

Towards neuroscience-inspired intelligent computing: Theory, methods, and applications

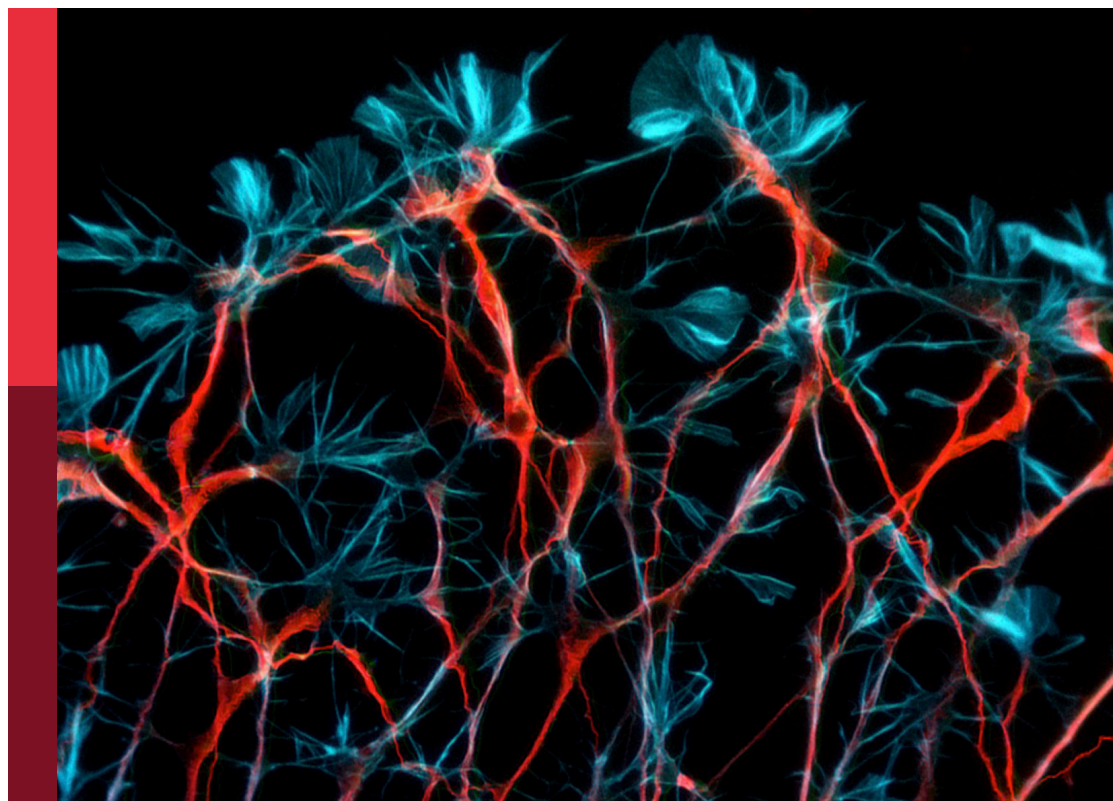
Edited by

Di Wu, Song Deng and Yujie Li

Published in

Frontiers in Computational Neuroscience

Frontiers in Neurorobotics



FRONTIERS EBOOK COPYRIGHT STATEMENT

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

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

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

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

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

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

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

ISSN 1664-8714
ISBN 978-2-83251-917-2
DOI 10.3389/978-2-83251-917-2

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

Towards neuroscience-inspired intelligent computing: Theory, methods, and applications

Topic editors

Di Wu — Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences (CAS), China

Song Deng — Nanjing University of Posts and Telecommunications, China

Yujie Li — Fukuoka University, Japan

Citation

Wu, D., Deng, S., Li, Y., eds. (2023). *Towards neuroscience-inspired intelligent computing: Theory, methods, and applications*. Lausanne: Frontiers Media SA. doi: 10.3389/978-2-83251-917-2

Table of contents

04	Measurement Method of Human Lower Limb Joint Range of Motion Through Human-Machine Interaction Based on Machine Vision Xusheng Wang, Guowei Liu, Yongfei Feng, Wei Li, Jianye Niu and Zhongxue Gan
19	Dynamical Conventional Neural Network Channel Pruning by Genetic Wavelet Channel Search for Image Classification Lin Chen, Saijun Gong, Xiaoyu Shi and Mingsheng Shang
30	Finger Gesture Recognition Using Sensing and Classification of Surface Electromyography Signals With High-Precision Wireless Surface Electromyography Sensors Jianting Fu, Shizhou Cao, Linqin Cai and Lechan Yang
41	cuSCNN: A Secure and Batch-Processing Framework for Privacy-Preserving Convolutional Neural Network Prediction on GPU Yanan Bai, Quanliang Liu, Wenyuan Wu and Yong Feng
54	An Adaptive Time-Varying Impedance Controller for Manipulators Xu Liang, Tingting Su, Zhonghai Zhang, Jie Zhang, Shengda Liu, Quanliang Zhao, Junjie Yuan, Can Huang, Lei Zhao and Guangping He
64	Generative Adversarial Training for Supervised and Semi-supervised Learning Xianmin Wang, Jing Li, Qi Liu, Wenpeng Zhao, Zuoyong Li and Wenhao Wang
74	Research and Application of Fine-Grained Image Classification Based on Small Collar Dataset Huang Chengcheng, Yuan Jian and Qin Xiao
85	An Adaptive Information Security System for 5G-Enabled Smart Grid Based on Artificial Neural Network and Case-Based Learning Algorithms Chengzhi Jiang, Hao Xu, Chuanfeng Huang and Qiwei Huang
95	Post-silicon nano-electronic device and its application in brain-inspired chips Yi Lv, Houpeng Chen, Qian Wang, Xi Li, Chenchen Xie and Zhitang Song
112	Cross-site scripting attack detection based on a modified convolution neural network Huyong Yan, Li Feng, You Yu, Weiling Liao, Lei Feng, Jingyue Zhang, Dan Liu, Ying Zou, Chongwen Liu, Linfa Qu and Xiaoman Zhang
125	Convolutional-de-convolutional neural networks for recognition of surgical workflow Yu-wen Chen, Ju Zhang, Peng Wang, Zheng-yu Hu and Kun-hua Zhong



Measurement Method of Human Lower Limb Joint Range of Motion Through Human-Machine Interaction Based on Machine Vision

Xusheng Wang¹, Guowei Liu², Yongfei Feng^{3*}, Wei Li¹, Jianye Niu² and Zhongxue Gan^{1*}

¹ Academy for Engineering & Technology, Fudan University, Shanghai, China, ² Parallel Robot and Mechatronic System Laboratory of Hebei Province and Key Laboratory of Advanced Forging & Stamping Technology and Science of Ministry of Education, Yanshan University, Qinhuangdao, China, ³ Faculty of Mechanical Engineering & Mechanics, Ningbo University, Ningbo, China

OPEN ACCESS

Edited by:

Di Wu,
Chongqing Institute of Green and
Intelligent Technology (CAS), China

Reviewed by:

Yi He,
Old Dominion University, United States
Ziyun Cai,
Nanjing University of Posts and
Telecommunications, China

*Correspondence:

Yongfei Feng
fengyongfei@nbu.edu.cn
Zhongxue Gan
ganzhongxue@fudan.edu.cn

Received: 05 August 2021

Accepted: 14 September 2021

Published: 15 October 2021

Citation:

Wang X, Liu G, Feng Y, Li W, Niu J and
Gan Z (2021) Measurement Method of
Human Lower Limb Joint Range of
Motion Through Human-Machine
Interaction Based on Machine Vision.
Front. Neurobot. 15:753924.
doi: 10.3389/fnbot.2021.753924

