (Team 1) is allowed to discard one of its cards and agrees to lose in that round; however the discarded card is never revealed to Bob (Team 2). By discarding, Alice do not gain one more card at hand, unlike the case of throwing a match.

However, if discarding is allowed, it may still be beneficial. We give an instance in which one may gain extra utility by discarding. Formally, we have the following result.

CLAIM 4. *For every integer $K > 0$, there exists a game G such that $V_K(G) > V_{K-1}(G)$, where $V_K(G)$ denotes the value of game G in which Team 1 is allowed to discard at most K players.*

EXAMPLE 6. $m = K+1, n = 2K+1, T = K+1, U = U_E, P_{i,j} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$.

The following claims together imply Claim 4.

C5 In Example 6, if Team 1 is only allowed to discard $K - 1$ times, it cannot win in any round.

C6 In Example 6, if Team 1 is allowed to discard K times, it can win some rounds in expectation.

PROOF OF C5: Suppose that Team 1 is only allowed to use discarding $K - 1$ times. Observer that, for any i in $1...m$, after A_i has played and revealed by Team 1, player B_i becomes invincible that he would win with certainty if he plays in the next rounds. Notice that there are K invincible players at beginning (which are $B_{K+2} \ldots B_{2K+1}$) and Team 1 has only $K - 1$ chances to hide a player by discarding. So, in any round, Team 2 has an invincible player at hand. Therefore, Team 2 can win all the rounds. □

PROOF OF C6: Consider the following strategy for Team 1. First, Team 1 randomly chooses an order of the players (say, each order with possibility $1/m!$), and then randomly chooses exactly K of its players so that these players will be discarded while playing. We argue that this strategy guarantees Team 1 to win positive rounds in expectation. It reduces to proving that no strategy of Team 2 can win all the rounds against this strategy. Suppose that Team 2 can do so. In the first round, it must select an invincible player (i.e., a player in $B_{T+1} \ldots B_{T+K}$). Otherwise, there is a

chance that it loses this round. And then, we know that there is a chance Team 1 discards in the first round. If this happens, Team 2 must also select an invincible player in the second round. Again, it is possible that Team 1 still discards in this round. Continuing this process we see that, it could happen that, in the first K rounds Team 2 use all its K invincible players, while Team 1 uses discarding K times. Then, Team 2 could lose in the $K+1$-th round. □

3. MATERIALS AND METHODS (FOR A VARIANT): TAKE ACTIONS IN TURN

We now turn to the aforementioned non-simultaneous variant where *Team 2 sends its player before Team 1 in each round*. As mentioned in section 1, our techniques for solving the original simultaneous variant easily extends to the non-simultaneous variant, and most of our results for the non-simultaneous variant are aligned with the results stated in the previous section; see the small difference in **Table 1**.

First, consider the easiest case where $n = m = T$. Recall that $\mathbb{S}$ denotes the set of all perfect matchings between the T players in Team 1 and the T players in Team 2 (defined in Lemma 2). A key observation is that no matter what Team 2 does, Team 1 can make sure that the matching result between the two teams equals the one in $\mathbb{S}$ that benefits Team 1 the most (for a given utility function U). Thus a simple SPE can be described easily based on this particular matching (However, we do not declare that there is always an efficient algorithm for computing this perfect matching. For example, we are not aware of any good algorithms for computing it when $U = U_M$. Yet there are efficient algorithms for $U = U_E$).

Theorem 5. *When both teams have no redundant players (i.e., $n = m = T$), then it is a SPE when Team 1 applies the strategy so that the matching result is the same as the one that benefits Team 1 the most.*

For the case of transitive strength, we have the following result which aligns with Theorem 2.

Theorem 6. *Assume monotone utility function. Then, (1) If $A_m \leq \ldots \leq A_1$, Team 1 has a SPE strategy which only selects, in each round, one of the players in $A_1, \ldots, A_T$. (2) If $B_n \leq \ldots \leq B_1$, Team 2 has a SPE strategy which only selects, in each round, one of the players in $B_1, \ldots, B_T$. (Be aware that (1) is not symmetric to (2) for the non-simultaneous variant as Team 2 is no longer symmetric to Team 1.)*

Recall the history classes below Lemma 1 and the terminologies introduced above Lemma 3. In addition, when $H = (k, X, Y, w)$ is a history class (as in the simultaneous case), let $H(\sigma)$ ($\sigma \in B - Y$) denote the history class (in the non-simultaneous case) indicating that Team 2 have sent player σ after arriving at H.

Proving Theorem 6 (1) reduces to proving the following lemma which is similar to Lemma 3.

Lemma 6. *1. Consider a pair of history classes $H_1 = (k, X_1, Y, w_1)$ and $H_2 = (k, X_2, Y, w_2)$, where the top $T-k$ players of $A - X_1$ and the top $T-k$ players of $A - X_2$ are the same. If $w_1 \geq w_2$, we have $V(H_1) \geq V(H_2)$.*

2. Consider a non-terminal history class $H(\sigma)$ where $H = (k, X, Y, w)$ and $k < T$. Let A_u be the rank $T-k$ player in $A-X$, and let A_v be any player in $A - X$ that is not a top $T-k$ player. Then, for Team 1, selecting A_u is at least as good as selecting A_v to play against σ at $H(\sigma)$.

Our proof of Lemma 6 is analogous to our proof of Lemma 3.

Proof: We prove it by backward induction. For $k = T$, claim 1 holds obviously and claim 2 holds naturally. Assume the lemma holds for $k+1$, we now prove that it also holds for k.

Proof of claim 1. Clearly, $V(H_1) = \min_{\sigma \in B-Y} V(H_1(\sigma))$ and $V(H_2) = \min_{\sigma \in B-Y} V(H_2(\sigma))$. Therefore, it reduces to proving that $V(H_1(\sigma)) \geq V(H_2(\sigma))$ for any σ that belongs to $B - Y$.

Applying claim 2 on $H_2(\sigma)$, we obtain that there exists a top $T-k$ player A_u in $A - X_2$ such that

$$V(H_2(\sigma)) = \underbrace{V\left((k+1, X_2 + \{A_u\}, Y + \{\sigma\}, w_1 + 1)\right)}_{a} \cdot P_{u,\sigma} + \underbrace{V\left((k+1, X_2 + \{A_u\}, Y + \{\sigma\}, w_1)\right)}_{b} \cdot (1 - P_{u,\sigma})$$

As the top $T-k$ players in $A - X_1, A - X_2$ are the same, A_u is also a player in $A - X_1$, and thus

$$V(H_1(\sigma)) \geq \underbrace{V\left((k+1, X_1 + \{A_u\}, Y + \{\sigma\}, w_1 + 1)\right)}_{a'} \cdot P_{u,\sigma} + \underbrace{V\left((k+1, X_1 + \{A_u\}, Y + \{\sigma\}, w_1)\right)}_{b'} \cdot (1 - P_{u,\sigma})$$

By the induction hypothesis, $a = a'$ and $b = b'$. Altogether, $V(H_1(\sigma)) \geq V(H_2(\sigma))$.

Proof of Claim 2. The utilities of selecting A_u and A_v at the history class $H(\sigma)$ are respectively

$$V = \underbrace{V\left((k+1, X + \{A_u\}, Y + \{\sigma\}, w + 1)\right)}_{a} \cdot P_{u,\sigma} + \underbrace{V\left((k+1, X + \{A_u\}, Y + \{\sigma\}, w)\right)}_{b} \cdot (1 - P_{u,\sigma})$$

$$V' = \underbrace{V\left((k+1, X + \{A_v\}, Y + \{\sigma\}, w + 1)\right)}_{a'} \cdot P_{v,\sigma} + \underbrace{V\left((k+1, X + \{A_v\}, Y + \{\sigma\}, w)\right)}_{b'} \cdot (1 - P_{v,\sigma})$$

Notice that $a = a'$ and $b = b'$ and $a \geq b$ according to the induction hypothesis. Therefore,

$$V - V' = a(P_{u,\sigma} - P_{v,\sigma}) + b(P_{v,\sigma} - P_{u,\sigma}) = (a-b)(P_{u,\sigma} - P_{v,\sigma}) \geq 0.$$

□

Briefly, (for the current round) it is by the definition that the player A_u with rank $T-k$ performs better than any player A_v

with rank bigger than $T-k$, and (for the remaining rounds) the set of the top $T-(k+1)$ players are the same regardless of who we choose between A_u and A_v. Thus, A_u dominates A_v.

Theorem 6 (2) can be proved by a similar argument; we only state a key lemma and omit its proof.

Lemma 7. *1. Consider a pair of history classes $H_1 = (k, X, Y_1, w_1)$ and $H_2 = (k, X, Y_2, w_2)$, where the top $T-k$ players of $B-Y_1$ and the top $T-k$ players of $B-Y_2$ are the same. If $w_1 \geq w_2$, we have $V(H_1) \geq V(H_2)$.*

2. Consider a non-terminal history class $H = (k, X, Y, w)$ (where $k < T$). Let B_u be the rank $T-k$ player in $B-Y$, and let B_v be any player in $B-Y$ that is not a top $T-k$ player. Then, for Team 2, selecting B_u is at least as good as selecting B_v in the subsequent $(k+1)$-th round.

We now move on to the more challenging case where the strength of the players is not transitive. For this case, our first theorem is analogous to Theorem 3 in that it demonstrates a special condition under which we can abandon some weak players.

Theorem 7. *Assume the utility function is monotone.*

1. *Suppose $m > T$ and each player in $A_{T+1}, \ldots, A_m$ is weaker than each player in $A_1, \ldots, A_T$. If $n = T$, Team 1 can abandon all the players in $A_{T+1}, \ldots, A_m$ without losing its utility. However, if $n > T$, abandoning these weaker players may decrease the utility of Team 1.*
2. *Suppose $n > T$ and each player in $B_{T+1}, \ldots, B_n$ is weaker than each player in $B_1, \ldots, B_T$. No matter m equals T or not, abandoning the players $B_{T+1}, \ldots, B_m$ may decrease the utility of Team 2.*

According to Theorem 7, when a team has redundant weaker players and its opponent team has no redundant players, whether the weaker players can be abandoned depends on which team takes action first.

Proof of Theorem 7: 1. First, assume $n = T$. Among all possible matching results between the m players in Team 1 and the T players of Team 2 that give Team 1 the highest (expected) utility, there exists a matching result s that matches $A_1, \ldots, A_T$ to the T players of Team 2 (because $A_{T+1}, \ldots, A_m$ are weaker than $A_1, \ldots, A_T$). Team 1 can gain the same utility (implied by s) even if $A_{T+1}, \ldots, A_m$ are abandoned.

If $n > T$ and $U \in \{U_E, U_m\}$, abandoning the weaker players may decrease the utility of Team 1. We prove this by constructing an example in the following (this is basically Example 2 yet U is more general).

Example 7. $m = n = 3, T = 2$. $U \in \{U_E, U_M\}$. $P = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$.

For this example, Team 1 can win exactly 1 round and will lose all rounds if A_3 is abandoned.

2. We give two examples to prove this claim (one for $m = T$ and the other for $m > T$).

Example 8. $m = 2, n = 3, T = 2$. $U \in \{U_E, U_M\}$. $P = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$.

In this example, Team 2 can win exactly 1 round and will lose all rounds if B_3 is abandoned.

Example 9. $m = 4, n = 5, T = 3$. $U = U_E$. $P = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}$.

In this example, Team 2 can win exactly 1 round and will lose all rounds if B_4, B_5 are abandoned. □

We now study the optimal number of dominated players. The following theorem is a counterpart of Theorem 4. It says that for $U = U_E$ we need $T-1$ in the worst case, and for $U = U_M$ we need $\lfloor T/2 \rfloor$ in the worst case. Interestingly, the same bounds hold for Team 1 and Team 2 and for the simultaneous case.

Theorem 8. *The following hold claims for Team 1 and Team 2.*

1. *Suppose $U = U_E$. Recruiting $T-1$ dominated players can be better than $T-2$, but recruiting T dominated players is the same as $T-1$. So, to achieve optimal utility, one may require $T-1$ dominated players. This number is tight.*
2. *Suppose $U = U_M$. Recruiting $\lfloor T/2 \rfloor$ dominated players can be better than $\lfloor T/2 \rfloor - 1$, but recruiting $\lfloor T/2 \rfloor + 1$ dominated players cannot be better than $\lfloor T/2 \rfloor$. So, to achieve optimal utility, one may require $\lfloor T/2 \rfloor$ dominated players, and this number is tight as well.*

Proof: The proof of the two claims on Team 1 is easy and is very similar to the proof of Theorem 4. Recall Example 4 and 5 in the proof of Theorem 4. It can be observed that for Example 4 (where $U = U_E$), Team 1 can win nothing when it recruits $T-2$ dominated players, and can win exactly one round when it recruits $T-1$ dominated players. This means that it needs $T-1$ dominated players to achieve the optimum utility (and more than $T-1$ dominated players is obviously not needed). For Example 5 (where $U = U_M$), Team 1 can win nothing when it recruits $\lfloor T/2 \rfloor - 1$ dominated players, and can win as many as $T - \lfloor T/2 \rfloor$ rounds when it recruits $\lfloor T/2 \rfloor$ dominated players. This means that it needs $\lfloor T/2 \rfloor$ dominated players to achieve the optimum utility (and more than $\lfloor T/2 \rfloor$ dominated players is clearly not needed).

The proof of the claims on Team 2 is also easy but have to use different examples (Note that these examples are not symmetric to the examples given in Example 4 and 5).

Example 10. $T \geq 1, m = T + T - 2, n = T, U = U_E, P_{i,j} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$.

Example 11. $T \geq 1, m = T + \lfloor T/2 \rfloor - 1, n = T, U = U_M, P_{i,j} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$.

For Example 10 (where $U = U_E$), Team 2 can win nothing when it recruits $T - 2$ dominated players, and can win exactly one round when it recruits $T - 1$ dominated players. Therefore it needs $T - 1$ dominated players to achieve the optimum utility (and more than $T - 1$ dominated players is obviously not needed).

For Example 11 (where $U = U_M$), Team 2 can win nothing when it recruits $\lfloor T/2 \rfloor - 1$ dominated players, and can win as many as $T - \lfloor T/2 \rfloor$ rounds when it recruits $\lfloor T/2 \rfloor$ dominated players. Therefore it needs $\lfloor T/2 \rfloor$ dominated players to achieve the optimum utility (and more than $\lfloor T/2 \rfloor$ is clearly not needed).

□

4. DISCUSSION

In this paper, we study a novel game-theoretic model of situations where two teams make sequential decisions about which of a set of exhaustible actions to select in each round. These actions can be interpreted as team members, cards in a hand, etc. This model has applications in solving the DMRTA problem we introduced at the beginning of this paper. We present a simple SPE for the case where there are no redundant players or the strength of players is transitive. For the other case, we exhibit evidence that the redundant dominated players cannot be easily discounted in their contribution to team performance, which may appear counterintuitive. We investigate the power of the dominated players in three directions: (1) When do they influence the value of the competition? (2) If additional dominated players can be recruited, how many should be required to attain the maximum utility? (3) How much utility might be lost at most if we abandon them? We obtain several non-trivial results that fully or partially answer these questions. We believe that our results are of particular interests to both designers and players of team competitions.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author/s.

AUTHOR CONTRIBUTIONS

KJ did the key analysis and experiments, wrote the paper, and generalized several preliminary results obtained during his discussions with the other authors. SC joined KJ in several important discussion, which helped KJ in obtaining many preliminary results. PT joined the discussion later, polished the paper, and especially helped in writing the introduction, including the related works. JP helped in finding some applications and joined after we made our conference version. All authors contributed to the article and approved the submitted version.

FUNDING

This work was supported in part by the National Natural Science Foundation of China under Grant 61972434, and the Fundamental Research Funds for the Central Universities, Sun Yat-sen University, under Grant 19LGPY292.

REFERENCES

Altman, A., Procaccia, A. D., and Tennenholtz, M. (2009). "Nonmanipulable selections from a tournament," in *IJCAI*, ed C. Boutilier (Pasadena, CA), 27–32.

Botelho, S., and Alami, R. (1999). "M+: a scheme for multi-robot cooperation through negotiated task allocation and achievement," in *Proceedings 1999 IEEE International Conference on Robotics and Automation* (Detroit, MI), Vol. 2, 1234–1239. doi: 10.1109/ROBOT.1999.772530

Dias, M. B., Zlot, R., Kalra, N., and Stentz, A. (2006). Market-based multirobot coordination: a survey and analysis. *Proc. IEEE* 94, 1257–1270. doi: 10.1109/JPROC.2006.876939

Gerkey, B., and Mataric, M. (2002). Sold! Auction methods for multirobot coordination. *IEEE Trans. Robot. Autom.* 18, 758–768. doi: 10.1109/TRA.2002.803462

Hwang, F. K. (1982). New concepts in seeding knockout tournaments. *Am. Math. Monthly* 89, 235–239. doi: 10.1080/00029890.1982.11995420

Jin, K., Tang, P., and Chen, S. (2016). "On the power of dominated players in team competitions," in *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems, AAMAS 2016* (Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems), 14–22.

Kleinberg, R. (2012). *Olympic Badminton Is Not Incentive Compatible, Turing's Invisible Hand*. Technical report, Cornell University.

Knuth, D. E. (1987). A random knockout tournament. *Am. Math. Monthly* 93, 127–129. doi: 10.1137/1029011

Kuhn, H. W. (1953). *Extensive Games and the Problem of Information*. Princeton University Press.

Lanctot, M., Lisý, V., and Winands, M. (2014). "Monte Carlo tree search in simultaneous move games with applications to goofspiel," in *Computer Games, Volume 408 of Communications in Computer and Information Science* (Beijing), 28–43. doi: 10.1007/978-3-319-05428-5_3

Osborne, M. J., and Rubinstein, A. (1994). *A Course in Game Theory*. MIT Press.

Procaccia, A. (2013). *Olympic Badminton Is Not Incentive Compatible-Revisited, Turing's Invisible Hand*. Technical report, Carnegie Mellon University.

Rosen, S. (1986). Prizes and incentives in elimination tournaments. *Am. Econ. Rev.* 76, 701–715. doi: 10.3386/w1668

Schwenk, A. J. (2000). What is the correct way to seed a knockout tournament? *Am. Math. Monthly* 107, 140–150. doi: 10.1080/00029890.2000.12005171

Tang, P., Shoham, Y., and Lin, F. (2009). "Team competition," in *Proceedings of AAMAS*, (Budapest).

Tang, P., Shoham, Y., and Lin, F. (2010). Designing competitions between teams of individuals. *Artif. Intell.* 174, 749–766. doi: 10.1016/j.artint.2010.04.025

Vu, T., Altman, A., and Shoham, Y. (2009). "On the complexity of schedule control problems for knockout tournaments," in *AAMAS '09* (Budapest), 225–232.

Wang, G., Yu, H., Xu, J., and Huang, S. (2004). "A multi-agent model based on market competition for task allocation: a game theory approach," in *IEEE International Conference on Networking, Sensing and Control, 2004* (Taipei), Vol. 1, 282–286. doi: 10.1109/ICNSC.2004.1297449

Wikipedia (2015a). *Goofspiel*. Technical report, Wikipedia.

Wikipedia (2015b). *Sun bin*. Technical report, Wikipedia.

Wu, H., and Shang, H. (2020). Potential game for dynamic task allocation in multi-agent system. *ISA Trans.* 102, 208–220. doi: 10.1016/j.isatra.2020.03.004

Zlot, R., and Stentz, A. (2006). Market-based multirobot coordination for complex tasks. *Int. J. Robot. Res.* 25, 73–101. doi: 10.1177/0278364906061160

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

The reviewer VC declared a shared affiliation, with no collaboration, with one of the authors SC to the handling editor at the time of the review.

frontiers in Neurorobotics

METHODS
published: 28 May 2021
doi: 10.3389/fnbot.2021.674322

Public Opinion Early Warning Agent Model: A Deep Learning Cascade Virality Prediction Model Based on Multi-Feature Fusion

*Liqun Gao, Yujia Liu, Hongwu Zhuang, Haiyang Wang, Bin Zhou and Aiping Li**

Software Engineering Center, College of Computer, National University of Defense Technology, ChangSha, China

With the rapid popularity of agent technology, a public opinion early warning agent has attracted wide attention. Furthermore, a deep learning model can make the agent more automatic and efficient. Therefore, for the agency of a public opinion early warning task, the deep learning model is very suitable for completing tasks such as popularity prediction or emergency outbreak. In this context, improving the ability to automatically analyze and predict the virality of information cascades is one of the tasks that deep learning model approaches address. However, most of the existing studies sought to address this task by analyzing cascade underlying network structure. Recent studies proposed cascade virality prediction for agnostic-networks (without network structure), but did not consider the fusion of more effective features. In this paper, we propose an innovative cascade virus prediction model named CasWarn. It can be quickly deployed in intelligent agents to effectively predict the virality of public opinion information for different industries. Inspired by the agnostic-network model, this model extracts the key features (independent of the underlying network structure) of an information cascade, including dissemination scale, emotional polarity ratio, and semantic evolution. We use two improved neural network frameworks to embed these features, and then apply the classification task to predict the cascade virality. We conduct comprehensive experiments on two large social network datasets. Furthermore, the experimental results prove that CasWarn can make timely and effective cascade virality predictions and verify that each feature model of CasWarn is beneficial to improve performance.

Keywords: agent system, deep learning, cascade virality prediction, feature fusion, classification

OPEN ACCESS

Edited by:
Zhaoquan Gu,
Guangzhou University, China

Reviewed by:
Xiaofeng Wang,
Jiangnan University, China
Xiangzhan Yu,
Harbin Institute of Technology, China

****Correspondence:***
Aiping Li
liaiping@nudt.edu.cn

Received: *01 March 2021*
Accepted: *12 April 2021*
Published: *28 May 2021*

Citation:
Gao L, Liu Y, Zhuang H, Wang H, Zhou B and Li A (2021) Public Opinion Early Warning Agent Model: A Deep Learning Cascade Virality Prediction Model Based on Multi-Feature Fusion. Front. Neurorobot. 15:674322. doi: 10.3389/fnbot.2021.674322

1. INTRODUCTION

Currently, the number of agents is increasing rapidly (ichocki et al., 2011), and smart agents are more efficient. With the advancement of artificial intelligence technology, more and more intelligent agents are being used in the industry. A deep learning model provides a potential solution for artificial intelligence, it is widely used in various agents fields (Westerlund, 2020). With the rapid development of the Internet, the growth rate of information in online social networks has become an evaluation indicator of public opinion. Some information in the network will become the source of viral dissemination, and this information will spread like a storm. Different industries need to monitor their own network public opinion, especially for government, enterprises, and

media industries. They pay attention to the impact of sudden public opinion on themselves. In other words, they want to know what information related to their own will become a viral cascade as early as possible (Tatar et al., 2014). Information cascade virality means that some information may be widely spread in the network in a short time. It may be organized, planned behavior, or the extension of controversial social emergencies (Kefato et al., 2018).

However, the cost of manually studying and judging information cascade virality is enormous. Automatically distinguishing early warnings through agents becomes a way to reduce labor costs in different industries. Designing efficient neural network algorithms to meet agents' needs to predict information cascade virality becomes the focus of research. Supposing this agent can use the features of significant differences and accurately warn when information becomes viral in the early stage. In that case, it plays a crucial role in the decision-making (blocking or guiding the dissemination) of the follow-up information dissemination. With the continuous progress of machine learning models, many advanced models apply information cascade scale prediction. These works use the network structure features of information dissemination and network nodes attribute features to establish a strong correlation, for example, the number of followers/followers of participating users, user connections and community structure, and user activity, etc. They mainly use machine learning models to predict the magnitude of the information forwarded at the future moment (Li et al., 2017, 2018; Wang et al., 2018). The emergence of deep learning technology improves manual feature selection in early work and obtains more high-dimensional space representation capabilities. However, in the field of cascade virality prediction, there are still two types of problems in previous works.

Firstly, a social network is usually scattered. Most of the previous virality predictions are based on many underlying social relationships (Subbian et al., 2017). It causes most models to rely on the underlying network features and uses users' network relationships to predict the cascade virality. However, it may be difficult to obtain such detailed network information in most cases. Besides, for different industries that only pay attention to their own information, it is not significant to obtain global social network relation data, such as the following relationship between users. It is not well-supported in terms of data volume and algorithm efficiency.

Secondly, network relational data are dynamic and complex. When applying deep learning models to solve network relational data, it usually requires more complicated information aggregation work (like GNN Zhou et al., 2018) to embed network nodes' representation, which requires a large number of model parameters. Simultaneously, in an information cascade process, new nodes will also cause new node embedding problems after joining the network, which brings about the problem of continuous training of new parameter models (Qiu et al., 2018). However, for many practical purposes, the timeliness of the virality of the information is more critical. If we can predict the virality earlier rather than later, such predictions are useful.

Some studies expect similar results with less feature information. They ignore the underlying network structure features (Zhao et al., 2015). Subbian et al. (2017) propose an agnostic network-based method to reduce the network structure information in the information cascade process. Kefato et al. (2018) apply deep learning models to agnostic-network virus prediction, use the number of forwarding in the time sequence process as the feature, and use the CNN model to predict whether the information will explode. However, the construction model lacks critical features strongly related to the information cascade.

To solve these problems, we propose a cascade virality prediction model based on deep learning, named CasWarn. First, we segment an information cascade with time slices and extract the cascaded features in different time slices, including dissemination scale, emotional polarity ratio, semantic evolution features, and use advanced models to vectorize these features. Next, we design a module with two neural network modules to aggregate these features. The first module uses a convolutional neural network to aggregate the relations between different features and uses asynchronous patterns to learn the potential relations of different time-series features. The second uses a variant of a recurrent neural network to learn the semantic evolution relations in the cascade process. Then, we use the gradient descent algorithm to train the classification model. The main contributions of our work are as follows:

(1) We design an intelligent agent model to predict the cascade virality of social network information, which can be applied to public opinion monitoring for different industries. We only need to monitor the dissemination content and time of information related to different industries and do not need to care about the user relationship involved in the dissemination. Using a relatively small amount of information, it can quickly and effectively predict social network information virality.

(2) We propose an improved deep learning model, CasWarn, for cascade virality prediction based on time series. CasWarn extracts the key features of information dissemination from the agnostic-network and fuses these features through a deep learning model, which makes it more suitable for cascade virality prediction tasks.

(3) We conduct extensive experiments on two public datasets, and our results prove that CasWarn outperforms the latest benchmarks in many agnostic-network cascade virality prediction tasks. Simultaneously, compared with the state-of-the-art knowable-network model, we have achieved comparable performance under the premise of a less information parameter.

The rest of this paper is structured as follows. In section 2, we briefly review related works. Section 3 gives the formal definition of an information cascade event and information cascade sequence and defines the problem of cascade virality prediction, while section 4 details the proposed CasWarn model. In section 5, we discuss the experimental evaluation of CasWarn against previous state-of-the-art baselines. We conclude our work in section 6.

2. RELATED WORK

With the continuous advancement of artificial intelligence technology, more and more deep learning models are deployed on agents to solve problems in different fields, such as visual recognition (Ruiz-del-Solar et al., 2018; Gu et al., 2021), behavior supervision (Quan et al., 2018; Ganesan et al., 2021; Jia et al., 2021), and artificial assistance (Xiao et al., 2020), etc. Deep learning is defined as a scientific field involving complex functions (for example, non-linear dynamics) to train a multi-layer neural network, embedding the data from the original, high-dimensional, multimodal state to the understandable state of the agent system (Goodfellow et al., 2016). Due to the flexibility and adaptability of neural networks, it is very suitable for agent systems (Ciresan et al., 2010), especially at the most active research frontier, to help researchers study agent systems' perception capabilities (Marsland, 2009).

In the work of social network public opinion supervision, intelligent auxiliary agents can help different industries perceive their own network evaluations, and prompt the industry to follow-up or block the information. Many models based on deep learning have emerged in this field, and most of the work is to predict the scale of the information cascade through the model (Tsur and Rappoport, 2012; Jenders et al., 2013; Cheng et al., 2014; Weng et al., 2014; Gao et al., 2015; Zhao et al., 2015; Li et al., 2017). They use large-scale cascade indicators for intelligent early warning, and the main focus is predicted performance indicators and timeliness indicators (early detection). In order to obtain the macro-level predictive value of information cascade in social networks in time, many works decided to use machine learning models and divide the scale of cascade into two types of tasks: One uses regression, like (Kefato et al., 2018; Zhu et al., 2018), to predict the potential scale of an information cascade, and the other uses the classification models, such as Zhao et al. (2015), to define the form of dissemination as viral or non-viral. Based on the machine learning models, most of the work focuses on the following information cascade features: (a) network topology (e.g., user relationship, first-order relationship network structure of user, etc.); (b) network node features (e.g., user features, discussion content, information sources, or key early dissemination participants, etc.); (c) temporal features (e.g., forwarding interval, etc.).

In an information cascade influencing factors, some studies suggest that user features play a crucial role in the information cascade. One of the most common features is the number of followers. As the representative of user influence, it means that key users affect the speed and timing of the future dissemination scale (Zaman et al., 2013). Those who have many followers, such as celebrities and news industries, are more likely to have a larger number of cascading effects than ordinary users because their information is more evident in the network (Suh et al., 2010; Bakshy et al., 2011; Jenders et al., 2013). However, a large-scale information cascade is not only generated by influential users, but also closely related to the content of the information, and it makes sense to study large-scale cascades generated by ordinary users rather than celebrities (Dow et al., 2013). Some studies confirm that the text semantic contained in the information cascade process is considered one of the internal driving forces and key factors leading to cascade virality (Dong et al., 2015, 2016). Moreover, semantic features have better performance (with higher content complexity) in the cascade of observation topics (such as hashtags). For example, breaking news, rumors/fake news, hotspots, controversies/special topics, etc., attract more attention than normal content (Yano and Smith, 2010; Yan et al., 2011). Simultaneously, a lot of work confirms that in the process of an information cascade, user emotions involved in dissemination are important factors affecting information dissemination (Stieglitz and Dang-Xuan, 2012; Chen et al., 2016; Yuan et al., 2016). Pfitzner et al. (2012) introduce the concept of emotional divergence, which combines the positive and negative points of emotion in a tweet, and can also predict the probability of a tweet post being forwarded. In general, tweets with high emotional diversity are more likely to be retweeted, which affects the spread of the information. Jenders et al. (2013) on the relationship between emotional divergence and retweet probability can also confirm the research results of Pfitzner et al. (2012), and the sentiment analysis task is significantly improved through deep learning models. Tian et al. (2020) introduce sentiment knowledge to enhance pre-training (SKEP), learning a unified emotional expression and achieving better performance. Some preliminary and reliable attempts to explore network agnostic methods proved that useful and timely predictions could be made only based on the information learned from the cascade itself without any other network structure information (Subbian et al., 2017).

As mentioned, recent models in this field apply a neural network framework like CNN (Kefato et al., 2018), RNN (Li et al., 2017), and GNN (Chen et al., 2019), etc. By using the above information cascade features, the neural network model's excellent feature extraction ability is efficient for agent systems. Moreover, the semantic information in vector space can be better obtained by embedding the semantic information in the representation, such as word2vec (Goldberg and Levy, 2014).

For the reasons above, we use the classification method of agnostic-network. The advantage is that it is relatively "cheap" to obtain information dissemination for different industries (no need for global relationship structure or to construct a complex relationship network). We use the temporal features of an information cascade and process relatively easy-to-obtain text semantics, emotions, and dissemination scale, which are widely regarded as key features. Furthermore, we construct a timing-based sequence based on these features and propose a neural network feature fusion method, which obtained performance results comparable to expensive operations (such as the knowable-network models).

3. MODEL FRAMEWORK

3.1. Problem Formulation

The problem of predicting the scale of each cascaded forwarding depends on the definition of virality. The most common and useful definition is its size (Cheng et al., 2014). For most practical purposes, it is much more difficult to predict the exact size

than to know whether the cascade will be larger than a certain threshold. The threshold can be set as a relative measure or an absolute measure. Relative measure is used when the cascade size is unknown and relative to the population observed in the latest data (if user engagement with the social network is changing). In our task, we need to detect new information according to the needs of different industries, so we use the method based on fixed thresholds to predict the cascade's virality according to the task's needs.

Definition 3.1.1 (Information Cascade Event) In the cascade virality prediction process of agnostic-network, we can regard the information cascade process as a event. It is constantly changing with the evolution of time. When we divide the event according to the timestamp, an information cascading event can be expressed as: $E = [e_{t_0}, e_{t_1} ... e_{t_n}]$, where e_{t_i} is the state of the cascade when the event occurs at time t_i, which is:

$$e_{t_i} = \{f_j^{t_i} : f_j^{t_i} \in R^{n \times m}, i, j, m, n \in N\} \tag{1}$$

where $f_j^{t_i}$ is the feature representation of the event at event t_i and j represents different feature types, which can be represented as normalized visible features. In this paper, we use three key features that cause cascade virality, semantic features, and local and emotional ratios. Section 3.2.1 introduces these features in detail.

Definition 3.1.2 (Information Cascade Sequence) When an event is split into a series of sub-events, we define $E(t_s, t_e) = [E : E \in (E_{t_s} ... E_{t_e})]$. For the sake of brevity, we simplify the writing of $E(t_e)$: $E(t_e) = E(t_0, t_e)$, which means that the sub-events all start from t_0. Through the above definition, we can get a cascade sequence of information as follows:

$$C = \{E(t_0), E(t_2), ...E(t_n)\}. \tag{2}$$

After that, we define two cascade sequences: C_{O_t} and C_{Δ_t}, we consider C_{O_t} to be an observable sequence of the event, that is $C_{O_t} = C(t_n)$, which represents the set of sub-events of an event from the start time t_0 to t_n, and $C_{\Delta_t} = C(t_n + \Delta_t)$ is considered to be an unobservable sequence.