To provide stroke patients with good rehabilitation training, the rehabilitation robot should ensure that each joint of the limb of the patient does not exceed its joint range of motion. Based on the machine vision combined with an RGB-Depth (RGB-D) camera, a convenient and quick human-machine interaction method to measure the lower limb joint range of motion of the stroke patient is proposed. By analyzing the principle of the RGB-D camera, the transformation relationship between the camera coordinate system and the pixel coordinate system in the image is established. Through the markers on the human body and chair on the rehabilitation robot, an RGB-D camera is used to obtain their image data with relative position. The threshold segmentation method is used to process the image. Through the analysis of the image data with the least square method and the vector product method, the range of motion of the hip joint, knee joint in the sagittal plane, and hip joint in the coronal plane could be obtained. Finally, to verify the effectiveness of the proposed method for measuring the lower limb joint range of motion of human, the mechanical leg joint range of motion from a lower limb rehabilitation robot, which will be measured by the angular transducers and the RGB-D camera, was used as the control group and experiment group for comparison. The angle difference in the sagittal plane measured by the proposed detection method and angle sensor is relatively conservative, and the maximum measurement error is not more than 2.2 degrees. The angle difference in the coronal plane between the angle at the peak obtained by the designed detection system and the angle sensor is not more than 2.65 degrees. This paper provides an important and valuable reference for the future rehabilitation robot to set each joint range of motion limited in the safe workspace of the patient.

Keywords: joint range of motion, machine vision, human-robot interaction, rehabilitation robot, human-machine systems

INTRODUCTION

According to the World Population Prospects 2019 (United Nations, 2019), by 2050, one in six people in the world will be over the age of 65 years, up from one in 11 in 2019 (Tian et al., 2021). The elderly are the largest potential population of stroke patients, which will also lead to an increase in the prevalence of stroke (Wang et al., 2019). The lower limb dysfunction caused

by stroke has brought a great burden to the family and society (Coleman et al., 2017; Hobbs and Artemiadis, 2020; Doost et al., 2021; Ezaki et al., 2021). At present, the more effective treatment for stroke is rehabilitation exercise therapy. According to the characteristics of stroke and human limb movement function, it mainly uses the mechanical factors, based on the kinematics, sports mechanics, and neurophysiology, and selects appropriate functional activities and exercise methods to train the patients to prevent diseases and promote the recovery of physical and mental functions (Gassert and Dietz, 2018; D'Onofrio et al., 2019; Cespedes et al., 2021). The integration of artificial intelligence, bionics, robotics, and rehabilitation medicine has promoted the development of the rehabilitation robot industry (Su et al., 2018; Wu et al., 2018, 2020, 2021b; Liang and Su, 2019). With the innovation of technology, the rehabilitation robot has the characteristics of precise motion and long-time repetitive work, which brings a very good solution to many difficult problems of reality, such as the difficulty of standardization of rehabilitation movement, the shortage of rehabilitation physicians, and the increasing number of stroke patients (Deng et al., 2021a,b; Wu et al., 2021a). Lokomat is designed as the most famous lower limb rehabilitation robot that has been carried out in many clinical research (Lee et al., 2021; Maggio et al., 2021; van Kammen et al., 2021). It is mainly composed of three parts: gait trainer, suspended weight loss system, and running platform. Indego is a wearable lower limb rehabilitation robot, designed by Vanderbilt University in the United States (Tan et al., 2020). The user can maintain the balance of the body with the help of a walking stick supported by the forearm or an automatic walking aid. Physiotherabot has the functions of passive training and active training and can realize the interaction between the operator and the robot through a designed human-computer interface (Akdogan and Adli, 2011). However, accurate training, based on the target joint range of motion of the patient, is helpful to limb rehabilitation efficiency of the patients. Joint range of motion, as an important evaluation of the joint activity ability of patients, refers to the angle range of limb joints of the patients to be allowed to move freely. In terms of the human-machine interaction of rehabilitation robots, it is very important to determine the setting of limb safe workspace of the patient and especially setting safety protection at the control level.

The traditional method of measuring joint range of motion is a goniometer. It is mainly composed of three parts: dial scale, fixed arm, and rotating arm. When measuring the joint range of motion, the center of the dial scale should coincide with the axis of the human joint. The traditional goniometer is easy to measure the joint range of motion in the human sagittal plane. However, it is difficult and inaccurate to determine the measurement base position in the human coronal plane. Meanwhile, it requires two rehabilitation physicians to complete the measurement task, one for traction movement of the limb of the patient and the other one for measurement of limb movement of the patient, respectively. The result through a goniometer has low accuracy and is also easily affected by the subjective influence of the physician. Humac Norm is an expensive and automatic measuring device. It includes many auxiliary fixation assemblies (Park and Seo, 2020). During the measurement, the measured

human joint is fixed on the auxiliary assembly. It calculates the joint range of motion by detecting the changes of the auxiliary mechanical assembly. The researchers have also carried out extensive research on the measurement method of joint range of motion by combining a variety of sensor technologies.

An inertial sensor is commonly used to capture the human joint range of motion (Beshara et al., 2020). An inertial measurement unit is developed to accurately measure the knee joint range of motion during the human limb dynamic motion (Ajdaroski et al., 2020). An inertial sensor-based three-dimensional motion capture tool is designed to record the knee, hip, and spine joint motion in a single leg squat posture. It is composed of a triaxial accelerator, gyroscopic, and geomagnetic sensors (Tak et al., 2020). Teufl et al. proposed a high effectiveness three-dimensional joint kinematics measurement method (Teufl et al., 2019). Feng et al. designed a lower limb motion capture system based on the acceleration sensors, which fixed two inertial sensors on the side of the human thigh and calf, respectively (Feng et al., 2016). A gait detection device is proposed for lower-extremity exoskeleton robots, which is integrated with a smart sensor in the shoes and has a compact structure and strong practicability (Zeng et al., 2021). With the development of camera technology, machine vision technology is also introduced into the field of human limb rehabilitation field (Gherman et al., 2019; Dahl et al., 2020; Mavor et al., 2020). However, most of the human limb function evaluation systems based on machine vision require a combination of cameras. The three-dimensional motion capture systems with 12 cameras provide excellent accuracy and reliability, but they are expensive and need to be installed in a large area (Linkel et al., 2016). At present, the MS Kinect (Microsoft Corp., Redmond, WA, USA) is a low-cost, off-the-shelf motion sensor originally designed for video games that can be adapted for the analysis of human exercise posture and balance (Clark et al., 2015). The Kinect could extract the temporal and spatial parameters of human gait, which does not need to accurately represent the human bones and limb segments, which solves the problem of event monitoring, such as the old people fall risk (Dubois and Bresciani, 2018). Based on a virtual triangulation method, an evaluation system for shoulder motion of patients based on the Kinect V2 sensors is designed, which can solve the solution of a single shoulder joint motion range of patients at one time (Cai et al., 2019; Çubukçu et al., 2020; Foreman and Engsberg, 2020). However, how to improve the efficiency of a multi-joint range of motion measurement combined with the teaching traction training method of the rehabilitation physician, how to use a single camera to accurately solve the problem of multi-joints spatial motion evaluation of human lower limbs, and how to avoid camera occlusion in the operation process of the rehabilitation physicians are an important basis for accurate input of lower limb motion information of rehabilitation robot.