Problem 1: Cascade Virality Prediction In order to better solve the problem of cascade virality prediction, according to the above definition, we can obtain an observable event subsequence C_{O_t}, which contains each event slice E_i and cascade features e_{t_i} within the time slice. We seek to predict the magnitude of the information cascade event; $C_{\Delta_t}^{Num} = |C_{\Delta_t}|$ is larger than the absolute threshold τ in the prediction time Δ_t. Specifically, given a cascade C_{O_t} and a absolute threshold $\tau \in N$, if $C_{\Delta}^{Num} \geq \tau$ then C_{O_t} is labeled as a viral cascade. That is, we need to quantify the activation probability of information virality after the time interval Δ_t, which is denoted as follows:

$$P_v = P(C_{\Delta_t}^{Num} \mid C_{O_t}). \tag{3}$$

Where P_v is the probability of whether the cascade is viral or not, and C_{O_t} is the cascade sequence containing features. Further, the cascade virality prediction can then be formulated as, given C_{O_t}, Δ_t, finding an optimal mapping function $\mathcal{L}$ that minimizes the following objective with parameters Θ:

$$\mathcal{L}(\Theta) = -\sum_{i=1}^{N} \log P_\Theta(C_{\Delta_t}^{Num} \mid C_{O_t}; \Theta). \tag{4}$$

3.2. CasWarn Model

We hope to use the key features to predict the possibility of a viral cascade under the premise of the least available data and return the prediction results in the form of early warning to realize the early warning agent. Through the previous work of Subbian et al. (2017), we know that the viral cascade may begin to spread rapidly in the first few hours. In contrast, a non-viral cascade takes a long time to reach a small number of users.

Unlike the previous network-agnostic cascade virality prediction models, we consider that more features can be used, but the challenge is fusing and embedding features. In addition to statistical features such as the dissemination time, dissemination scale, etc., many useful features can be obtained, such as text and tweets' emotional features. We know that these features significantly impact the information cascade's virality and apply to agnostic networks from previous studies. Based on the above assumptions, we use deep learning models to model a time series-based multi-feature cascade prediction model. In summary, the model construction process is as follows:

1. For each information cascade process C in our dataset, we extract C_{t_n} in the observable time window, where t_n is the end of the timestamp we can observe.

2. We label the cascade C and determine whether it is viral or not by threshold τ at time to Δ_t.

3. Regarding whether the cascade sequence C will become viral, we segment the cascade based on a time window, and use different embedding methods to extract the dissemination scale, semantic evolution feature, and emotional polarity ratio features in different time slices, then use them as the input of the neural network model, and predict the probability as a classification task.

3.2.1. Data Preprocessing

We slice the observable cascade sequence C based on time series and sample different features with equal time windows. In the time slice, we extract the cascaded features through the following three steps:

a) Dissemination scale feature

Dissemination scale feature is a crucial indicator to determine the virality of the cascade. We process it into a sequence by extracting the number of forwardings in the time slice, as shown in **Figure 1a–2**. Specifically, by extracting the forwarding times in the time slice, similar to Kefato et al. (2018), the sequence of integers representing the number of events included in each slice becomes:

$$C^{dpf} = [|C_{O_{t_i}}| : 0 \leq t_i < t_o] \tag{5}$$

where $C_{O_{t_i}}$ represents the total number of reposts at time t_i, t_o represents the sequence observation time, and i represents the $i - th$ time slice.

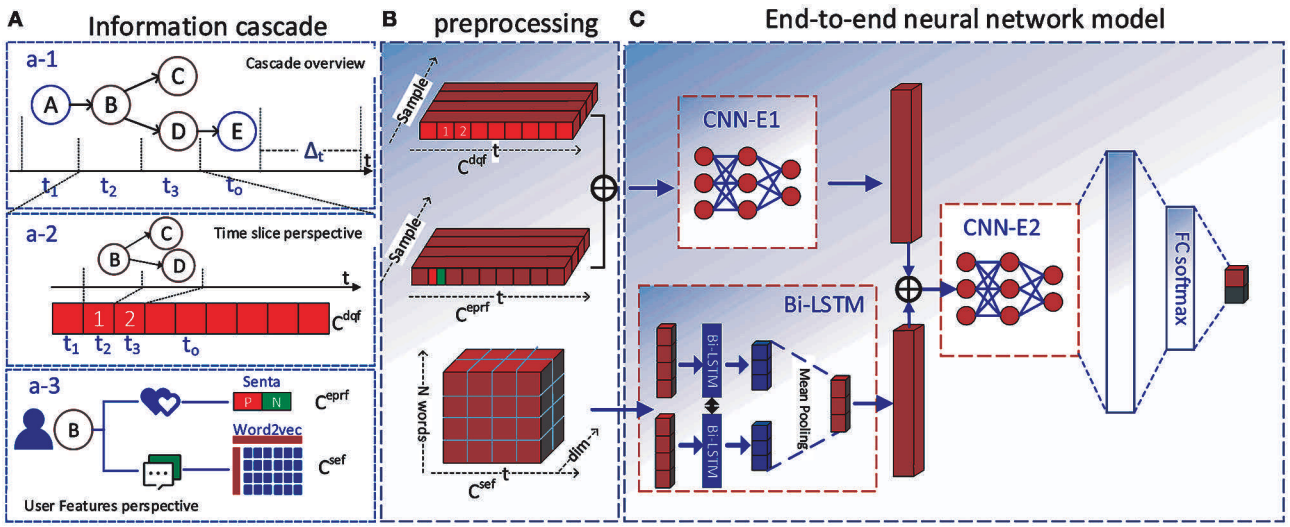


FIGURE 1 | The overall architecture of CasWarn. **(A)**: Three perspectives of information cascade. (a-1): The overall information cascade after time slice; (a-2): the composition form of dissemination scale feature after time slice; (a-3): features in the user's view, including emotional polarity ratio and semantic evolution features. **(B)**: Different feature preprocessing and embedding representation processes. **(C)**: The end-to-end neural network model. It first fuses the quantitative and emotional features through the CNN-E1 layer and then embeds the temporal semantic evolution features through the Bi-LSTM model, it next uses the CNN-E2 model to fuse the three features again. Finally, the FC-softmax layer predicts the result.

b) Emotional polarity ratio feature We calculated the emotional polarity of users participating in the information cascade for forwarding and commenting in the time slice, as shown in **Figure 1a–3**. Specifically, we use the Senta model, which can obtain the comments' emotional polarity to the original tweets (Tian et al., 2020). Then we construct the emotional ratio in a time window. This ratio is a two-tuple, indicating the degree of opposition of emotions in the time window:

$$C^{eprf} = [(sum(p_s^{t_i}) : sum(n_s^{t_i}), p_s = 1, n_s = 0, 0 \leq t_i < t_o] \quad (6)$$

where p_s is positive emotion and n_s is negative emotion.

c) Semantic evolution feature

For viral cascades, the evolution of topic semantics is more likely to cause the "mutation." By transforming topics in different time slices, we want to capture the impact of the features of topic evolution on the spread of the virus. We use the word2vec method to vectorize the semantic information of high-frequency topics in the time window, as shown in **Figure 1a–3**. Specifically, we extracted $topic_n$ words with the highest word frequency in each time window to represent the key semantic features of this time window and form a matrix sequence based on the time series:

$$\begin{aligned} C^{sem} &= [X^{t_i} : X \in R^{d \times n}, 0 \leq t_i < t_o] \\ X^{t_i} &= [T_i^t : 0 \leq i < topic_n, T \in W] \end{aligned} \quad (7)$$

where X is the subject word in the time slice, T is the keyword, and W is the corpus. It is worth noting that if the number of samples in a specific time slice is less than $topic_n$, we perform a zero-padding operation.

3.2.2. Neural Network Model

After obtaining the three features, we design an end-to-end neural network model (**Figure 1C**) to predict the cascade virality. We use the CNN as the main framework to solve the prediction task. Because in the time series classification task, a CNN has the advantages of high efficiency and high performance (Gundersen et al., 2020), we changed the frame structure of the original CNN to better adapt to the task needs.

As shown in **Figure 1C**, we use two layers of CNN convolutional layers in the neural network model to obtain the feature representation after feature fusion. In particular, we have designed an improved two-way method for the representation of semantic features in the time window. The Bi-LSTM layer (Yulita et al., 2017) learns the potential connections in the topic evolution process and finally uses the fully connected layer to predict the results.

3.2.2.1. Input Layer

As illustrated in **Figure 1B**, the input layer constructs a feature vector for three types of features. We can see from **Figure 1C** that these features are synchronized in the time series, but when input as a model, the steps are asynchronous.

3.2.2.2. CNN-E1 Feature Fusion Layer

As shown in the CNN-E1 part of **Figure 1C**, we concatenate C^{dqf} and C^{eprf} as a vector into the convolution layer of the CNN model, as:

$$h_c = [h_{c^{dqf}} || h_{c^{eprf}}]. \quad (8)$$

It is worth noting that we want to obtain the implied features of filters with different sliding windows, so we apply n filters of

different sizes on every possible slice of the fused features:

$$f_l(h_c^i) = \sigma(\mathcal{W}^i h_c^i + b^i) \tag{9}$$

where h_c^i can be regarded as a filter with step size i, $i \in (1, slice - 1)$, σ is the activation function, and we use the relu method. Similar to the n-grams method, we can regard W^i as the weight of the filter layer of i^{th}, and $h_c^i \in R^{kd}$ is the feature representation with the async length i. We sample the dependencies between subsequences by obtaining different convolutional layers. Then we obtain the fused feature representation by summation average:

$$f_1(h_c) = \frac{1}{i}\sum_{i \in n} f_l(h_c^i) \tag{10}$$

where, $f_1(h_c) \in R^{d1}$, and n represents the number of filters.

3.2.2.3. Bi-LSTM Semantic Evolution Layer

For the feature representation of semantic evolution, we design an improved architecture based on bi-directional LSTM (Bi-LSTM) to obtain the potential relationship of topic evolution in different time slices. Unlike the fully connected layer, it can capture the potential relations of semantic changes, as shown in the Bi-LSTM layer in **Figure 1C**. Firstly, since the number of keywords extracted in each time interval is fixed, we first concatenate the semantic information which represents the semantic feature vector in each time slice:

$$h_{c^{sef}}^t = ||^{i \in topic_n} h_i^t \tag{11}$$

where h_i^t represents the representation of the i-th feature word under the t-th time slice. Then, for different time slices, the content embedding of $h_{c^{sef}}$ is computed as follows:

$$f_2(h_{c^{sef}}^t) = \frac{\sum_{t \in slice}[\overrightarrow{LSTM}\{\mathcal{W}(h_{c^{sef}}^t)\}||\overleftarrow{LSTM}\{\mathcal{W}(h_{c^{sef}}^t)\}]}{|slice|} \tag{12}$$

where $f_2(h_{c^{sef}}) \in R^{d \times 1}$ (d: content embedding dimension), *slice* represents the number of time slices, and $\mathcal{W}$ represents the learning parameters of neural networks. The operator || denotes concatenation. We use the Bi-LSTM model to learn the potential relationship of semantic evolution. The LSTM is formulated as:

$$\begin{aligned}
\mathbf{z}_i &= \sigma\left(\mathcal{U}_z\left(h_{c^{sef}}^t\right) + \mathcal{W}_z\mathbf{h}_{i-1} + \mathbf{b}_z\right)\\
\mathbf{f}_i &= \sigma\left(\mathcal{U}_f\left(h_{c^{sef}}^t\right) + \mathcal{W}_f\mathbf{h}_{i-1} + \mathbf{b}_f\right)\\
\mathbf{o}_i &= \sigma\left(\mathcal{U}_o\left(h_{c^{sef}}^t\right) + \mathcal{W}_o\mathbf{h}_{i-1} + \mathbf{b}_o\right)\\
\hat{\mathbf{c}}_i &= \tanh\left(\mathcal{U}_c\left(h_{c^{sef}}^t\right) + \mathcal{W}_c\mathbf{h}_{i-1} + \mathbf{b}_c\right)\\
\mathbf{c}_i &= \mathbf{f}_i \circ \mathbf{c}_{i-1} + \mathbf{z}_i \circ \hat{\mathbf{c}}_i\\
\mathbf{h}_i &= \tanh(\mathbf{c}_i) \circ \mathbf{o}_i
\end{aligned} \tag{13}$$

where $\mathbf{h}_i \in \mathbb{R}^{(d/2)\times 1}$ is the output hidden state of i-th content, $\circ$ denotes the Hadamard product, $\mathcal{U}_j \in \mathbb{R}^{(d/2)\times d_f}$, $\mathcal{W}_j \in \mathbb{R}^{(d/2)\times d_f}$, $\mathbf{b}_j \in \mathbb{R}^{(d/2)\times 1}$, and $(j \in \{z, f, o, c\})$ are learnable parameters, z_i, f_i, and o_i are the forget gate vector, input gate vector, and output gate vector of the i-th semantic evolution feature, respectively. It is worth noting that the Bi-LSTM model can aggregate the ordered semantic information in order to obtain the implicit association of the semantic evolution process in different time slices.

3.2.2.4. CNN-E2 Feature Fusion Layer

Next, we concatenate the semantic evolution feature $f_2(h_{c^{sef}}^t)$ with the output feature $f_1(h_c)$ of the previous layer:

$$h_{des} = f_1(h_c)||f_2(h_{c^{sem}}). \tag{14}$$

As shown in the CNN-E2 layer in **Figure 1C**, the concatenated data are fused again by the CNN feature fusion layer to learn the potential relationship between different features:

$$f_3(h_{des}) = \frac{1}{i}\sum_{i \in n} \sigma(\mathcal{W}^i h_{des} + b^i). \tag{15}$$

Then, $f_3(h_{des})$ is followed by a fully connected (FC-softmax layer) logistic classification layer:

$$h(c_i) = softmax(\mathcal{W} f_3(h_{des}) + b^i). \tag{16}$$

The vector $h(c_i) \in \mathbb{R}^2$ can be regarded as the last feature representation in the model, which will be used to predict the virality of the cascade.

3.2.2.5. Output Layer and Loss Function

This layer outputs a two-dimensional representation vector for each information cascade. We compare the representation of the dissemination scale feature with the ground truth, and then optimize the log-likelihood loss, as follows:

$$\min \sum_i y_i \log\left(h(c_i)\right) + \left(1 - y_i\right)\log\left(1 - h(c_i)\right) \tag{17}$$

where $h(c_i)$ is the predicted value, y_i is ground truth, and the model parameters are trained using the back-propagation algorithm.

There are three main advantages for this framework: (1) It has concise structures with relatively low complexity (fewer parameters), making the model implementation and tuning relatively easy; (2) it can fuse the key features of information cascade in the agnostic-network and has a strong classification performance; (3) since the model is modular, it is flexible to add extra features, making the model extension more available.

4. EXPERIMENTS AND RESULTS

In this section, we conduct extensive experiments to answer the following research questions: **(RQ1)** How does CasWarn perform the virus cascade prediction task compared with the state-of-the-art baselines? **(RQ2)** How does CasWarn compare with most state-of-the-art baselines in terms of early detection capability? **(RQ3)** How do different features, such as emotional polarity or semantic evolution, affect the performance of

the model? **(RQ4)** How do various hyper-parameters, e.g., number of slices, word embedding dimension, impact the model performance?

4.1. Datasets

Our experiment selected two real-world social network datasets. One mostly exists to evaluate their methods of predicting diffusions on single social network data (Zhang et al., 2013, 2015; Cao et al., 2020). Another is a dataset of Twitter posts that we collected for specific semantics for different industries' needs to evaluate the proposed CasWarn model quantitatively.

4.1.1. Weibo Dataset

Sina Weibo is the most popular Chinese Weibo service. The dataset is from (Zhang et al., 2013) and can be downloaded here. The complete dataset contained 1.78 million users and 23 million tweets between September 28, 2012 and October 29, 2012. It is worth noting that 300,000 original tweets in this dataset become information cascades. We sample these cascades, as shown in **Table 1**.

4.1.2. Twitter Dataset

During the unrest in Hong Kong in 2019, the government was more concerned about whether Hong Kong-related tweets would become viral cascades. Unlike Sina Weibo, different industries' social public opinion events are more aggregated at a semantic level. Therefore, in order to better verify the impact of fusion features on the information cascade, we collected anonymized tweets related to Hong Kong from September to October 2019 for a total of 30 days by using the Twitter API, to better verify the authenticity of the cascade virality prediction of the content we care about.

As **Table 1** shows, we sample data with a cascade scale of more than 1,000 times ($\tau = 1,000$) as positive samples, negative samples are obtained by random sampling.

4.2. Comparison Methods

We compare CasWarn with a set of strong baselines, including feature-based models used for cascade prediction (Logistic Regression, SEISMIC), deep learning models based on cascade embedding in knowable networks (Deepcas), and the state-of-the-art deep learning models based on cascade embedding in agnostic networks (Cas2Vec).

Logistic Regression (LOR): This baseline is used in previous studies. We concatenate the vector of each time window and calculate it as the LOR method's input for training a classification model. It should be noted that each time window contains the stitching of three feature vectors.

SEISMIC: This is a recent study that uses point estimation models to predict the popularity of tweets (Zhao et al., 2015). It evaluates the influence of tweets based on the number of retweets at time t, then the estimated infectiousness is used to predict the ultimate size of the tweet. We follow a similar strategy as Cas2Vec to label tweets based on fixed size, that is viral if and only if fixed size is larger than τ.

DeepCas: This is the state-of-the-art deep representation learning model for knowable-network popularity prediction (Li et al., 2017), which learns the representation of cascade graphs in an end-to-end model. Specifically, it represents the cascade graph. DeepCas significantly improves the performance of hand-crafting feature-based methods. As a result, we here take DeepCas as a knowable-network method to compare with CasWarn. Specifically, we use formula (17) to modify the loss function of DeepCas from regression task to classification task.