In this paper, a measurement method for the multi-joint range of motion of the lower limb based on machine vision is proposed and only one RGB-D camera will be used as image information acquisition equipment. Through the analysis of the imaging principle of the RGB-D camera, the corresponding relationship between the image information and coordinates in

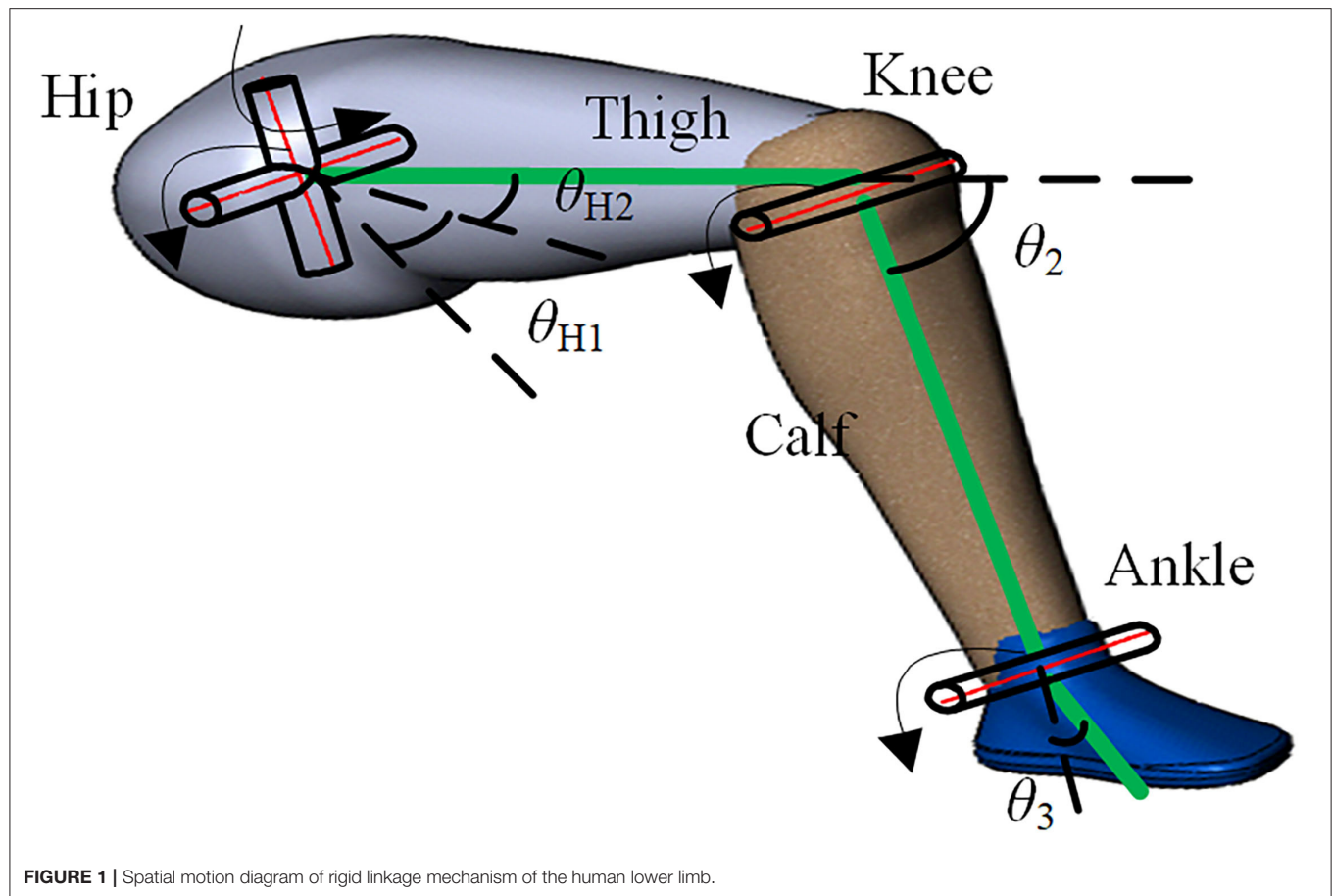


FIGURE 1 | Spatial motion diagram of rigid linkage mechanism of the human lower limb.

three-dimensional space is established. The markers are arranged reasonably on the patient and the rehabilitation robot, and the motion information of the lower limb related joints is transformed into the motion information of the markers. Then, the threshold segmentation method and other related principles are used to complete the extraction of markers. The hip joint range of motion in the coronal plane and sagittal plane and knee joint range of motion in the sagittal plane were calculated by the vector product method. Finally, the experiment is conducted to verify the proposed method.

MATERIALS AND METHODS

Spatial Motion Description of the Human Lower Limbs

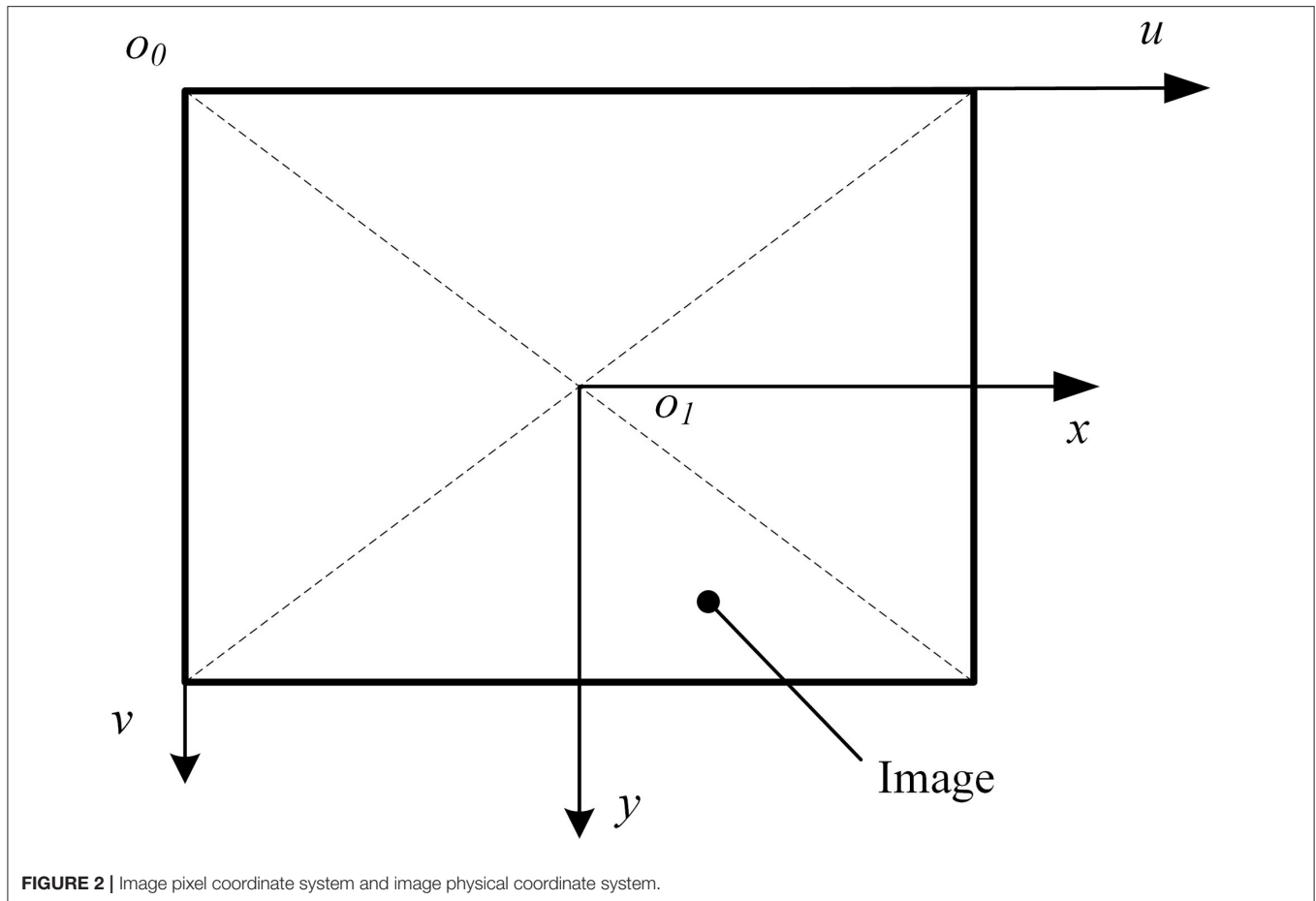
The human lower limb bones are connected by the joints, which could form the basic movement ability. To accurately describe the motion of human lower limb joints in the sagittal plane and the hip joint in the coronal plane, the human hip joint is simplified as two rotation pairs, which rotates around the parallel axis, such as the sagittal axis and the coronal axis, respectively. The knee joint and ankle joint are simplified as one rotation pair, which rotates around the parallel axis of the coronal axis. The thigh, calf, and foot on the human lower limb are simplified as connecting rods. **Figure 1** shows the spatial motion diagram of the rigid linkage mechanism of the human lower limbs. Set the direction of motion

for counterclockwise rotation of the hip joint and ankle joint as positive, while the direction of knee joint motion for clockwise rotation as positive. For the description of the motion of the hip joint in the sagittal plane, the x -axis is taken as the zero-reference angle of the hip joint range of motion, and the angle θ_{H2} between the thigh and the positive direction of the x -axis is taken as the hip joint range of motion. The extension line of the thigh rigid linkage is taken as the zero-reference angle of the knee joint movement angle, and the angle θ_2 between the extension line of the thigh rigid linkage and the calf rigid linkage is the knee joint range of motion. For the hip joint range of motion in the coronal plane, the sagittal plane is taken as the zero-reference plane, and the angle between the plane containing the human thigh and calf and the zero reference plane is taken as the hip joint range of motion θ_{H1} in the coronal plane, in which the outward expansion direction is set as the forward direction of the joint range of motion.

Motion Information Abstraction of Lower Limb Based on Machine Vision

Three Dimensional Coordinate Transformations of Pixels in the Image

Because of the movement of the limb in the three-dimensional space, the depth information of the object is lost from the RGB camera imaging, and the plane information is scaled according



to certain rules. Meanwhile, the lens of the depth camera and the RGB camera is inconsistent, the corresponding pixels are not aligned, so the depth information obtained by the depth camera cannot be directly used for the color images. It is necessary to analyze the relationship between the RGB camera and the depth camera and determine the three-dimensional coordinates of the target object by combining the color images and the depth images. The color camera imaging model is actually the transformation of a point from three-dimensional space to a pixel, involving the pixel coordinate system in the image, the physical coordinate system in the image, and the camera coordinate system in three-dimensional space. The process of camera imaging is that the object at the camera coordinate system in three-dimensional space is transformed into the pixel coordinate system.

As shown in **Figure 2**, an image physical coordinate system x - y is created. The origin of the coordinate system is the center of the image, the x -axis is parallel to the length direction of the image, and the y -axis is parallel to the width direction of the image. The image pixel coordinate system u - v is created. The origin of the coordinate system is the top left corner vertex of the image, the u -axis is parallel to the x -axis of the physical coordinate system, and the v -axis is parallel to the y -axis of the physical coordinate system. Let point P be (u_p, v_p) in the

pixel coordinate system of the image and be (x_p, y_p) in the physical coordinate system. Relative to the pixel coordinates, the physical coordinate system is scaled α times on the u -axis and β times on the v -axis; relative to the origin of the pixel coordinate system, the translation of the origin of the physical coordinate system is (u_0, v_0) . According to the relationship between the above-mentioned coordinate systems, it can be obtained:

$$\begin{cases} u_p = \alpha x_p + u_0 \\ v_p = \beta y_p + v_0 \end{cases} \quad (1)$$

Let the focal distance of the camera lens be f , the main optical axis of the camera is perpendicular to the imaging plane and passes through O_1 , where the optical center of the camera is located on the main optical axis and the distance from the imaging plane is f . As shown in **Figure 3**, the camera coordinate system is created with the optical center as the coordinate origin. The X -axis and Y -axis are parallel to the x -axis and y -axis of the image coordinate system, respectively. Then, the Z -axis is created according to the right-hand rule. Let the coordinates of point P in the camera coordinate system be (X_p, Y_p, Z_p) , and the corresponding projection coordinates in the image physical coordinate system be (x_p, y_p) . According to the relationship