Cas2Vec: Cas2Vec is the state-of-the-art deep representation learning model for network-agnostic cascade virality prediction (Kefato et al., 2018). This model applies convolutional neural networks to model the sequence of retweet size within the time window and predicts information cascade virality.

4.3. Evaluation Settings and Implementation Details

To evaluate our algorithm, we use the following settings: As required by problem 1 (section 3.1), we want to predict the task based on the observable time t_o and the forecast time window Δ_t. Since the distribution of viral cascades is highly skewed and sparse, we set the ratio of positive and negative samples to 1:2 and use validation sets during training to adjust hyper-parameters, such as the size of filters.

When the hyper-parameters are fixed, we use three-fold cross-validation without the validation set, and record the average results and errors. Regarding the embedding of topic words, we use a fixed value $topic_n = 15$ and use F-score with $\beta = 3$ (because it is a rare classification prediction).

4.4. Results

4.4.1. Virus Cascade Prediction Performance (RQ1)

To answer RQ1, we design experiments to evaluate CasWarn on cascade virality prediction tasks. In this set of experiments, our goal is to evaluate the effective classification performance of our model under different baselines. We set the observable time $t_o = 1$ (1 h), and then evaluate F1 values under different $\Delta_t = t_o + \Delta$. The results are shown in the table.

Table 2 reports the performance of all models and shows the best results in bold. Comparing different baselines on the two datasets shows that the best baselines in most cases are our model. We can see that most of CasWarn's performance is better than the DeepCAS model based on the knowable-network. However, CasWarn is based on the agnostic-network, which requires less data and is more efficient. Compared with the agnostic-network model, the relative improvements of CasWarn over the Cas2Vec range from 1.3 to 2.3% and 1.5 to 6.0% for the Twitter and Weibo datasets. In general, the experimental results on the F1 value show that the deep learning framework we proposed is effective and demonstrates that it can outperform state-of-the-art baselines in the cascade virality prediction task of the agnostic-network.

4.4.2. Early Prediction (RQ2)

In order to solve this problem, we analyze the prediction experiment of early information dissemination. And observed that most events occurred twice in the median time of spread of all cascade viruses. Similar to the work of Kefato et al. (2018), we selected the median of the viral cascade of two datasets. The

TABLE 1 | Summary of dataset statistics.

Dataset	Original tweets	Retweets average	Positive samples	Positive retweets
Weibo	300,131	7.91	6,734	5,725,352
Twitter	470,435	4.43	9,723	7,786,556

TABLE 2 | Prediction performance of different models on the two datasets (%).

Model	Twitter						Weibo					
	1.5 h	2 h	5 h	10 h	20 h	60 h	1.5 h	2 h	5 h	10 h	20 h	60 h
LOR	83.23	83.13	82.47	80.35	76.32	77.39	80.57	81.64	79.87	77.56	72.23	69.14
SEISMIC	82.61	79.77	77.26	74.53	72.24	59.57	77.44	75.32	72.21	69.57	62.27	60.50
DeepCas	86.97	87.42	**87.90**	83.21	82.77	**79.83**	83.74	**84.42**	83.21	**80.22**	79.38	76.77
Cas2Vec	85.98	86.72	85.43	84.35	82.78	77.21	83.26	83.33	**84.23**	74.57	73.31	72.29
CasWarn	**87.77**	**88.97**	87.45	**86.23**	**84.33**	78.77	**84.98**	82.21	83.44	79.90	**79.62**	**77.23**

The bold values are represent the optimal performance values.

median of Weibo is 16 h and Twitter is 7 h (due to Twitter data being more focused on topics, and that it is easier to quickly forward various pieces of network information). We choose a different (but fixed) prediction time Δ_t for each dataset, that is, 16 h for Twitter and 34 h for Weibo, and then change the size of the prediction window from 1 h to Δ_t h (the step is fixed, 1 h) to evaluate the time.

Note that the forecast time is fixed. In both cases, the step is 1 h. The rest of the hyper-parameters are the same as RQ1, but we use the recall rate to evaluate the performance of early detection.

Figure 2 shows that, as expected, our model achieves the best recall rate at the minimum value (1 h), which shows that we predict that the virus cascade's performance is the best within 1 h. At the same time, we hope that the prediction of the algorithm is robust as Δ_t increases. We can see that the performance of baselines declines faster than CasWarn, and CasWarn gets the best recall. In addition to the previous cascade virality prediction, our experiments also show that CasWarn is more robust than state-of-the-art models and can predict cascade virality as early as possible.

4.4.3. Ablation Study (RQ3)

CasWarn is a framework for a deep learning early warning agent that fuses multiple cascade features. How different features impact the model performance and whether emotion polarity or semantic evolution aggregation effectively improves the model's predictive ability need to be addressed. To answer RQ3, we conducted an ablation study to evaluate the performance of several model variants, including: **(a)**: No-Sen which uses semantic evolution feature and dissemination scale feature encoding to represent each cascade sequence embedding (without emotional polarity feature). **(b)**: No-Sem which uses emotional polarity and dissemination scale feature encoding to represent each cascade sequence embedding (without semantic evolution feature), and **(c)**: Semantic-FC which employs a fully connected neural network to embed semantic evolution features.

In this group of experiments, we set the observation window as t0 = 0.5, 1, 3 (hours) and use the same parameter in RQ1 for our hyper-parameter. The results on two datasets are reported in **Figure 3**. From this figure, we can see that the performance of CasWarn is better than that of No-Sen in most cases, demonstrating that the cascade sequence embedded with the emotional polarity feature is more efficient for cascade virality prediction. Similarly, we can find that the performance of CasWarn is better than that of No-Sem, demonstrating that the fusion of the two cascade sequence features is effective in improving the performance of the model. Semantic-FC outperforms No-Sen, which shows that our improved Bi-LSTM-based semantic evolution feature embedding is better than the embedding methods used to capture "deep" content feature interactions such as the FC layer. It is worth noting that the two datasets show better results on the Twitter dataset due to the more aggregated semantic content of Twitter.

4.4.4. Hyper-Parameter Sensitivity (RQ4)

Hyper-parameters play an essential role in CasWarn because optimal parameters for cascade virality prediction determine the prediction results' accuracy. We conduct experiments to analyze the impacts of three key parameters, i.e., the number of slices (see section in the supplement for detailed setup) and the topic word's embedding dimension. Cascade virality predicts the performances of CasWarn as a function of the two datasets, which are shown in **Figure 4**.

For the number of slices parameter, from the upper part of **Figure 4**, we can see that performance improves as the number of slices increases and then drops at a peak. For the Twitter dataset, we can see that 40 to 50 is the best metric. In the Weibo dataset, the optimal solution is between 30 and 40. So we find that a value between 30 and 50 provides the best result. For the parameters of embedded dimensions of semantic evolution features, we set them to float before 16 to 256 dimensions. From the lower part of **Figure 4**, we can see that the performance is significantly

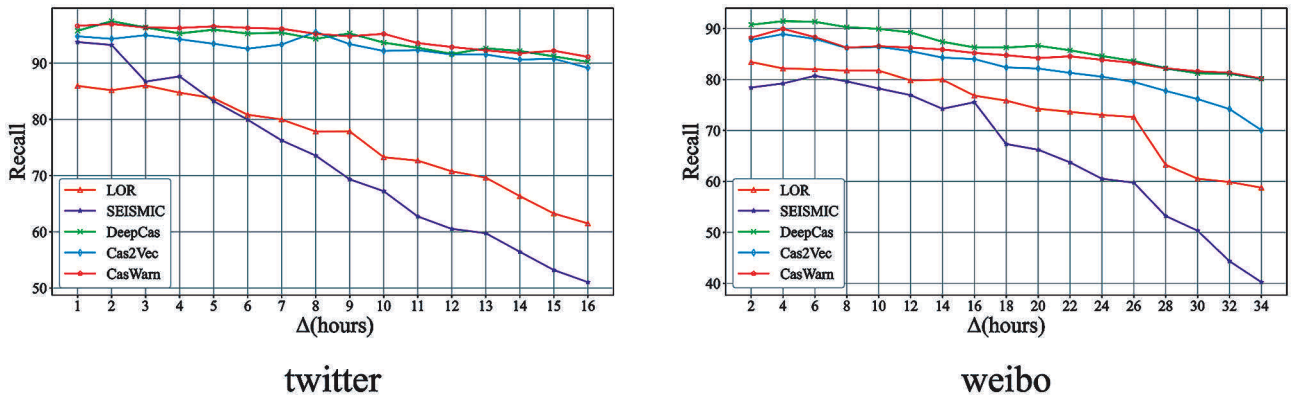


FIGURE 2 | Evaluation results of early prediction experiments for the Twitter and Weibo datasets.

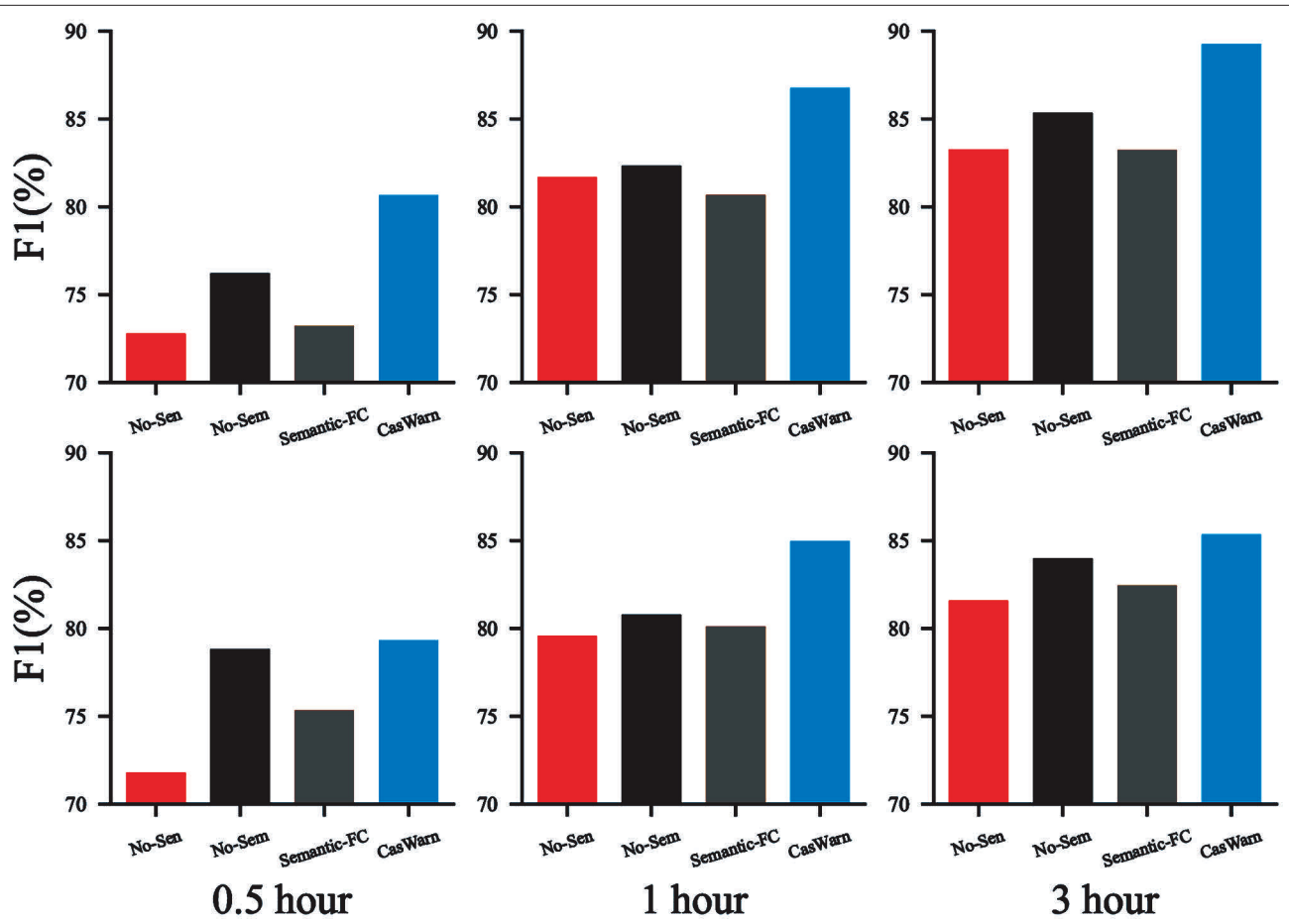


FIGURE 3 | Performances of variant proposed models.

improved with the increase of dimensions in the early stage. Then the performance has decreased with the increase of dimensions, which may be the result of over-fitting. The dimension when set to 32 meets the requirements of fewer variable parameters and timeliness.

5. CONCLUSIONS

In this paper, we propose a viral cascade early warning model, which can be deployed on intelligent agents to assist different industries in monitoring the public opinion effect of

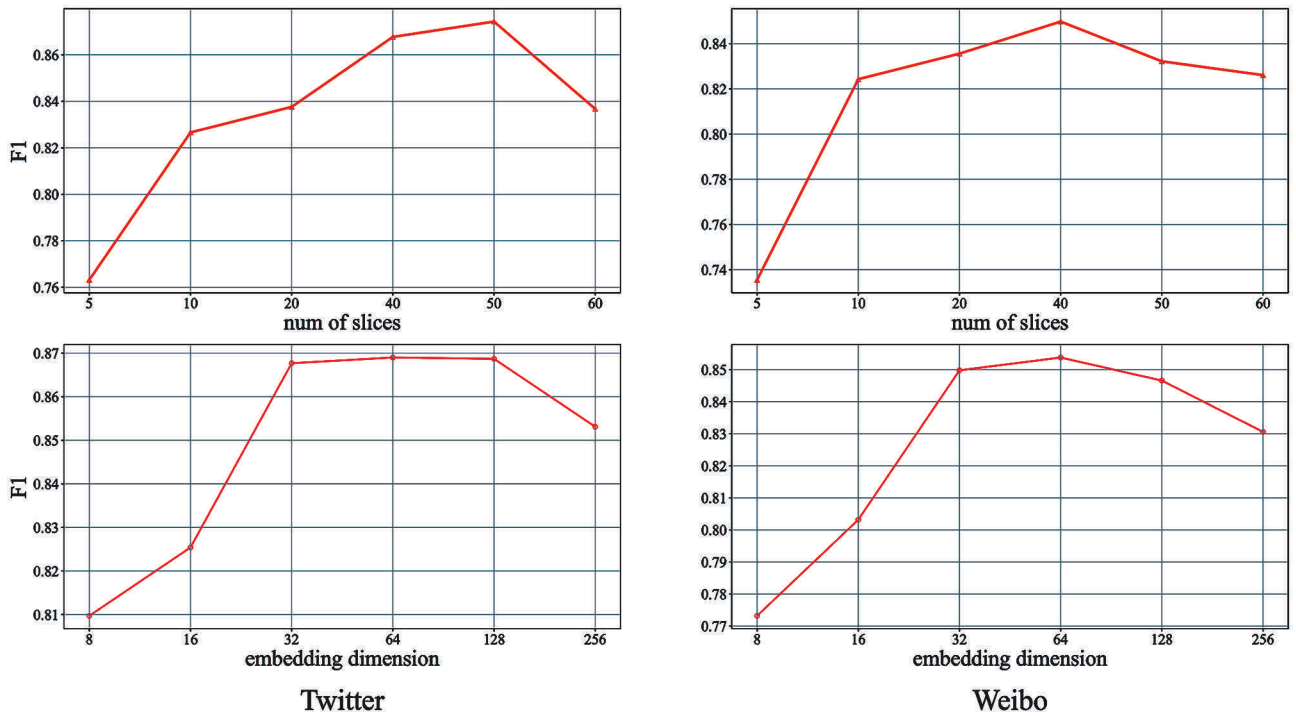


FIGURE 4 | Impact of different hyper-parameters on prediction performance.

relevant information. Inspired by the agnostic-network cascade prediction, we design an innovative deep learning model based on feature fusion named CasWarn. This model serializes the cascade features through time slices, then fuses and embeds different key features through our designed neural network module, and then predicts the cascade sequence's virality. Our model incorporates the key features of the information cascade and does not need to consider the cascade network's underlying relationship structure, which is more suitable for the needs of fast, effective, and easy to deploy on agent systems. We conducted comprehensive experiments on two large social network datasets to prove that CasWarn can make timely and effective cascade virality predictions and verified that each feature model of CasWarn is beneficial to improve performance.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author/s.

AUTHOR CONTRIBUTIONS

LG contributed to the core idea of the experiment design and analysis results under the guidance of BZ. YL assisted in the code for experiments and experiment analysis. HZ and HW analyzed the comparative experiment. BZ and AL supervised the research, provided financial support, provided the experimental equipment, and AL is the corresponding author. All authors discussed the results and contributed to the final manuscript.

FUNDING

This work was supported by the National Key Research and Development Program of China (2018YFC0831703) and the National Natural Science Foundation of China (Nos. 61732022, 61732004, 62072131, and 61672020).

REFERENCES

Bakshy, E., Hofman, J. M., Mason, W. A., and Watts, D. J. (2011). "Everyone's an influencer: quantifying influence on twitter," in *Proceedings of the Forth International Conference on Web Search and Web Data Mining, WSDM 2011* (Hong Kong: ACM), 65–74.