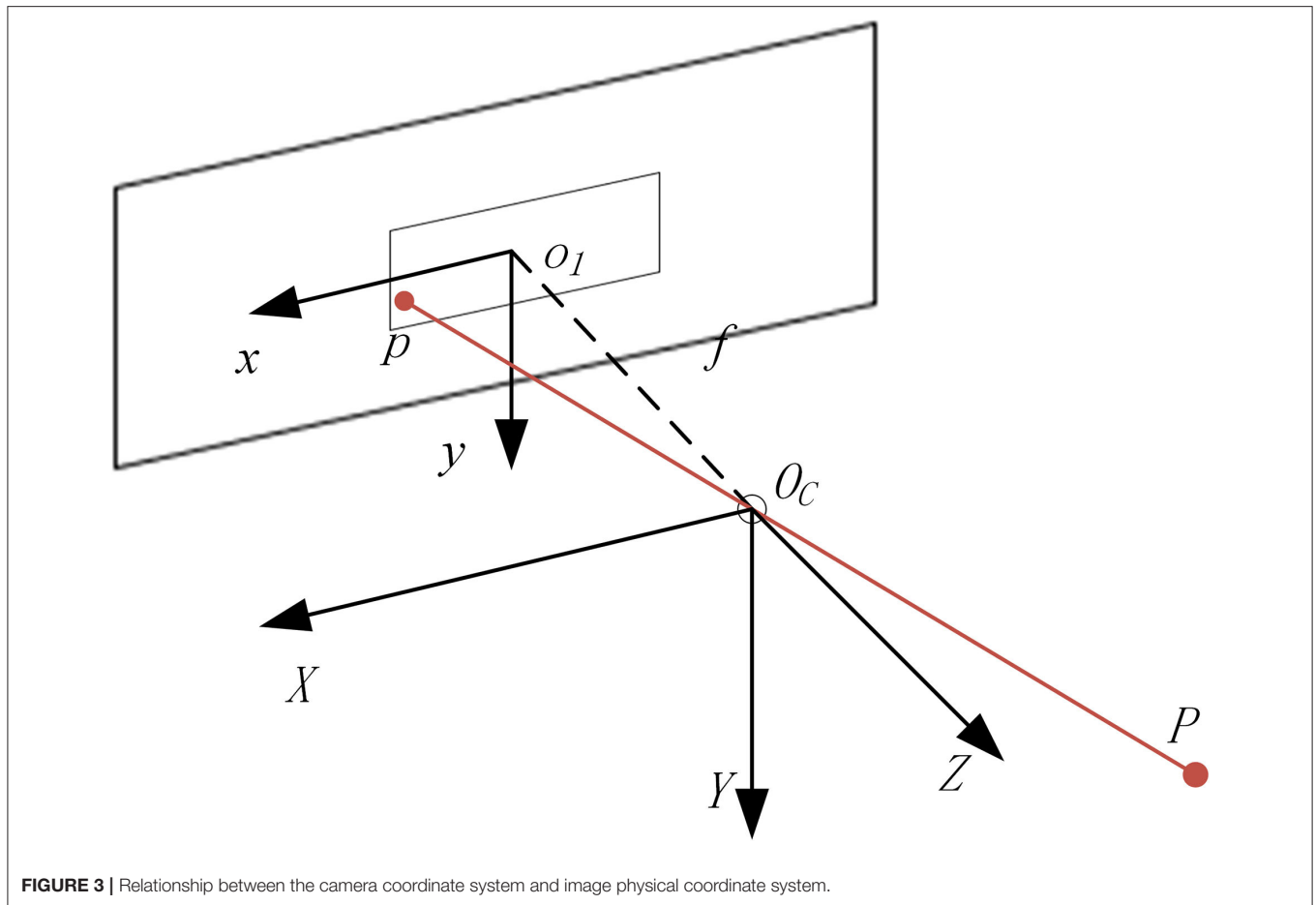


FIGURE 3 | Relationship between the camera coordinate system and image physical coordinate system.

between the camera coordinate system and image coordinate system, the relationship can be obtained:

$$\frac{Z_p}{f} = \frac{X_p}{-x_p} = \frac{Y_p}{y_p} \quad (2)$$

The minus sign in the formula indicates that the image obtained on the physical imaging plane is an inverted image, which can be translated to the front of the camera, and the translation distance along the positive direction of the Z-axis of the camera coordinate system is $2f$. After the phase plane is translated along the positive direction of the z-axis, according to the imaging principle, the imaging at this time is an equal size upright image, and equation (2) is transformed into the following:

$$\frac{Z_p}{f} = \frac{X_p}{x_p} = \frac{Y_p}{y_p} \quad (3)$$

Let $f_x = \alpha f$, $f_y = \beta f$, then by combining formula (1) and formula (3), we can get:

$$\begin{bmatrix} X_p/Z_p \\ Y_p/Z_p \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} u_p \\ v_p \\ 1 \end{bmatrix} \quad (4)$$

Where $(X_p/Z_p, Y_p/Z_p, 1)$ is the projection point of point P in

the normalized plane, let $K = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}$, which represents the internal parameter matrix of the camera.

In the actual imaging process, due to the physical defects of the optical elements in the camera and the mechanical errors in the installation of the optical elements, the images will be distorted. This distortion can be divided into radial distortion and tangential distortion. For any point on the normalized plane, if its coordinate is (x, y) and the corrected coordinate will be $(x_{\text{distorted}}, y_{\text{distorted}})$, then the relationship between the point coordinate and corrected coordinate can be described by five distortion coefficients, and be expressed as follows:

$$\begin{cases} x_{\text{distorted}} = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1 xy + p_2(r^2 + 2x^2) \\ y_{\text{distorted}} = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_2 xy + p_1(r^2 + 2y^2) \end{cases} \quad (5)$$

Where $r = \sqrt{x^2 + y^2}$, k_1, k_2 , and k_3 are the correction coefficients of radial distortion, p_1 and p_2 are the correction coefficients of tangential distortion.

As the depth image and color image are not captured by the same camera, they are not described in the same coordinate system. For the same point in space, their coordinates are inconsistent. Because the pixel coordinate system and



FIGURE 4 | Position arrangement of the camera and markers.

camera coordinate system of depth camera and color camera is established in the same way, and the relative physical positions of the depth camera and color camera are invariable on the same equipment, the rotation matrix R and translation vector t can be used to transform the coordinates between the two camera coordinate systems. Set depth camera internal parameter as K_d and color camera internal parameter as K_c . Let the coordinates of point p in the image be (u_d, v_d) , and the depth value of the point p be z_d . Let the coordinates of the point P in the space coordinate system from the color camera be (X_p, Y_p, Z_p) , then

$$\begin{bmatrix} X_p \\ Y_p \\ Z_p \end{bmatrix} = R \left(z_d K_c^{-1} \begin{bmatrix} u_d \\ v_d \\ 1 \end{bmatrix} \right) + t \quad (6)$$

It is easy to obtain the coordinate in the color coordinate system from Equation (4), and depth information is added to the pixels on the color plane based on Equation (6).

The Position Arrangement of Markers and RGB-D Cameras

To improve the accuracy of joint motion information acquisition, a marker-based motion capture method is adopted. By placing specially designed markers on the seats of the human lower limb and the lower limb rehabilitation robot, the task of obtaining the motion information of human limbs is transformed into the task of capturing and analyzing the spatial position changes of markers. The color information provided by the markers is used as the analysis object. As the detection angle of the target is the hip joint range of motion in the coronal plane and the sagittal plane, and the knee joint range of motion in the sagittal plane, the marker is set as a color strip. The markers of human lower limbs are, respectively, arranged on one side of the thigh and calf, and the direction is along the direction of thigh and calf. When the angle of the knee joint is zero, the two markers should be collinear. The color of the selected marker should be obviously different from the background color, select the blue color here, as shown in **Figure 4**. Because the zero reference angle of the hip joint needs to be set in the sagittal plane, a marker is arranged



FIGURE 5 | Measurement of the hip joint range of motion in the sagittal plane.

on one side of the seat of the lower limb rehabilitation robot, and its length direction is required to be parallel to the seat surface, which is the zero reference of the thigh movement angle. When placing the RGB-D camera, it should face the sagittal plane of the patient, and all markers should be within the capture range of the camera during the movement of the limb of the patient.

Acquisition of Image Information

When measuring the joint range of motion of the patient, the rehabilitation physician drags the leg of the patient in a specific form, then the pictures are collected as shown in **Figure 5**. This section will describe the measurement method of a volunteer. When measuring the hip joint range of motion in the sagittal plane, the rehabilitation physician shall drag the thigh of the patient to move in the sagittal plane, and set no limit on the state of the calf. The rehabilitation physician needs to drag the hip joint of the patient to his maximum and minimum movement limited angle in a sitting position. When measuring the knee joint range of motion in the sagittal plane, the hip joint should be kept still. The rehabilitation physician drags the foot of the patient to drive the calf to move in the sagittal plane. The RGB image collected is shown in **Figure 6**. When determining the hip joint range of motion in the coronal plane, the knee joint of the patient is bent at a comfortable angle. Then, the leg of the patient is dragged to rotate the hip in the coronal plane. The RGB image is shown in **Figure 7**. It should be noted that in the process of dragging, the marker should not be blocked, so as not to affect the camera's acquisition of image information.

Marker Extraction Based on Threshold Segmentation

After the completion of the image acquisition, the motion information of the patient is contained in the markers of each frame of the image. The task at this time is converted to the extraction of markers from the color images. Because the color of the designed marker is obviously different from the background color, the information will be used as the basis of marker extraction.

The rehabilitation training is carried out indoors, and the light is more uniform, and the information of the designed markers will be known, so the color of the markers in RGB space can be obtained in advance and the reference value (R_1, G_1, B_1) can be set. By obtaining the RGB values (R_i, G_i, B_i) of each pixel in the processed image, the distance L between the pixel and the reference value can be obtained. Comparing L with the set threshold T , the pixel whose distance value is less than the set threshold T is set to $(255, 255, 255)$, otherwise, it is set to $(0, 0, 0)$, which can be expressed as follows:

$$(R, G, B) = \begin{cases} (0, 0, 0) & L \leq T \\ (255, 255, 255) & L > T \end{cases} \quad (7)$$

Then, the image is binarized, and the three channels image is converted into a single channel. When the pixel value is $(255, 255, 255)$, the single channel value is set to 255, and when the pixel value is $(0, 0, 0)$, it is set to 0. Then, the extraction task of the markers is completed as shown in **Figure 8**. It shows the binary image results of the measurement of the hip joint



FIGURE 6 | Measurement of the knee joint range of motion in the sagittal plane.

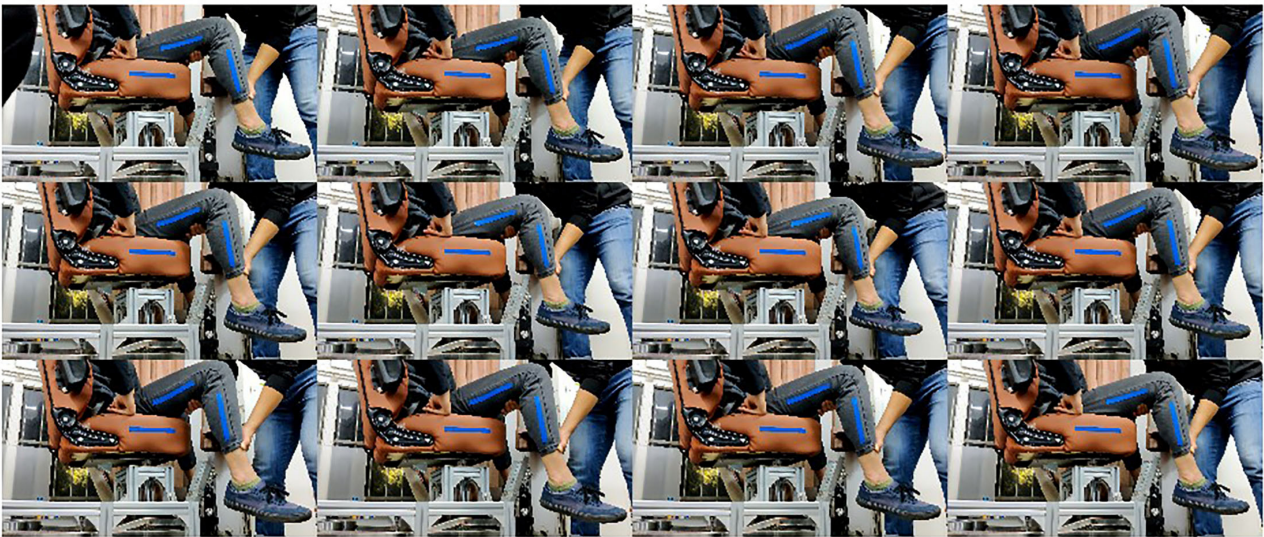


FIGURE 7 | Measurement of the hip joint range of motion in the coronal plane.

range of motion in the sagittal plane, which is processed by threshold segmentation.

Range of Motion Determination of Hip and Knee Joint Based on Image Information Establishment of the Coordinate System in the Sagittal Plane

For the motion of the hip and knee joint in the sagittal plane, to facilitate analysis, a coordinate system is created in the sagittal plane, as shown in **Figure 4**. As the coordinates of the markers

are described in the camera coordinate system, it is necessary to establish the transformation relationship between the coordinate system and the camera coordinate system. The image data are collected according to the motion mode of the measurement of the range of motion of the knee joint in the way described in section Acquisition of Image Information, and the coordinates of the obtained markers on the calf in the camera coordinate system are plane fitted. In the pixel coordinates of multiple pixels, only one pixel is selected to participate in the analysis, and the depth value of the point should be the median value of the depth value of the group of pixels. The coordinates of the

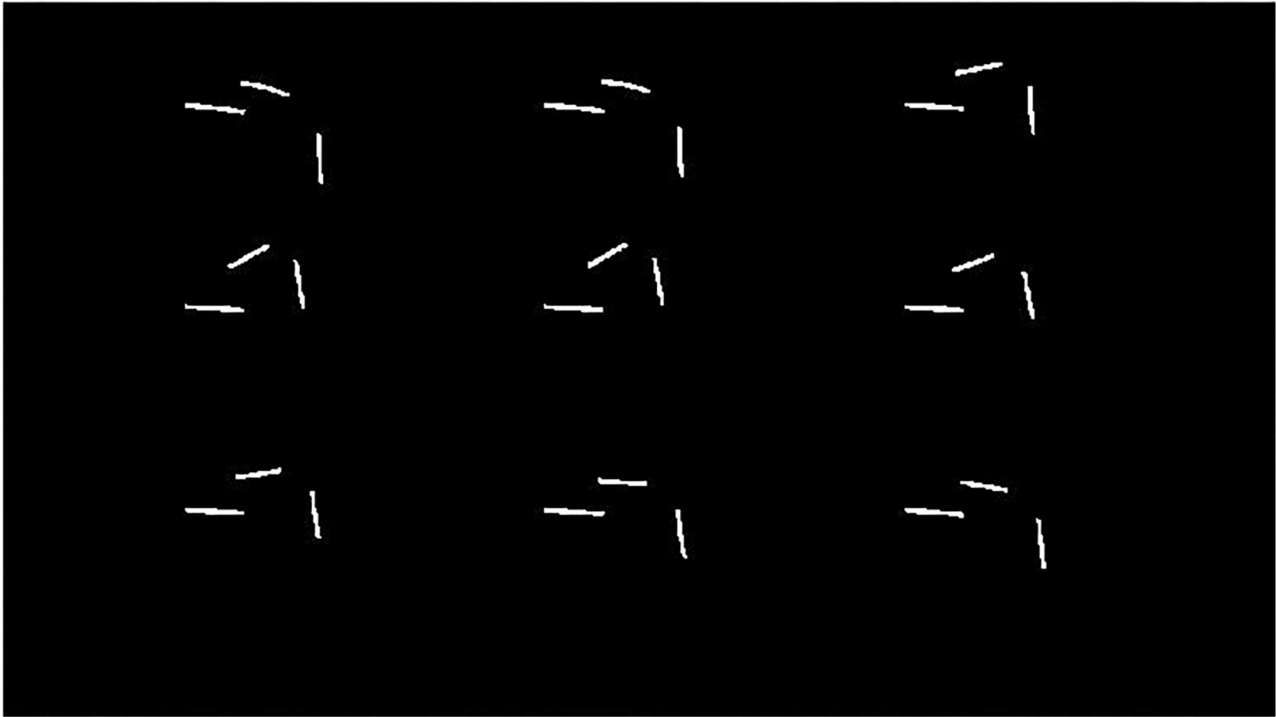


FIGURE 8 | Binary image processed by threshold segmentation.

required pixels in the pixel coordinate system, combined with their depth values, are transformed into the camera coordinate system for description, and the coordinate (x_i, y_i, z_i) in the camera coordinate system can be obtained, where the maximum value of i is equal to k , which is the number of pixels.

Let the fitted plane equation be:

$$ax + by + cz + d = 0 \quad (8)$$

The least square method is used to solve the related unknown parameters, that is, to minimize the value f ,

$$f = \min \left(\sum_{i=1}^k (ax_i + by_i + cz_i + d)^2 \right) \quad (9)$$

where, $a^2 + b^2 + c^2 = 1, a > 0$.