Cao, Q., Shen, H., Gao, J., Wei, B., and Cheng, X. (2020). "Popularity prediction on social platforms with coupled graph neural networks," in *WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining* eds J. Caverlee, X. B. Hu, M. Lalmas, and W. Wang (Houston, TX: ACM), 70–78.

Chen, J., Song, X., Nie, L., Wang, X., Zhang, H., and Chua, T. (2016). "Micro tells macro: predicting the popularity of micro-videos via a transductive model," in

Proceedings of the 2016 ACM Conference on Multimedia Conference, MM 2016 (Amsterdam: ACM), 898–907.

Chen, X., Zhou, F., Zhang, K., Trajcevski, G., Zhong, T., and Zhang, F. (2019). "Information diffusion prediction via recurrent cascades convolution," in *35th IEEE International Conference on Data Engineering, ICDE 2019* (Macao: IEEE), 770–781.

Cheng, J., Adamic, L. A., Dow, P. A., Kleinberg, J. M., and Leskovec, J. (2014). "Can cascades be predicted?," in *23rd International World Wide Web Conference, WWW '14* (Seoul: ACM), 925–936.

Ciresan, D. C., Meier, U., Gambardella, L. M., and Schmidhuber, J. (2010). Deep, big, simple neural nets for handwritten digit recognition. *Neural Comput.* 22, 3207–3220. doi: 10.1162/NECO_a_00052

Dong, Y., Johnson, R. A., and Chawla, N. V. (2015). "Will this paper increase your *h*-index?: scientific impact prediction," in *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, WSDM 2015* (Shanghai: ACM), 149–158.

Dong, Y., Johnson, R. A., and Chawla, N. V. (2016). Can scientific impact be predicted? *IEEE Trans. Big Data* 2, 18–30. doi: 10.1109/TBDATA.2016.2521657

Dow, P. A., Adamic, L. A., and Friggeri, A. (2013). "The anatomy of large facebook cascades," in *Proceedings of the Seventh International Conference on Weblogs and Social Media, ICWSM 2013* (Cambridge, MA: The AAAI Press).

Ganesan, D. V., Das, S., Kundu, T., Naren, J., and Bushra, S. (2021). "Deep learning based smart survilance robot," in *Proceedings of the Fist International Conference on Advanced Scientific Innovation in Science, Engineering and Technology, ICASISET 2020* (Chennai).

Gao, S., Ma, J., and Chen, Z. (2015). "Modeling and predicting retweeting dynamics on microblogging platforms," in *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, WSDM 2015* (Shanghai: ACM), 107–116.

Goldberg, Y., and Levy, O. (2014). word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *CoRR*, abs/1402.3722.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. Boston, MA: The MIT Press.

Gu, Z., Wang, L., Chen, X., Tang, Y., Wang, X., Du, X., et al. (2021). Epidemic risk assessment by a novel communication station based method. *IEEE Trans. Netw. Sci. Eng.* 1. doi: 10.1109/TNSE.2021.3058762

Gundersen, K., Alendal, G., Oleynik, A., and Blaser, N. (2020). Binary time series classification with bayesian convolutional neural networks when monitoring for marine gas discharges. *Algorithms* 13:145. doi: 10.3390/a13060145

ichocki, P., Billard, A., and Kahn, P. H. (2011). The ethical landscape of robotics. *IEEE Robot. Autom. Magaz.* 18, 39–50. doi: 10.1109/MRA.2011.940275

Jenders, M., Kasneci, G., and Naumann, F. (2013). "Analyzing and predicting viral tweets," in *22nd International World Wide Web Conference, WWW '13* (Rio de Janeiro: International World Wide Web Conferences Steering Committee / ACM), 657–664.

Jia, Y., Gu, Z., and Li, A., editors (2021). *MDATA: A New Knowledge Representation Model*. Springer International Publishing.

Kefato, Z. T., Sheikh, N., Bahri, L., Soliman, A., Montresor, A., and Girdzijauskas, S. (2018). "CAS2VEC: network-agnostic cascade prediction in online social networks," in *Fifth International Conference on Social Networks Analysis, Management and Security, SNAMS 2018* (Valencia: IEEE), 72–79.

Li, C., Guo, X., and Mei, Q. (2018). "Joint modeling of text and networks for cascade prediction," in *Proceedings of the Twelfth International Conference on Web and Social Media, ICWSM 2018* (Stanford, CA: AAAI Press), 640–643.

Li, C., Ma, J., Guo, X., and Mei, Q. (2017). "Deepcas: An end-to-end predictor of information cascades," in *Proceedings of the 26th International Conference on World Wide Web, WWW 2017* (Perth, WA: ACM), 577–586.

Marsland, S. (2009). *Machine Learning - An Algorithmic Perspective*. Chapman and Hall / CRC machine learning and pattern recognition series. CRC Press.

Pfitzner, R., Garas, A., and Schweitzer, F. (2012). "Emotional divergence influences information spreading in twitter," in *Proceedings of the Sixth International Conference on Weblogs and Social Media* (Dublin: The AAAI Press).

Qiu, J., Tang, J., Ma, H., Dong, Y., Wang, K., and Tang, J. (2018). Deepinf: social influence prediction with deep learning. *CoRR*, abs/1807.05560.

Quan, Y., Jia, Y., Zhou, B., Han, W., and Li, S. (2018). Repost prediction incorporating time-sensitive mutual influence in social networks. *J. Comput. Sci.* 28, 217–227. doi: 10.1016/j.jocs.2017.11.015

Ruiz-del-Solar, J., Loncomilla, P., and Soto, N. (2018). A survey on deep learning methods for robot vision. *CoRR*, abs/1803.10862.

Stieglitz, S., and Dang-Xuan, L. (2012). "Political communication and influence through microblogging-an empirical analysis of sentiment in twitter messages and retweet behavior," in *45th Hawaii International Conference on Systems Science (HICSS-45 2012)* (Maui: IEEE Computer Society), 3500–3509.

Subbian, K., Prakash, B. A., and Adamic, L. A. (2017). "Detecting large reshare cascades in social networks," in *Proceedings of the 26th International Conference on World Wide Web, WWW 2017* (Perth, WA: ACM), 597–605.

Suh, B., Hong, L., Pirolli, P., and Chi, E. H. (2010). "Want to be retweeted? large scale analytics on factors impacting retweet in twitter network," in *Proceedings of the 2010 IEEE Second International Conference on Social Computing, SocialCom / IEEE International Conference on Privacy, Security, Risk and Trust, PASSAT 2010* (Minneapolis, MN: IEEE Computer Society), 177–184.

Tatar, A., de Amorim, M. D., Fdida, S., and Antoniadis, P. (2014). A survey on predicting the popularity of web content. *J. Internet Serv. Appl.* 5, 8:1–8:20. doi: 10.1186/s13174-014-0008-y

Tian, H., Gao, C., Xiao, X., Liu, H., He, B., Wu, H., et al. (2020). "SKEP: sentiment knowledge enhanced pre-training for sentiment analysis," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020* (Association for Computational Linguistics), 4067–4076.

Tsur, O., and Rappoport, A. (2012). " What's in a hashtag?: content based prediction of the spread of ideas in microblogging communities," in *Proceedings of the Fifth International Conference on Web Search and Web Data Mining, WSDM 2012* (Seattle, WA: ACM), 643–652.

Wang, W., Zhang, W., Wang, J., Yan, J., and Zha, H. (2018). "Learning sequential correlation for user generated textual content popularity prediction," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018* (Stockholm: ijcai.org), 1625–1631.

Weng, L., Menczer, F., and Ahn, Y. (2014). "Predicting successful memes using network and community structure," in *Proceedings of the Eighth International Conference on Weblogs and Social Media, ICWSM 2014* (Ann Arbor, MI: The AAAI Press).

Westerlund, M. (2020). An ethical framework for smart robots. *Technol. Innov. Manage. Rev.* 10, 35–44. doi: 10.22215/timreview/1312

Xiao, H., Balaan, Muthu, and Kadry, S. (2020). Artificial intelligence with robotics for advanced manufacturing industry using robot-assisted mixed-integer programming model. *Intell. Serv. Robot.* 6–13:3. doi: 10.1007/s11370-020-00330-x

Yan, R., Tang, J., Liu, X., Shan, D., and Li, X. (2011). "Citation count prediction: learning to estimate future citations for literature," in *Proceedings of the 20th ACM Conference on Information and Knowledge Management, CIKM 2011* (Glasgow: ACM), 1247–1252.

Yano, T., and Smith, N. A. (2010). "What's worthy of comment? content and comment volume in political blogs," in *Proceedings of the Fourth International Conference on Weblogs and Social Media, ICWSM 2010* (Washington, DC: The AAAI Press).

Yuan, N. J., Zhong, Y., Zhang, F., Xie, X., Lin, C., and Rui, Y. (2016). "Who will reply to/retweet this tweet?: the dynamics of intimacy from online social interactions," in *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining* (San Francisco, CA: ACM), 3–12.

Yulita, I. N., Fanany, M. I., and Arymurthy, A. M. (2017). "Bi-directional long short-term memory using quantized data of deep belief networks for sleep stage classification," in *ICCSCI*, 530–538.

Zaman, T., Fox, E. B., and Bradlow, E. T. (2013). A bayesian approach for predicting the popularity of tweets. *CoRR*, abs/1304.6777.

Zhang, J., Liu, B., Tang, J., Chen, T., and Li, J. (2013). "Social influence locality for modeling retweeting behaviors," in *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence* (Beijing: IJCAI/AAAI) ed F. Rossi, 2761–2767.

Zhang, J., Tang, J., Li, J., Liu, Y., and Xing, C. (2015). Who influenced you? predicting retweet via social influence locality. *ACM Trans. Knowl. Discov. Data* 9, 25:1–25:26. doi: 10.1145/2700398

Zhao, Q., Erdogdu, M. A., He, H. Y., Rajaraman, A., and Leskovec, J. (2015). "SEISMIC: a self-exciting point process model for predicting tweet popularity," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Sydney, NSW: ACM), 1513–1522.

Zhou, J., Cui, G., Zhang, Z., Yang, C., Liu, Z., and Sun, M. (2018). Graph neural networks: A review of methods and applications. *CoRR*, abs/1812.08434.

Zhu, X., Wang, Z., Yang, Y., Zhou, B., and Jia, Y. (2018). Influence efficiency maximization: how can we spread information efficiently? *J. Comput. Sci.* 28, 245–256. doi: 10.1016/j.jocs.2017.11.001

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

frontiers in Neurorobotics

ORIGINAL RESEARCH
published: 23 August 2021
doi: 10.3389/fnbot.2021.648101

Rinegan: A Scalable Image Processing Architecture for Large Scale Surveillance Applications

Xi Luo[1], *Lei Feng*[1], *Hao Xun*[1], *Yuanfei Zhang*[1], *Yixin Li*[2] *and Lihua Yin*[1]*

[1] Cyber Space Institute of Advanced Technology, Guangzhou University, Guangzhou, China, [2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China

Image processing is widely used in intelligent robots, significantly improving the surveillance capabilities of smart buildings, industrial parks, and border ports. However, relying on the camera installed in a single robot is not enough since it only provides a narrow field of view as well as limited processing performance. Specially, a target person such as the suspect may appear anywhere and tracking the suspect in such a large-scale scene requires cooperation between fixed cameras and patrol robots. This induces a significant surge in demand for data, computing resources, as well as networking infrastructures. In this work, we develop a scalable architecture to optimize image processing efficacy and response rate for visual ability. In this architecture, the lightweight pre-process and object detection functions are deployed on the gateway-side to minimize the bandwidth consumption. Cloud-side servers receive solely the recognized data rather than entire image or video streams to identify specific suspect. Then the cloud-side sends the information to the robot, and the robot completes the corresponding tracking task. All these functions are implemented and orchestrated based on micro-service architecture to improve the flexibility. We implement a prototype system, called *Rinegan*, and evaluate it in an in-lab testing environment. The result shows that *Rinegan* is able to improve the effectiveness and efficacy of image processing.

Keywords: smart gateway, large scale, image processing, intelligent security robot, microservice

OPEN ACCESS

Edited by:
Zheng Wang,
Southern University of Science and Technology, China

Reviewed by:
Zha Fusheng,
Harbin Institute of Technology, China
Lei Yang,
The University of Hong Kong, Hong Kong, SAR China

***Correspondence:**
Lihua Yin
yinlh@gzhu.edu.cn

Received: *31 December 2020*
Accepted: *26 July 2021*
Published: *23 August 2021*

Citation:
Luo X, Feng L, Xun H, Zhang Y, Li Y and Yin L (2021) Rinegan*: A Scalable Image Processing Architecture for Large Scale Surveillance Applications. Front. Neurorobot. 15:648101. doi: 10.3389/fnbot.2021.648101*

1. INTRODUCTION

Image process has been widely implemented in intelligent robots, which significantly improve the visual ability of smart buildings, industrial parks, border ports and so forth. For example, patrol robots, a critical partner of police officers or area administrators for security surveillance, are usually equipped with cameras that can track suspicious persons by collecting and processing images or video streaming. However, such track task solely relying on individual and narrow view of a single robot is no longer effectiveness in large scale environment. Specifically, the suspect may appear anywhere, and the coverage of a single robot is limited, which is not enough to deal with this situation. Hence, the patrol robots have to cooperate with surveillance cameras distributed in such monitored area. When a suspect appears somewhere, once the surveillance camera captures him, it will send the information to a nearby patrol robot through a communication channel, and the robot will come and track the target person in time.

Such a large scale cooperation of robots and cameras induces a significant surge in demand for image or video data collection, computing resources for object detection and suspect recognition,

as well as communication bandwidth of networking infrastructures. A number of researches (Li et al., 2015; Cai et al., 2016; Redmon et al., 2016; Shi et al., 2018; Tan et al., 2020) have dedicated to improve the ability in object detection by employing supervised learning or reinforcement learning method. These works consider only the visual information collected from the camera implemented inside robot, which are confined in a small surveillance area. Hence, some researchers, e.g., Dolberg et al. (2016), Meng et al. (2017), Chen et al. (2018), Lee et al. (2018), Bevi et al. (2019), Bistritz and Bambos (2019), Chang et al. (2019), and Li et al. (2019), focus on designing edge computing architecture to achieve a high level of scalability and fast response rate, comparing with the widely deployed centralized cloud-based(IoT) solutions. Some others, e.g., Kovatsch et al. (2012), Morabito and Beijar (2016), Morabito et al. (2016, 2017), Morabito (2017), Rufino et al. (2017), Cheng et al. (2018), Dolui and Király (2018), Mendki (2018), and Ogawa et al. (2019), also works on the virtualization of edge-side gateways to enhance the flexibility of above edge computing architecture. All these works either improve the communication and computation efficacy at infrastructure level or design an edge-cloud cooperation training mechanism in which training task runs on cloud side while detection and recognition tasks are executed on the edge. However, they neglect the advantage of the decoupling between the object detection and suspect recognition. Specifically, a patrol robot requires only the location and suspect information rather than the entire image or video stream. Therefore, only the required data are transferred to save the bandwidth. A camera with limited resources can only run an object detection process and send result to cloud side for recognition, and the robots receive only the location and suspect information for tracking process. In this case, both the resource consumption and the risk of data leakage caused by network communication are minimized.

In this work, we present a scalable edge-cloud cooperation architecture, which harmonizes the object detection and recognition applications to facilitate the image processing ability of intelligent robots. In our architecture, the image processing is separate into four phases, i.e., pre-process, object detection, representation, and recognition. The pre-process and object detection tasks are deployed on the edge-side gateway to minimize the response delay and the bandwidth consumption. The cloud-side obtains and processes solely the detected objects from the edge-side smart gateways. All these functions are orchestrated using micro-service techniques, which provide a high level of modularity and interoperability, to optimize the resource allocation. As a result, a robot receives only the location and suspect information for tracking process. Facilitated by the proposed architecture, the robots are able to effectively surveil the entire environment and therefore can make accurate response than that rely on their inside cameras. We implement a prototype system, called *Rinegan*, and evaluate it in an in-lab testing environment. The result shows the efficacy and effectiveness.

In summary, the contributions of our work are as follows.

1. We propose a hierarchical architecture to enhance the efficacy of object recognition applications. The lightweight object detection functions are assigned distributively on gateways to minimize bandwidth consumption between edge and cloud. Therefore, the cloud-side servers receive only structural objects and the recognition result is send to the nearby robots. This architecture greatly reduces the load pressure and computation resources and improves the tracking ability of robots.
2. We develop a prototype system, i.e., *Rinegan*, by implementing the object functions into micro-service instances. Therefore, *Rinegan* can reasonably and deftly orchestrate the tasks like object detection and recognition. In addition, the isolated property in this system can also helps to protect the sensitive information reserved in video data (Wang et al., 2020), i.e., only the required data is transferred through network.
3. We deploy *Rinegan* in a in-lab environment to evaluate its performance. The result shows that *Rinegan* achieves outstanding data processing efficacy and excellent scalability comparing to traditional centralized mode. We have reason to believe that Rinegan can also be applied to large-scale scenarios like smart building (Qiu et al., 2019).

1.1. Organization

The rest of the paper is organized as follows. In section 2, we illustrate the background and motivation, as well as the related work. In section 3, we provide the designation of our proposed gateway system. In section 4, we conduct experiments in a in-lab environment and make comparative analysis and evaluation of its performance. In section 5, we discuss the shortcomings of this work and looked forward to the follow-up work. At last, in section 6, we conclude our work.