Taking the marker information of each frame image obtained above in section Motion Information Abstraction of Lower Limb Based on Machine Vision as the processing object, the coordinates $(x_{kij}, y_{kij}, z_{kij})$, $(x_{xij}, y_{xij}, z_{xij})$, and $(x_{lij}, y_{lij}, z_{lij})$ of the pixel points of the thigh marker, the calf marker, and the marker on the seat in the camera coordinate system can be obtained, respectively. Where j represents the number of frames of the picture, i represents the number of pixels of the marker described at frame j . It should be noted that the value range of j in the three groups of coordinates is the same, but the value range of i is not the same. By projecting the above coordinates on the sagittal

plane, the camera coordinates $(x'_{kij}, y'_{kij}, z'_{kij})$, $(x'_{xij}, y'_{xij}, z'_{xij})$, and $(x'_{lij}, y'_{lij}, z'_{lij})$ can be obtained.

As the relative position of the seat and the camera does not change during the measurement of joint range of motion, the markers placed on the seat in a frame of the image are taken for line fitting. The fitting space line L must pass through the center of gravity $(\bar{x}, \bar{y}, \bar{z})$ of the marker. Let the direction vector of the line be (l, m, n) . The least square method is used to fit the following equation:

$$\sum_{i=1}^k (x_i - \bar{x})^2 + (y_i - \bar{y})^2 + (z_i - \bar{z})^2 - [l(x_i - \bar{x}) + m(y_i - \bar{y}) + n(z_i - \bar{z})]^2 \quad (10)$$

The formula has a constraint:

$$\begin{cases} l^2 + m^2 + n^2 = 1 \\ l > 0 \end{cases} \quad (11)$$

The unit vectors (u, v, w) perpendicular to the straight line in the plane can be obtained from the obtained direction vectors (l, m, n) and the fitted plane equation, where v is a non-negative value. Take one point (x_o, y_o, z_o) in the plane as the coordinate origin, the direction of the unit vector (l, m, n) is the positive direction of the x -axis, and the direction of the unit vector (u, v, w) is the positive direction of the y -axis. The mathematical

description of the z -axis is determined by the right-hand rule. So far, the establishment of the coordinate system x - o - y - z is completed. The coordinates $(x'_{kij}, y'_{kij}, z'_{kij})$, $(x'_{xij}, y'_{xij}, z'_{xij})$, and $(x'_{lij}, y'_{lij}, z'_{lij})$ in the camera coordinate system are transformed into the coordinate system x - o - y - z , and the coordinates are transformed into $(x''_{ki1}, y''_{ki1}, 0)$, $(x''_{xi1}, y''_{xi1}, 0)$, and $(x''_{li1}, y''_{li1}, 0)$. Since the value z of each coordinate is 0, the three-dimensional coordinate task has been transformed into a two-dimensional task in the coordinate system x - o - y .

Determination of the Hip and Knee Joint Range of Motion in the Sagittal Plane

The markers on the thigh and calf in each frame are all based on the least square method. Take any marker on the thigh in an image as an example to analyze. Let the fitted linear equation be:

$$0 = ax + by + c \quad (12)$$

The least square method is used to solve the parameters a , b , and c , that is, to minimize the value of the polynomial $\sum_{i=1}^k (ax_{ki1} - by_{ki1} + c)^2$, and there is a constraint $a^2 + b^2 = 1$. We can get the coefficients a_k of x and b_k of y , that is, the direction vector $e_k = (b_k, a_k)$ of the straight line is obtained. Similarly, the direction vector $e_x = (b_x, a_x)$ and $e_l = (b_l, a_l)$ representing the fitting line of the calf marker and seat marker, respectively, can also be obtained. The parameters b_k , b_x , and b_l are non-negative, and the motion angle of the thigh is given as follows:

$$\theta_k = \begin{cases} \arccos \frac{e_k \cdot e_l}{|e_k| |e_l|} (e_l \times e_k \geq 0) \\ -\arccos \frac{e_k \cdot e_l}{|e_k| |e_l|} (e_l \times e_k < 0) \end{cases} \quad (13)$$

The motion angle of the calf is:

$$\theta_x = \begin{cases} \arccos \frac{e_x \cdot e_k}{|e_x| |e_k|} (e_k \times e_x \geq 0) \\ -\arccos \frac{e_x \cdot e_k}{|e_x| |e_k|} (e_k \times e_x < 0) \end{cases} \quad (14)$$

Using the same processing method, the angles of hip and knee joints in the different frames can be obtained. Let the angle of the hip joint in frame j be θ_{kj} and the angle of the knee joint in frame j be θ_{xj} . Then, the maximum and minimum of the angle θ_{kj} ($1 \leq j \leq k$) could be obtained, which will be defined as $\theta_{k \max}$ and $\theta_{k \min}$, respectively; the maximum and minimum of the angle θ_{xj} ($1 \leq j \leq k$) could be also achieved, which will be defined as $\theta_{x \max}$ and $\theta_{x \min}$, respectively.

Determination of the Hip Joint Range of Motion in the Coronal Plane

When measuring the patient's hip joint range of motion in the coronal plane, the plane of the thigh and calf of the patient is parallel to the side of the chair at the start, that is, the angle of the hip joint in the coronal plane is 0 degrees. According to the above methods, the images are collected and processed, and the markers on the thigh and calf of each frame are fitted in the way of formula (11), and the normal vectors $e_j = (a_j, b_j, c_j)$ of each

plane are obtained, where j is the number of frames of the image. The motion angle of the hip joint in the coronal plane is:

$$\theta_j = \arccos \frac{|e_j \cdot e_1|}{|e_j| \cdot |e_1|} \quad (15)$$

Let the angle of the hip joint in the coronal plane of frame j be θ_{kgj} , the maximum and minimum values of θ_{kgj} ($1 \leq j \leq k$) can be obtained, which can be set as $\theta_{kg \max}$ and $\theta_{kg \min}$, respectively.