2. BACKGROUND AND MOTIVATION

When human beings build the world around us, we often expect to make copies like ours, which can better adapt to the environment and habits in human life. Studies have found that a certain area of the human brain is specifically used to recognize things and make cognitive responses. As a result, people also hope to implement a function such as face recognition on a intelligent security robot to help humans work. In this section, we briefly introduce the background of intelligent security robot, image processing, gateway virtualization and micro-service architecture to illustrate the motivation of our work.

2.1. Related Work

Regarding target detection and recognition, many people are committed to using cloud computing to achieve it. Yaseen et al. (2018) proposed a video analysis system based on cloud computing to realize automatic analysis of video stream data. The experimental results show that the system can be expanded to a certain extent according to the number and size of video streams. However, the author did not consider the load capacity of the cloud in large-scale scenarios. When the amount of data that the system needs to process is particularly large, the cloud may be overloaded, and we can use edge computing to solve this problem. Qi et al. (2017) combined smart phones and cloud

computing to implement a DNN-based target detection system, which is mainly used in vehicle detection and recognition. We noticed that in their work, the authors use smartphones as a video stream acquisition tool. Tasks such as image processing and object detection are processed in the cloud. Such a distributed system may not be suitable for processing a large number of images and videos on congested roads. Our system is implemented through microservices, which can schedule edge-side gateways to process images, taking into account both distributed and large-scale scalability.

Using cloud computing alone is not enough to deal with large-scale scenarios. Some people consider reducing the pressure on the cloud at the edge. Sun et al. (Tian et al., 2019) points out that edge computing can share the load of the cloud center and provide a service environment and computing capabilities at the edge of the network. Moreover, edge computing has low latency and higher bandwidth. Ren et al. (2018) combined edge computing, but they use servers on the edge side, which is difficult to deploy. Moreover, the cloud is only responsible for training the model, and does not use microservice technology to schedule image processing tasks. And Tian et al. (2020) and Luo et al. (2020) mentioned that although cloud computing and cloud storage have brought great convenience, the security issues cannot be ignored. The distributed attack detection system deployed on edge devices in this article also supplements the deficiencies of cloud-only.

About the architecture of edge computing, Cha et al. (2018) designed a blockchain-based smart gateway solution, which uses a digital signature mechanism to effectively protect privacy, and at the same time, it can adaptively maintain user privacy of devices in the network. Mouradian et al. (2018), referred to the ideas of network function virtualization and SDN, and propose a distributed and dynamic configuration gateway structure to solve large-scale disaster management applications. Constant et al. (2017), developed an intelligent gateway system based on fog computing. This architecture can use a knowledge-based model to enhance the quality of interaction between wearable IoT devices and the cloud. Rahmani et al. (2018), used the key position of the gateway at the edge to provide higher-level services, and propose a smart electronic health gateway for smart medical care. We use the concept of fog computing to propose a fog-assisted system architecture that can cope with many of the medical systems challenge. Li et al. (2016), proposed an SDN-based architecture for solving development-level IoT solutions, making devices and gateways programmable for developers, which can quickly reuse ready-made programs and data to create IoT applications. However, none of the above articles takes into account the scalability of the architecture. We use microservice technology to deploy and schedule tasks for edge devices more flexibly to adapt to large-scale scenarios.

2.2. Layout of Intelligent Security Robot Applications

Figure 1 plots the common architecture of intelligent security robot applications. Devices, e.g., security cameras, connect to gateway. Such a gateway provides Internet accessibility for a set of devices. In general, the robot involves functionalities like authentication, data analysis and user interface.

Basically, the gateway should have capability of discovering devices when user makes a "Discover" request. Then, polling of discovered devices should be initiated once the manage platform makes a "Approve" request. The gateway should also make a necessary control action and return "Success" or "Failure" response whenever it receives a control request from the intelligent security robot.

However, the ever-increasing large scale data streams significantly challenge the bandwidth and computation resources of the centralized intelligent security robot infrastructure. For instance, image data collected by the cameras in a smart city application will exhaust the communication bandwidth between gateway and intelligent security robot and disrupt the computation ability of robot-side.

2.3. Image Processing

In the object recognition function of nowadays intelligent security robot, image processing is one of the most important functionality. It enables visibility for these things and applications. **Figure 2** briefly illustrates the workflow of it, which consists of four phases, i.e., pre-process, object detection, representation, and recognition.

In pre-process phase, the raw image data will be transformed, compressed, enhanced, restored and so forth to improve its quality and facilitate the following process. In detection phase, specific objects involved in will be detected and extracted by using corresponding models such as face, vehicle, animal, and building models. In representation phase, these objects will be described as a set of features. Finally, in recognition phase, these objects will be classified as specific person, car, animal, or building.

In general, above four phases are treated as a whole when deploying into intelligent security robot system. Specifically, all of them are either deployed on the edge-side devices or on the cloud-side servers. Both scenarios are not scalable enough to large scale environment. On one hand, most of edge-side devices, even the gateways, are resource-constraint, which are not capable of executing all these tasks. On the other hand, cloud-side deployment significantly increase the stress of such centralized architecture.

2.4. Gateway Virtualization

The possibility of introducing lightweight virtualization technologies, and in particular container-based virtualization, in this kind of environments allows having a system that benefits of the main features introduced by containers, if compared to alternative solutions such as hypervisor-based virtualization or hybrid solutions (Chang et al., 2019): (i) Fast building process, instantiation and initialization of containers. (ii) High density of application/services due to the small container images.

Container-based virtualization (**Figure 3**) provides a different level of abstraction in terms of virtualization and isolation compared to hypervisors. Hypervisors virtualize hardware and device drivers, which generates overhead. On the contrary,

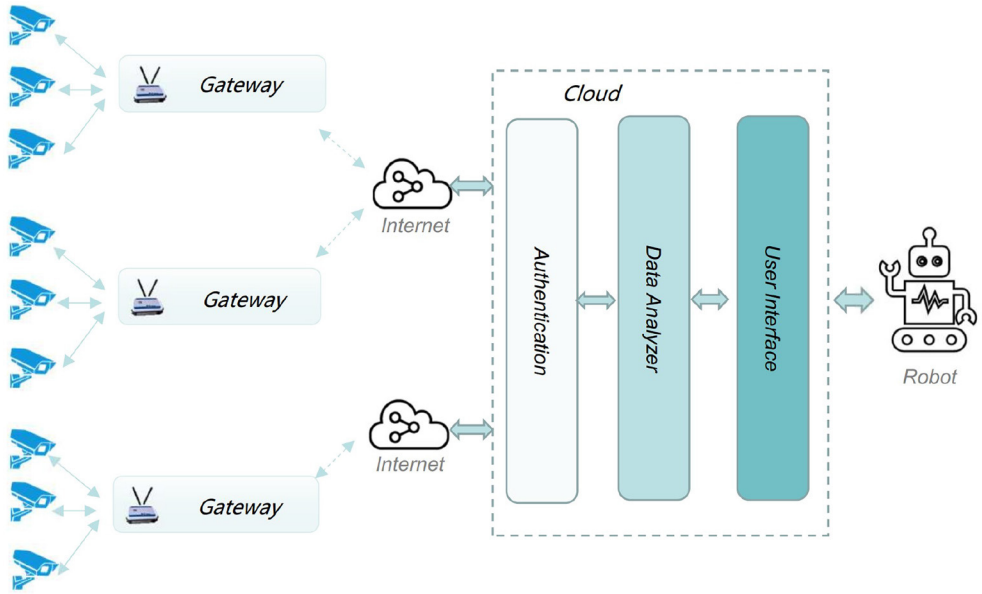


FIGURE 1 | Layout of intelligent security robot applications.

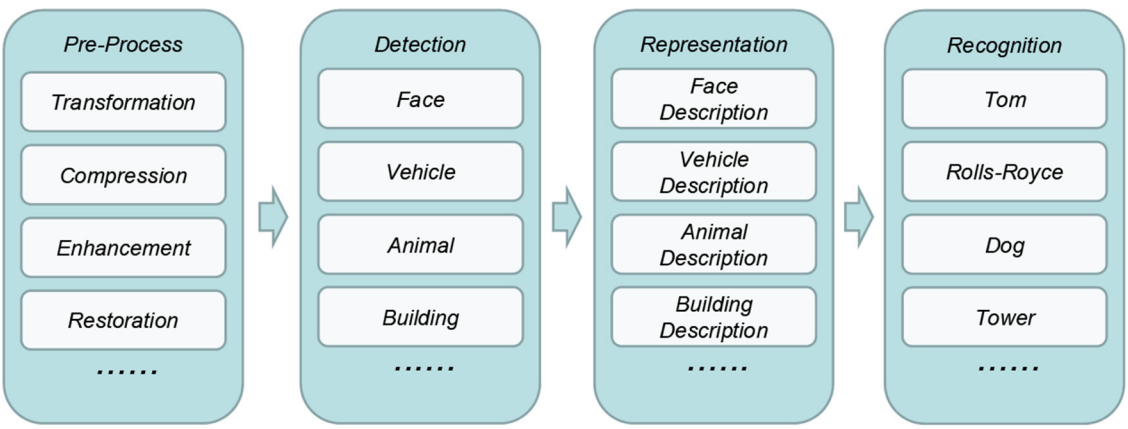


FIGURE 2 | Workflow of image processing.

containers implement isolation of processes at the operating system level, thus, avoiding such overhead (Chang et al., 2019). Due to the shared kernel (as well as operating system libraries), an advantage of container-based solutions is that they can achieve a higher density of virtualized instances, and disk images are smaller compared to hypervisor-based solutions. Moreover, an application can be designed to work in multiple containers, which can interact each other by mean of linking system, with a guarantee of no conflicts with other application containers running on the same machine. It is exactly these features that make possible the integration of the functionality of containers in a wide range of contexts: smart devices, the intelligent security robot, and embedded systems.

2.5. Micro-Service Architecture

Micro-services are an architectural and organizational approach to software development where software is composed of small independent services that communicate over well-defined APIs. These services are owned by small, self-contained teams. Micro-services architectures make applications easier to scale and faster to develop, enabling innovation and accelerating time-to-market for new features.

As shown in **Figure 4**, each component service in a micro-services architecture can be developed, deployed, operated, and scaled without affecting the functioning of other services. Services do not need to share any of their code or implementation with other services. Any communication between individual components happens via well-defined APIs. Each service is

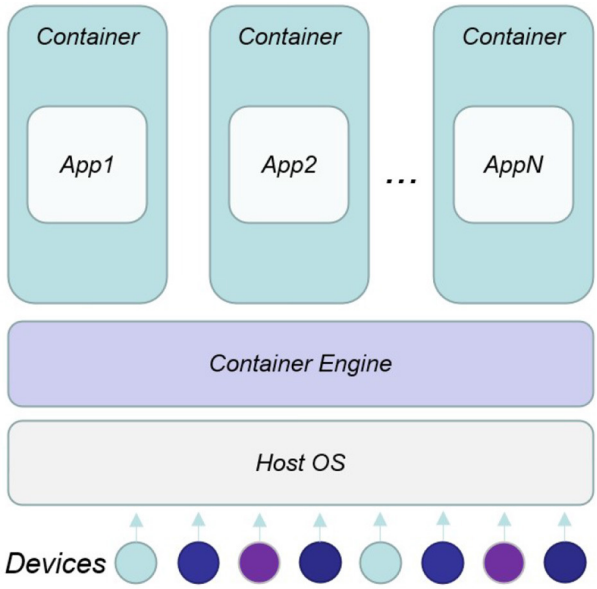


FIGURE 3 | Architecture of containerized gateway.

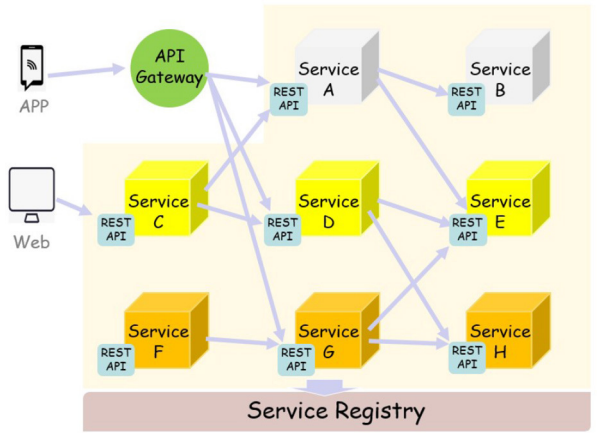


FIGURE 4 | Micro-service architecture.

designed for a set of capabilities and focuses on solving a specific problem. If developers contribute more code to a service over time and the service becomes complex, it can be broken into smaller services.

Micro-services allow each service to be independently scaled to meet demand for the application feature it supports. This enables teams to right-size infrastructure needs, accurately measure the cost of a feature, and maintain availability if a service experiences a spike in demand. Micro-services enable continuous integration and continuous delivery, making it easy to try out new ideas and to roll back if something doesn't work. The low cost of failure enables experimentation, makes it easier to update code, and accelerates time-to-market for new features. Service independence increases an application's resistance to failure. In a monolithic architecture, if a single component fails, it can cause the entire application to fail. With micro-services, applications handle total service failure by degrading functionality, and not crashing the entire application.

3. SYSTEM DESIGN

The image processing architecture we proposed can optimize the object recognition capabilities of the intelligent security robot and improve the efficiency of image processing. We developed a system *Rinegan* based on this architecture. In this section, we will introduce this system in detail.

3.1. System Overview

The *Rinegan* system can effectively improve the image processing performance of the intelligent security robot in a large-scale environment. It allocates part of the image processing task flow to the gateway on the edge-side, allowing the gateway and the cloud to work together to complete the entire image processing flow. The system relieves the pressure of data transmission in the cloud, while also solving the problem of insufficient computing on the edge side. In order to achieve these effects, we have used micro-service technology to develop *Rinegan*, which has good flexibility and scalability. The reason why we use micro-services is because micro-services have a high degree of modularity and interoperability, which can improve the flexibility of the system and optimize the resource allocation of the system. In this section, we will introduce two aspects of system architecture and workflow.

3.1.1. System Architecture

As presented in **Figure 5**, inspired by the structure of the IoT system, *Rinegan* can also be divided into the following three layers: *perception layer*, *network layer*, and *application layer*. Each layer corresponds to a kind of entity, and each layer has its own functions and services. Below we introduce the specific situation of these three layers, respectively.

1. ***Perception layer*** takes the terminal *devices* as the core, the most common ones are various sensors, in this work, specifically the cameras. The perception layer is the data foundation of the whole system, and it's also a key part of information collection. The cameras collect raw data including images and videos, then upload these data to the next layer—the smart gateway.
2. ***Network layer*** is also called the transport layer, in this work, since we are studying a scalable architecture for image processing, it can be interpreted as "*gateway* layer" in a narrow sense here. In this layer, the gateway is responsible for simple processing of images and video streams uploaded by the perception layer to reduce the amount of data transmission. Specifically, simple processing refers to prepocess images and the process of object detection, such as image enhancement, image restoration, We deployed micro-services in the gateways and made reasonable orchestration components for these micro-service. The orchestrator component contains functions like service registry, service detector, load balancer and

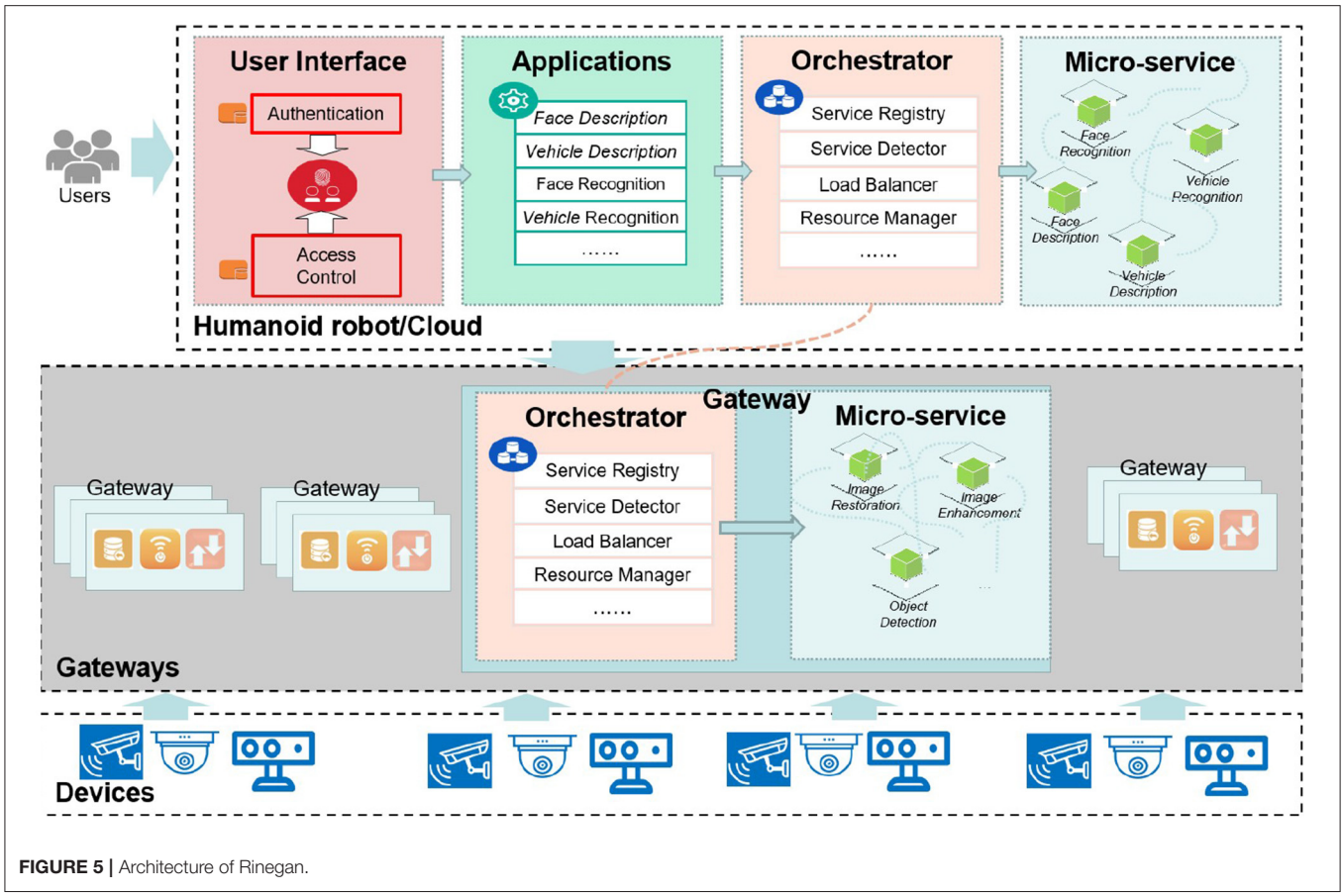


FIGURE 5 | Architecture of Rinegan.

resource manager to ensure the normal and orderly operation of micro-services.

3. ***Application layer*** can be understood as cloud server in the intelligent security robot, in this work, this layer contains four parts, including *user interface*, *applications*, *orchestrator*, *micro-service*. *User interface* is divided into two modules:Authentication and Access control. The authentication module is based on the identity of each user, adopts a standardized identity authentication format, and follows a certain security mechanism to ensure that the access user is a safe user. The access control module authorizes the authenticated user through the control strategy to ensure the legal use of the information resources by the user within the authorized scope. *Applications* are user-oriented tasks, when the cloud receives the image processed by the gateway, this module is responsible for object description and recognition, such as human tracking, traffic flow detection, traffic violation identification and so on. *Orchestrator* module is responsible for the orderly organization of micro-services deployed in the huamnoid robot cloud. The orchestration process is mainly implemented in the cloud server, and there a few scheduling processes in the gateway, as mentioned in the *gateway* layer. *Micro-service* is a small granular service that can be independently developed and deployed. Micro-services generally perform specific functions according to the orchestration of the *Orchestrator*, such as face recognition, face description, vehicle recognition, and vehicle description.

3.1.2. System Workflow

Above we introduced the three-layer architecture of the system we designed, as well as the composition and function of each layer. In this part, we introduce how *Rinegan* works.

As shown in the **Figure 6**, we assume an application scenario to make the workflow more specific. Imaging a scenario where a criminal is driving a car to escape in the city. We need to make the intelligent security robot recognize the vehicle color and license plate number of the vehicle to lock its position.

The terminal camera at the perception layer can collect and store large-scale image data, which we call original images, and the camera transmits the collected data to the gateway layer. After the gateway on the edge-side receives these raw images, the image processing micro-services deployed in them start to work. The first step is *image pre-processing*. Specifically, image transformation includes grayscale and geometric transformation to reduce the amount of data that needs to be processed. Image enhancement means to enhance the useful information in the image. It can be a distorted process. Its purpose is to improve the visual effect of the image. For the application of a given image, it purposefully emphasizes the overall or local characteristics of the image. Image compression refers to the process of reducing

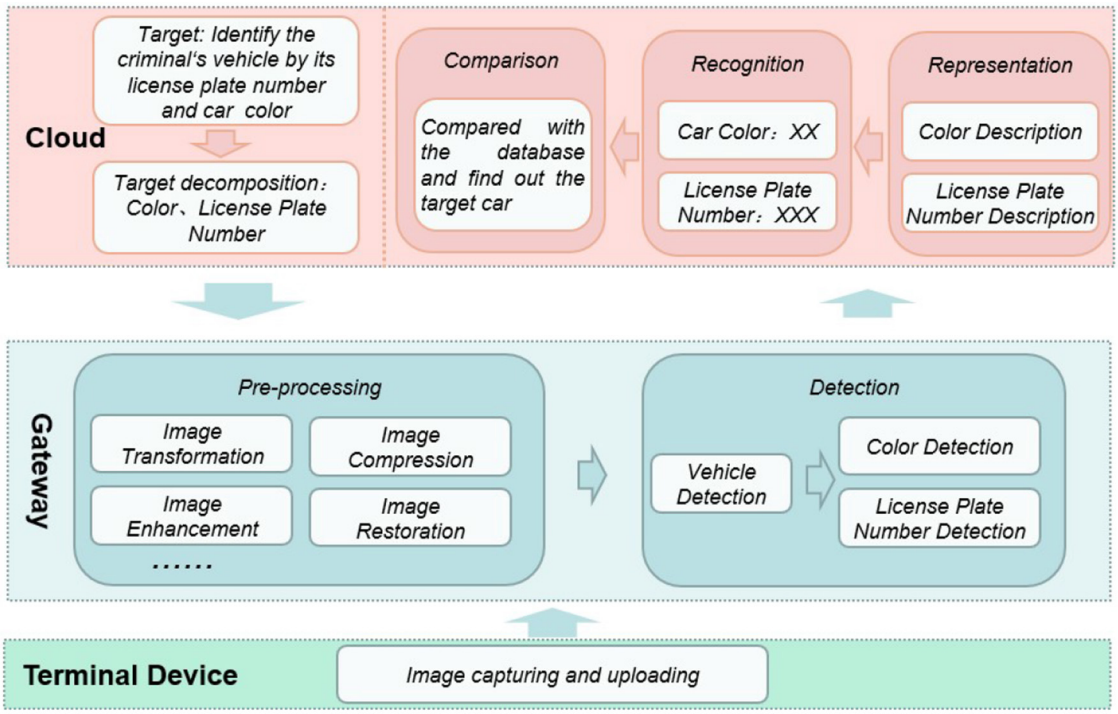


FIGURE 6 | Workflow of *Rinegan*.

image storage or reducing image bandwidth. Image restoration is an objective process that attempts to restore the original content or quality of images with reduced or distorted quality.

The gateway can perform object detection on the pre-processed image. The object detection is roughly divided into three steps. For this scenario, the first is the classification operation. In a given image or a video, the gateway must determine what type of target is in it, that is, the target vehicle. Then there is the positioning operation, to locate the location of the target vehicle. Finally, the detection operation is to detect the color and license plate of the targeted vehicle.

The gateway uploads the processed image to the cloud of the intelligent security robot, and executes the feature representation in the cloud. Through the feature extraction algorithm, the micro-service describes the license plate number and color of the target vehicle for identification. After the cloud recognizes the color and license plate number, compares and matches with all vehicle feature information in the database, and then the target vehicle is identified.

3.2. System Description

In this part, we give a detailed description of each module in the three-layer architecture of the system.

3.2.1. User Interface

For a system, security is the primary consideration in the design process. To ensure the security of the system, it is necessary to ensure that the users interacting with the system are not malicious. There are two aspects of user security that need to be considered. One is the authentication of the user's identity, and the other is the acquisition of the user's operating authority to the system. In response to these two problems, we have developed two functions in the User interface, identity authentication and access control.

We have realized the authentication of the user's identity through the digital certificate. The digital certificate contains the user's identity information and digital signature, which can prove their identity to the entities in the system. The signature certificate in the digital certificate is used to sign user information. To ensure the non-repudiation of information, the encryption certificate is mainly used to encrypt information transmitted by users to ensure the authenticity and integrity of the information. Access control means that after the system completes the identification of the user, it determines the access request authority to the information resource according to the user's identity. We use the discretionary access control strategy to allow legitimate users to access the objects allowed by the policy as users or user groups, and at the same time prevent illegal users from accessing.

3.2.2. Application

This module is user-oriented. After the system completes the user's identity authentication and grants the user certain access rights, the user initiates the corresponding function request to the system according to his needs, so the function of this module is specific. For example, the user may need our system to do face

recognition, track a person, recognize a vehicle, etc., then this module will send these specific function requests to the server, and call the micro-service interface to complete these tasks.

3.2.3. Orchestration

When a system adopts the micro-service architecture, the original business may not change, but the system has been split into many new micro-services. Compared with the traditional architecture, the micro-service architecture will rely more on the collaboration between the micro-services to achieve a complete business process. This collaboration is the service orchestration, which requires a complete orchestration framework to support.

Orchestration is oriented toward executable processes, an executable process is used to coordinate internal and external service interactions, and the overall goal, involved operations, and service call sequence are controlled through the process. The advantage of the orchestrator is that the process control service always knows where each business is going, and monitoring the business has become a relatively simple matter.

3.2.4. Micro-Service

The reason why we use micro-services is because micro-services have a high degree of modularity and interoperability, which can improve the flexibility of the system and optimize the resource allocation of the system. Whether it is the lightweight image processing tasks that we deploy on the edge (e.g., image preprocessing and target detection), or the subsequent image processing tasks that we deploy in the cloud, they are all based on the corresponding micro-service module.

We have used virtualization technology and built these micro-services using Spring boot in the docker container. To ensure that the micro-services in the container are running properly, we set a module named *monitor&protection*, which has three functions: link tracking, service fusing, and service monitor. Service fusing provides a proxy for micro-services. It implements fault tolerance and self-protection by setting timeout and circuit breaker modes for network requests to prevent cascading failure when the service is impossible, which leads to an avalanche effect. Link tracking is used in the local area network to update the connection information. This function can ensure the real-time performance of the micro-service status. Service monitor is used to monitor the running status of each micro-service. When a micro-service has a problem, it will report to the server. It works with link tracking to ensure the normal operation of the micro-service in the container.

For the configuration management and service discovery of micro-services, we also designed a module called *discovery&config*. *Discovery* is RESTful-based service discovery functions, which are mainly used for the discovery and registration of micro-services. When a new micro-service is deployed, this function is responsible for registering the micro-service with the server, which ensures high availability, flexibility, and scalability through heartbeat checking, client-side caching, and other mechanisms. *Config* is used to uniformly manage the configuration of micro-services. Unlike traditional monolithic applications, in293the micro-service architecture, an application system using the micro-service architecture may contain294hundreds of micro-services. Therefore, centralized management configuration is necessary.

As shown in the **Figure 7**, we not only deploy micro-services in the cloud, but also the gateway on the edge side uses the micro-services developed by us when processing images. The entire working process of the system involves the participation of micro-services from beginning to end. For example, at the gateway layer, the gateway completes image restoration, image enhancement and object detection by calling the RESTAPI of the corresponding micro-service. After the cloud receives the data processed by the gateway, the micro-services continue to perform feature description and feature recognition operations on the image, and the process of calling the micro-services is basically the same as the gateway layer.

3.2.5. Gateway

In our proposed architecture and developed system, the gateway is a device on the edge-side that cooperates with the cloud to complete image processing tasks in a large-scale environment.

In addition to its own original functions, for the flexibility of the system, and to reduce the consumption of the cloud to improve the efficiency of the system, we deployed the first two steps of the image processing process in the gateway for execution, namely image pre-processing and object detection.

Similar to the cloud, we also use the docker container-based virtualization technology to deploy the corresponding micro-services, which greatly enhances the scalability of the entire system, and the orchestration and details of the micro-services are basically the same as those of the cloud.

3.2.6. Terminal Devices

In this work, the terminal device is the camera, the physical device of the perception layer. These cameras collect or capture images in specific places and store them in its memory, but the cameras do not have the function of processing images, so these images are transmitted to the gateway for further screening and processing.

4. EVALUATION

In this section, we present our experiment to evaluate the performance of *Rinegan*. We first introduce the implementation detail and then analyze the result.

4.1. Implementation

In order to verify that the architecture we proposed can effectively enhance the efficacy of object recognition, we implemented the protype system based on our proposed image processing architecture—*Rinegan* in our laboratory environment. According to related research (Nefian and Hayes, 1998; Lal et al., 2018), face recognition is currently one of the most commonly used function, such as tracking suspects in cities. So we choose the face recognition function to test the performance of our architecture. Note that, *Rinegan* can not only facilitate the face recognition task but also the other recognition applications such as car and animal recognition. This because these applications are all based on the image process ability just the same as face recognition.

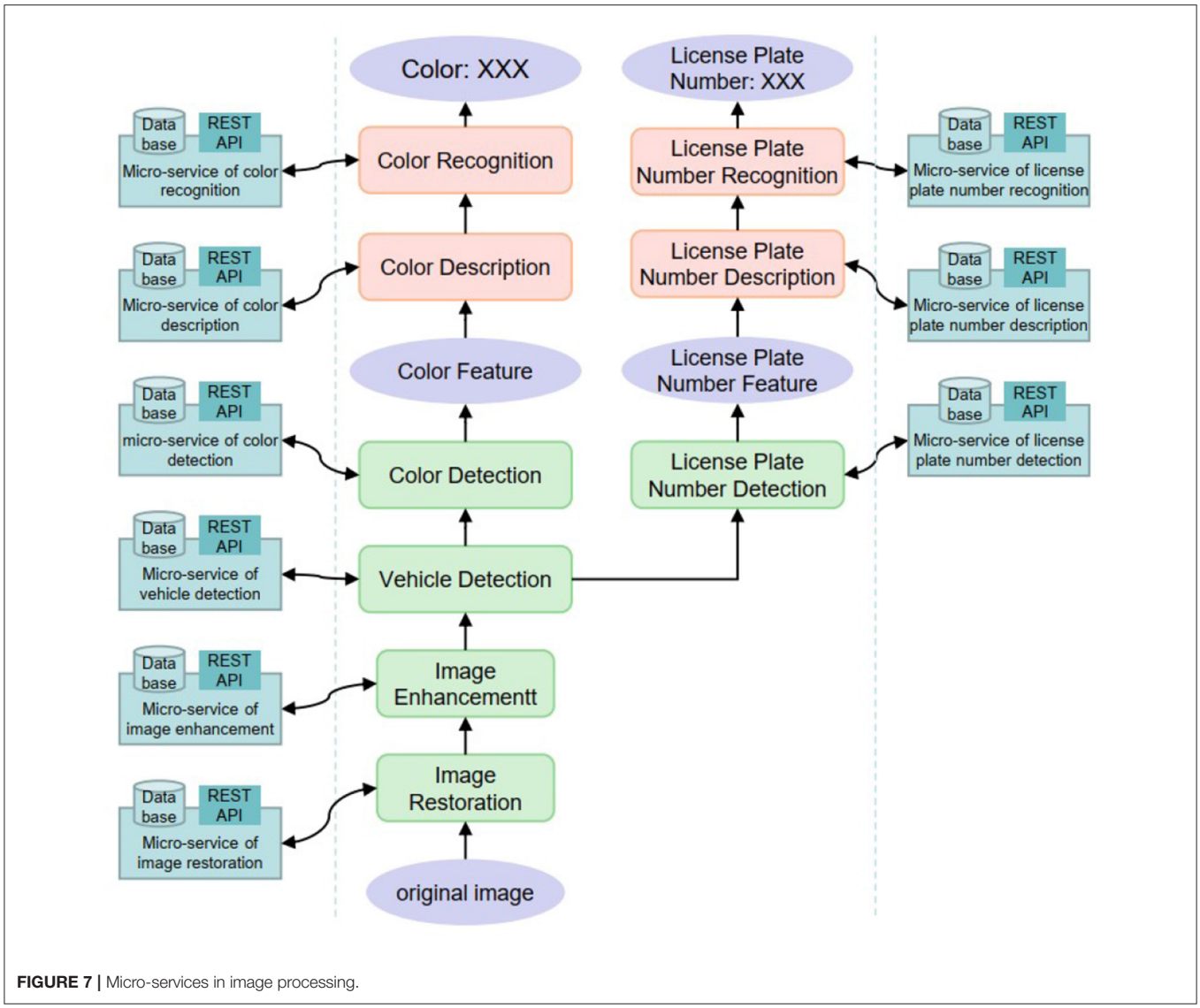


FIGURE 7 | Micro-services in image processing.

We select 1,000 pictures (7.8MB in total) collected from "Large-scale CelebFaces Attributes (CelebA) Dataset"[1] as our experimental data. Our purpose is to find a random select specific target person in these pictures. The face recognition and detection algorithms used in this work are "Face-recognition" project.[2] This project provides convenient APIs, i.e., the "face_location"[2] and "face_distance"[2] APIs, for face detection and recognition, respectively. Moreover, we collect a MP4 video file, named "hamilton_clip.mp4"(640*360, 29.97 fps, 4.9 MB, 78 s),[3] for evaluation. Indeed, the process of video data is not different from that of pictures except the loading task which converts the video stream to a set of frames, i.e., pictures.

In the process of image processing, we deploy the task of object detection on the edge side in which the devices detect the object in the input pictures or video, that is, the person. Finally, the picture will be cropped, and other parts except the face will be cropped off, and the edges will be sent to the cloud to recognize the face. Therefore, the cloud server only needs to do the recognition task. It is worth noting that we ignored the process of sending information from the cloud to the robot since it is a simple task which transfers only the location received from corresponding gateways and the suspect information such as name, passport ID and so forth. At the same time, we also deployed object detection and face recognition tasks all in the cloud (we call it *centralized* later), as a comparative experiment of our architecture.

In the laboratory environment, we used one cloud server and ten edge devices for experiments. In the *Rinegan*, these edge-side devices receive the images or video collected by the camera, then they execute the object detection and image cropping tasks. The

[1] *Celeba*. Available online at: http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html.
[2] *Face-Recognition*. Available online at: https://face-recognition.readthedocs.io/en/latest/face_recognition.html.
[3] *Face Recognition Example*. Available online at: https://github.com/ageitgey/face_recognition/tree/master/examples.

FIGURE 8 | Gateway.

TABLE 1 | Detail of devices used in this experiment.

Device	CPU	Memory
Cloud	Intel(R) Xeon(R) Gold 5218 CPU @ 2.30GHz 64cores	251G
Device1	Intel I7-9700 (8 CPUs * 1 core)	2G
Device2	Intel I7-9700 (8 CPUs * 1 core)	4G
Device3	Intel I7-9700 (8 CPUs * 1 core)	8G
Device4	Intel I7-9700 (1 CPU * 8 cores)	4G
Device5	Intel I7-9700 (1 CPU * 4 cores)	8G
Device6	Intel I7-9700 (1 CPU * 1 core)	8G
Device7	RK 3309	2G
Device8	RK 3309	2G
Device9	RK 3309	4G
Device10	RK 3309	4G

configurations of the CPU, memory of these devices are shown in the **Table 1**. Among these devices, device 1 to device 6 are virtual machine with heterogenous configurations, devices 7 to 10 are the real gateway containers of our laboratory (as shown in **Figure 8**). All these module on the devices are implemented based on Spring microservice architecture[4] which exchange data through RESTFul api, i.e., the data is transferred based on HTTP. Our purpose is to simulate the performance of different operation systems with various configurations to reflect the adaptiveness of heterogeneity of our proposed system.

[4]*Spring Boot*. Available online at: https://spring.io/projects/spring-boot.

4.2. Performance Evaluation

We compared the running time of the two methods we mentioned above. In the *Rinegan*, the running times for face detection and image cropping on the edge devices (100 images per device) are 62, 58, 56, 59, 64, 66, and 67 s in order, respectively. The face recognition on the cloud is about 120 s. In total, the entire process consumes about 187 s. As for the *Centralized* case, it takes 230 s to process all the images in the cloud. Therefore, in our experimental environment, our architecture can increase the processing speed by about 19% (43 s less). It should be noted that in the real environment, there may be far more than 10 edge nodes. So it is foreseeable that the processing speed of our architecture will be faster also.

We also measure the video processing ability of our architecture. In this experiment, when running with 10 edge side devices (we clipping the video into 10 pieces each with 7.8 s per device), we find a abnormal result that the detection tasks takes about 3 s per frame, i.e., 3*30*7.8 = 700 s in total, while the cloud spends only about 570 s to run both recognition and detection. This result indicates that the entire resource of edge side is less than the cloud server in **Table 1**. Therefore, we implement 10 more edge devices with the same configurations to reexamine the performance, each processes about 3.9 s video. In total, the detection and recognition tasks take about 340(edge)+125(cloud) = 465 s, which shows a 18% improvement of efficacy. In real world, the edge side devices may be much more than cloud side and can be equipped with more computing resource if necessary. As we mentioned above, since the video processing is essentially similar to process a sequence of pictures, the increased efficacy depends on the entire computation resources of edge side devices. Moreover, the low rate of video process is caused by the higher resolution ratio rather than the difference of video and picture format. We have to notice that how to improve the efficacy of processing high quality pictures is not concerned in this work. Our purpose is to present a distributed architecture that achieves a better performance than centralized mode.

The cropping process performed on the edge side significantly reduces the size of the total pictures and video by about 45 and 90%, respectively, which therefore greatly reduces the amount of data to be transferred as well as the bandwidth consumption. Another advantage of cropping is that it can prevent the 45 and 90% data from being leaked when transferring through untrusted network channel. Specifically, the surrounding environment which may contains sensitive information can be cropped before transmission.

In summary, the *Rinegan* system we developed effectively reduces bandwidth consumption and significantly improves the execution speed. Moreover, the data leakage risk can be remarkably decreased by our cropping process. More importantly, our system has flexibility and scalability compared with the centralized image process architecture, and can be deployed more flexibly at the edge. Additionally, though the process of all frames (2,356 in total) in our video is resource intensive one can select a part of them, e.g., 1 frame per second, or compress the pictures to improve the detection efficacy.

5. DISCUSSION

From the above experimental results and comparative analysis, the *Rinegan* system we designed performed well, but we also discussed and thought about this work, as shown below.

1. *Limitation of this work.* Our experiment is not large enough as real world IoT scene. However, considering our heterogeneous devices, one of which even configures 1 cpu core, we can infer that our system can have good performance in large-scale scenarios. In this work, we have not determined how to allocate tasks on the edge and cloud to maximize the performance of the system, but this does not affect the current use of the system.
2. *Prospect.* In future work, we should conduct multiple experiments to determine the optimal task volume ratio between the edge and the cloud to optimize the performance of the system. Moreover, our research found that the architecture we proposed is not only suitable for intelligent security robots, but also for the construction in the IoT large-scale scenarios (Qiu et al., 2020; Shafiq et al., 2020). For example, we can deploy this system on smart light poles to achieve collaborative work between the edge and the cloud. We have reason to believe that this will greatly improve the efficiency of smart city construction.

6. CONCLUSION

In this work, we propose a scalable architecture that can improve the image and video processing capabilities of the intelligent security robot and facilitate the tracking task. We reduce the bandwidth consumption of the cloud by deploying distributed image processing functions on the edge. At the same time, by cropping pictures, our architecture can also effectively protect privacy. We developed a system *Rinegan* with this architecture and tested the system in a laboratory environment. The result shows that *Rinegan* consumes less resources and Executes in a shorter time in cloud compared with centralized system. At the same time, our system takes into account the scalability and performs better in large-scale scenarios.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author/s.

AUTHOR CONTRIBUTIONS

XL provided the core idea of this work and did background research. LF and YL wrote the paper and designed the thesis plan. HX and YZ designed the experimental plan and provided help for the result analysis. LY provided the research group with financial support and experimental equipments, as well as being a supportive corresponding author. All authors contributed to the article and approved the submitted version.

FUNDING

This work supports in part by National Key R&D Program of China (No. 2018YFB2100400), Industrial Internet Innovation and Development Project of China (2019), National Science Foundation of China (No. 61872100), and State Grid Corporation of China Co., Ltd technology project (No. 5700-202019187A-0-0-00).

REFERENCES

Bevi, A. R., Shakthipriya, P., and Malarvizhi, S. (2019). Design of software defined networking gateway for the internet-of-things. *Wireless Pers. Commun.* 107, 1273–1287. doi: 10.1007/s11277-019-06335-9

Bistritz, I., and Bambos, N. (2019). "Asymptotically optimal distributed gateway load-balancing for the internet of things," in *10th International Conference on Networks of the Future, NoF 2019* (Rome), 98–101.

Cai, Z., Fan, Q., Feris, R. S., and Vasconcelos, N. (2016). "A unified multi-scale deep convolutional neural network for fast object detection," in *European Conference on Computer Vision* (Amsterdam: Springer), 354–370. doi: 10.1007/978-3-319-46493-0_22

Cha, S-C., Chen, J-F., Su, C., and Yeh, K-H. (2018). A blockchain connected gateway for ble-based devices in the internet of things. *IEEE Access* 6, 24639–24649. doi: 10.1109/ACCESS.2018.2799942

Chang, C-C., Lee, W-K., Liu, Y., Goi, B-M., and Phan, R. C.-W. (2019). Signature gateway: offloading signature generation to iot gateway accelerated by GPU. *IEEE Internet Things J.* 6, 4448–4461. doi: 10.1109/JIOT.2018.2881425

Chen, C., Lin, M., and Liu, C. (2018). Edge computing gateway of the industrial internet of things using multiple collaborative microcontrollers. *IEEE Network* 32, 24–32. doi: 10.1109/MNET.2018.1700146

Cheng, B., Solmaz, G., Cirillo, F., Kovacs, E., Terasawa, K., and Kitazawa, A. (2018). Fogflow: easy programming of IoT services over cloud and edges for smart cities. *IEEE Internet Things J.* 5, 696–707. doi: 10.1109/JIOT.2017.2747214

Constant, N., Borthakur, D., Abtahi, M., Dubey, H., and Mankodiya, K. (2017). Fog-assisted wIoT: a smart fog gateway for end-to-end analytics in wearable internet of things. *arXiv preprint arXiv:1701.08680.*

Dolberg, L., François, J., Chowdhury, S. R., Ahmed, R., Boutaba, R., and Engel, T. (2016). "A generic framework to support application-level flow management in software-defined networks," in *IEEE NetSoft Conference and Workshops, NetSoft 2016* (Seoul: IEEE), 121–125.

Dolui, K., and Király, C. (2018). "Towards multi-container deployment on iot gateways," in *IEEE Global Communications Conference, GLOBECOM 2018* (Abu Dhabi), 1–7.

Kovatsch, M., Lanter, M., and Duquennoy, S. (2012). "Actinium: a restful runtime container for scriptable internet of things applications," in *3rd IEEE International Conference on the Internet of Things, IOT 2012* (Wuxi: IEEE), 135–142.

Lal, M., Kumar, K., Arain, R. H., Maitlo, A., Ruk, S. A., and Shaikh, H. (2018). Study of face recognition techniques: a survey. *Int. J. Adv. Comput. Sci. Appl.* 9, 42–49.

Lee, K., Kuhl, S. J., Bockholt, H. J., Rogers, B. P., and Reed, D. A. (2018). "A cloud-based scientific gateway for internet of things data analytics," in *Proceedings of the Practice and Experience on Advanced Research Computing, PEARC 2018* (Pittsburgh, PA), 34:1–34:8. doi: 10.1145/3219104.3219123

Li, B., Sun, X., and Yu, S. (2019). Designing of internet of things sensor based information gateway using SDN concept. *Int. J. Distribut. Syst. Technol.* 10, 13–24. doi: 10.4018/IJDST.2019010102

Li, H., Lin, Z., Shen, X., Brandt, J., and Hua, G. (2015). "A convolutional neural network cascade for face detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5325–5334.

Li, Y., Su, X., Riekki, J., Kanter, T., and Rahmani, R. (2016). "A SDN-based architecture for horizontal internet of things services," in *2016 IEEE International Conference on Communications, ICC 2016* (Kuala Lumpur: IEEE), 1–7.

Luo, C., Tan, Z., Min, G., Gan, J., Shi, W., and Tian, Z. (2020). A novel web attack detection system for internet of things via ensemble classification. *IEEE Trans. Indus. Inform.* 17, 5810–5818. doi: 10.1109/TII.2020.3038761

Mendki, P. (2018). "Docker container based analytics at iot edge video analytics usecase," in *2018 3rd International Conference On Internet of Things: Smart Innovation and Usages (IoT-SIU)*, 1–4.

Meng, W., Gu, Z., Zhang, M., and Wu, Z. (2017). Two-bit networks for deep learning on resource-constrained embedded devices. *arXiv preprint arXiv:1701.00485*.

Morabito, R. (2017). Virtualization on internet of things edge devices with container technologies: a performance evaluation. *IEEE Access* 5, 8835–8850. doi: 10.1109/ACCESS.2017.2704444

Morabito, R., and Beijar, N. (2016). "Enabling data processing at the network edge through lightweight virtualization technologies," in *2016 IEEE International Conference on Sensing, Communication and Networking (SECON Workshops)*, 1–6.

Morabito, R., Farris, I., Iera, A., and Taleb, T. (2017). Evaluating performance of containerized IoT services for clustered devices at the network edge. *IEEE Internet Things J.* 4, 1019–1030. doi: 10.1109/JIOT.2017.2714638

Morabito, R., Petrolo, R., Loscrì, V., and Mitton, N. (2016). "Enabling a lightweight edge gateway-as-a-service for the internet of things," in *7th International Conference on the Network of the Future, NOF 2016* Búzios, 1–5.

Mouradian, C., Jahromi, N. T., and Glitho, R. H. (2018). NFV and SDN-based distributed iot gateway for large-scale disaster management. *IEEE Internet Things J.* 5, 4119–4131. doi: 10.1109/JIOT.2018.2867255

Nefian, A. V., and Hayes, M. H. (1998). "Hidden markov models for face recognition," in *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98* (Seattle, WA: IEEE), 2721–2724.

Ogawa, K., Kanai, K., Nakamura, K., Kanemitsu, H., Katto, J., and Nakazato, H. (2019). "IoT device virtualization for efficient resource utilization in smart city IoT platform," in *IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops 2019* (Kyoto), 419–422. doi: 10.1109/PERCOMW.2019.8730806

Qi, B., Wu, M., and Zhang, L. (2017). "A DNN-based object detection system on mobile cloud computing," in *17th International Symposium on Communications and Information Technologies, ISCIT 2017* (Cairns, QLD: IEEE), 1–6.

Qiu, J., Du, L., Zhang, D., Su, S., and Tian, Z. (2019). Nei-tte: intelligent traffic time estimation based on fine-grained time derivation of road segments for smart city. *IEEE Trans. Indus. Inform.* 16, 2659–2666. doi: 10.1109/TII.2019.2943906

Qiu, J., Tian, Z., Du, C., Zuo, Q., Su, S., and Fang, B. (2020). A survey on access control in the age of internet of things. *IEEE Internet Things J.* 7, 4682–4696. doi: 10.1109/JIOT.2020.2969326

Rahmani, A. M., Gia, T. N., Negash, B., Anzanpour, A., Azimi, I., Jiang, M., et al. (2018). Exploiting smart e-health gateways at the edge of healthcare internet-of-things: a fog computing approach. *Future Gener. Comput. Syst.*, 78, 641–658. doi: 10.1016/j.future.2017.02.014

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). "You only look once: unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV: IEEE Computer Society), 779–788. doi: 10.1109/CVPR.2016.91

Ren, J., Guo, Y., Zhang, D., Liu, Q., and Zhang, Y. (2018). Distributed and efficient object detection in edge computing: challenges and solutions. *IEEE Netw.* 32, 137–143. doi: 10.1109/MNET.2018.1700415

Rufino, J., Alam, M., Ferreira, J., Rehman, A., and Tsang, K. F. (2017). "Orchestration of containerized microservices for iiot using docker," in *IEEE International Conference on Industrial Technology, ICIT 2017* (Toronto, ON: IEEE), 1532–1536.

Shafiq, M., Tian, Z., Bashir, A. K., Du, X., and Guizani, M. (2020). CorrAUC: a malicious Bot-IoT traffic detection method in IoT network using machine learning techniques. *IEEE Internet Things J.* 8, 3242–3254. doi: 10.1109/JIOT.2020.3002255

Shi, X., Shan, S., Kan, M., Wu, S., and Chen, X. (2018). "Real-time rotation-invariant face detection with progressive calibration networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Salt Lake City, UT: IEEE Computer Society), 2295–2303. doi: 10.1109/CVPR.2018.00244

Tan, M., Pang, R., and Le, Q. V. (2020). "Efficientdet: scalable and efficient object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Seattle, WA: IEEE), 10781–10790. doi: 10.1109/CVPR42600.2020.01079

Tian, Z., Luo, C., Qiu, J., Du, X., and Guizani, M. (2020). A distributed deep learning system for web attack detection on edge devices. *IEEE Trans. Ind. Inform.* 16, 1963–1971. doi: 10.1109/TII.2019.2938778

Tian, Z., Shi, W., Wang, Y., Zhu, C., Du, X., Su, S., et al. (2019). Real-time lateral movement detection based on evidence reasoning network for edge computing environment. *IEEE Trans. Indus. Inform.* 15, 4285–4294. doi: 10.1109/TII.2019.2907754

Wang, Y., Tian, Z., Sun, Y., Du, X., and Guizani, N. (2020). LocJury: an IBN-based location privacy preserving scheme for IoCV. *IEEE Trans. Intell. Transport. Syst.* doi: 10.1109/TITS.2020.2970610

Yaseen, M. U., Anjum, A., Rana, O. F., and Hill, R. (2018). Cloud-based scalable object detection and classification in video streams. *Future Gener. Comput. Syst.* 80, 286–298. doi: 10.1016/j.future.2017.02.003

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

Advantages of publishing in Frontiers

OPEN ACCESS
Articles are free to read for greatest visibility and readership

FAST PUBLICATION
Around 90 days from submission to decision

HIGH QUALITY PEER-REVIEW
Rigorous, collaborative, and constructive peer-review

TRANSPARENT PEER-REVIEW
Editors and reviewers acknowledged by name on published articles

Frontiers
Avenue du Tribunal-Fédéral 34
1005 Lausanne | Switzerland

Visit us: www.frontiersin.org
Contact us: frontiersin.org/about/contact

REPRODUCIBILITY OF RESEARCH
Support open data and methods to enhance research reproducibility

DIGITAL PUBLISHING
Articles designed for optimal readership across devices

FOLLOW US
@frontiersin

IMPACT METRICS
Advanced article metrics track visibility across digital media

EXTENSIVE PROMOTION
Marketing and promotion of impactful research

LOOP RESEARCH NETWORK
Our network increases your article's readership