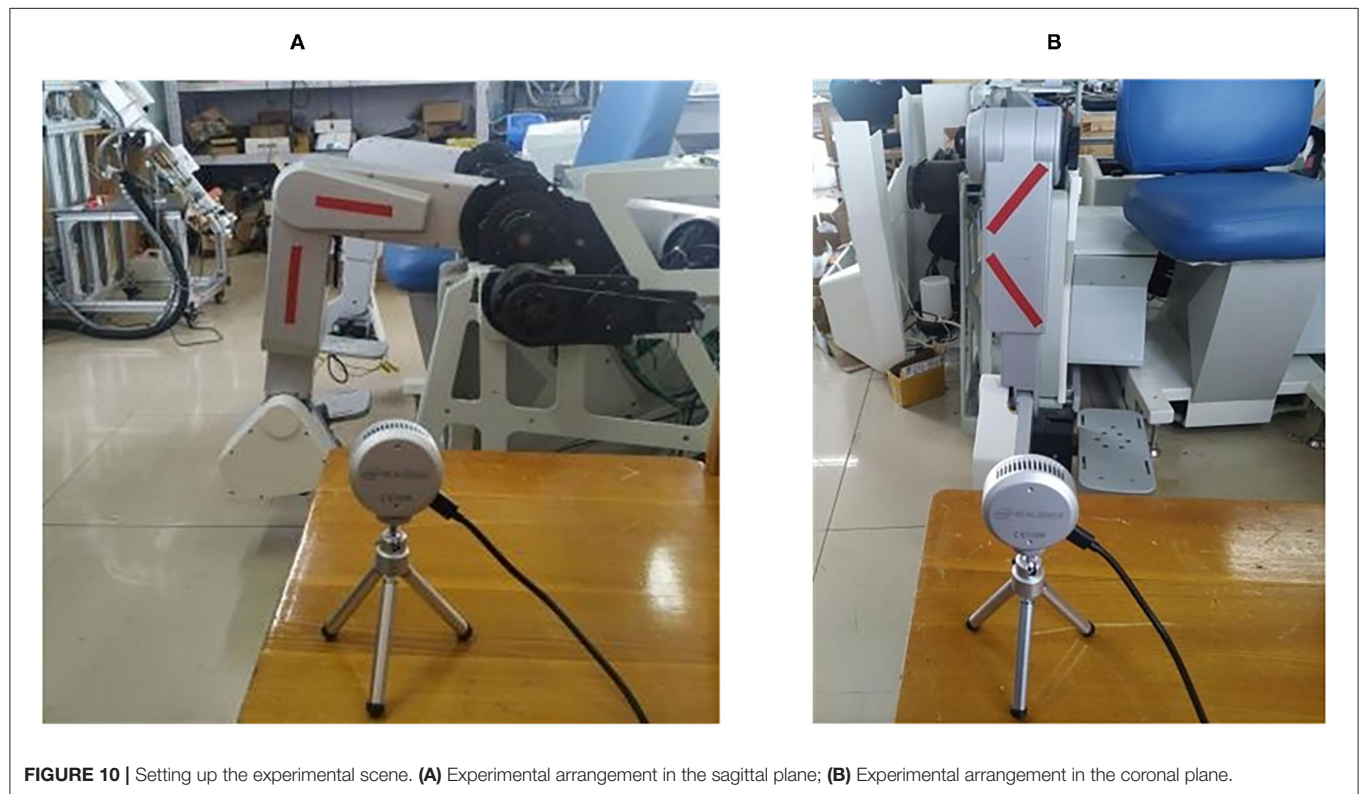
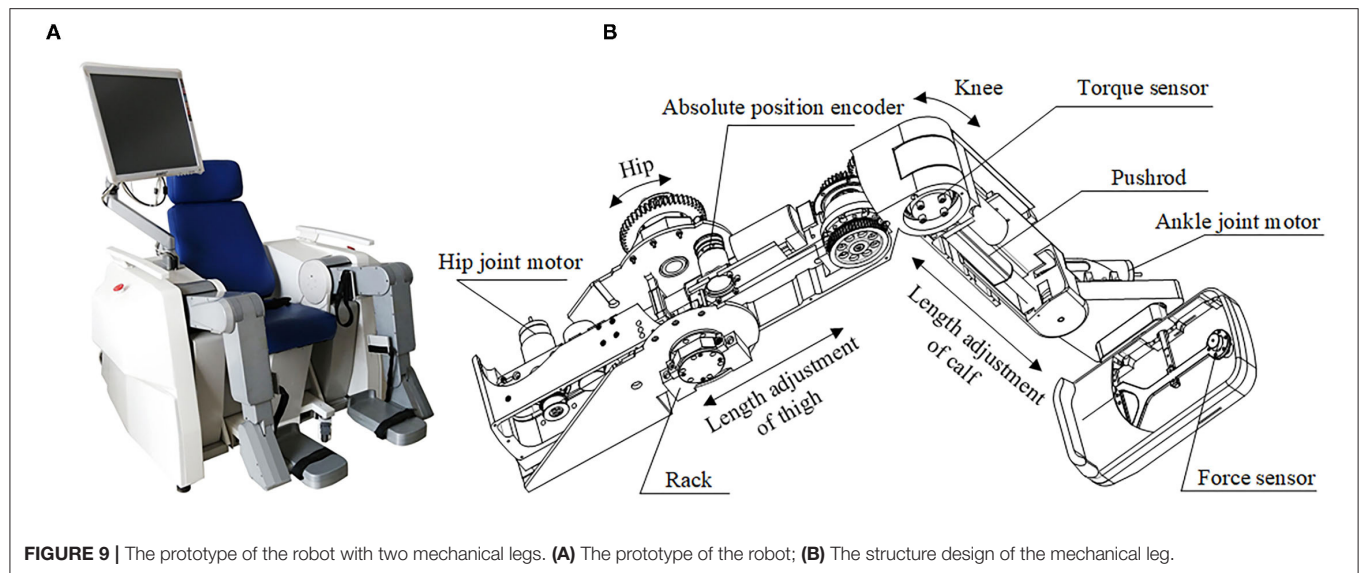
RESULTS

Precision Verification Experiment of the Proposing Detection System

To verify the feasibility of the proposed method based on an RGB-D camera for patients' limb joint range of motion detection, considering the frame rate, resolution, and accuracy of cameras, the L515 camera, produced by Intel Company (CA, USA), is selected. The resolution of the color image and depth image of the camera can reach 1280*720, and both the frame rates can reach 30 fps. As the experiment needs to obtain the coordinate information of the marker in three-dimensional space, the accuracy of depth information will have a direct impact on the accuracy of the detection system. The accuracy of the L515 camera is <5 mm when the distance is at 1 m, and <14 mm at 9 m. It is necessary to ensure that the camera can capture the markers during the movement of the limb of the patient, and the distance between the camera and the affected limb is 0.8–1.5 m. As the control group cannot be set accurately to prove the correctness of the angle measured in the human lower limb experiment, the mechanical leg that replaced the human lower limb is adapted as shown in **Figure 9**.

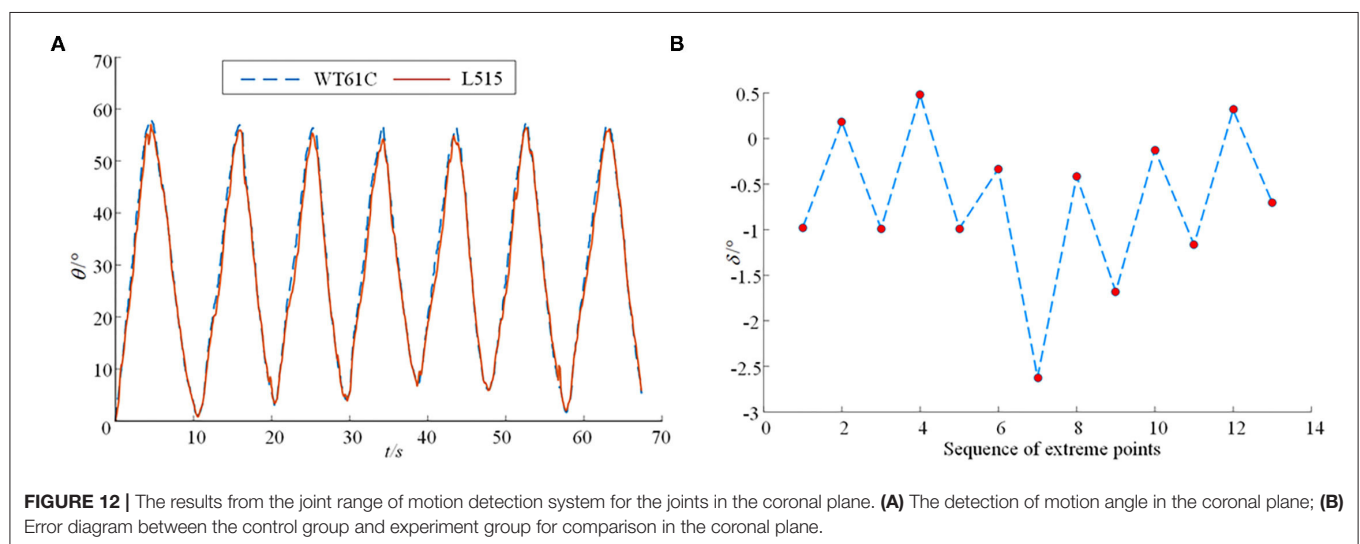
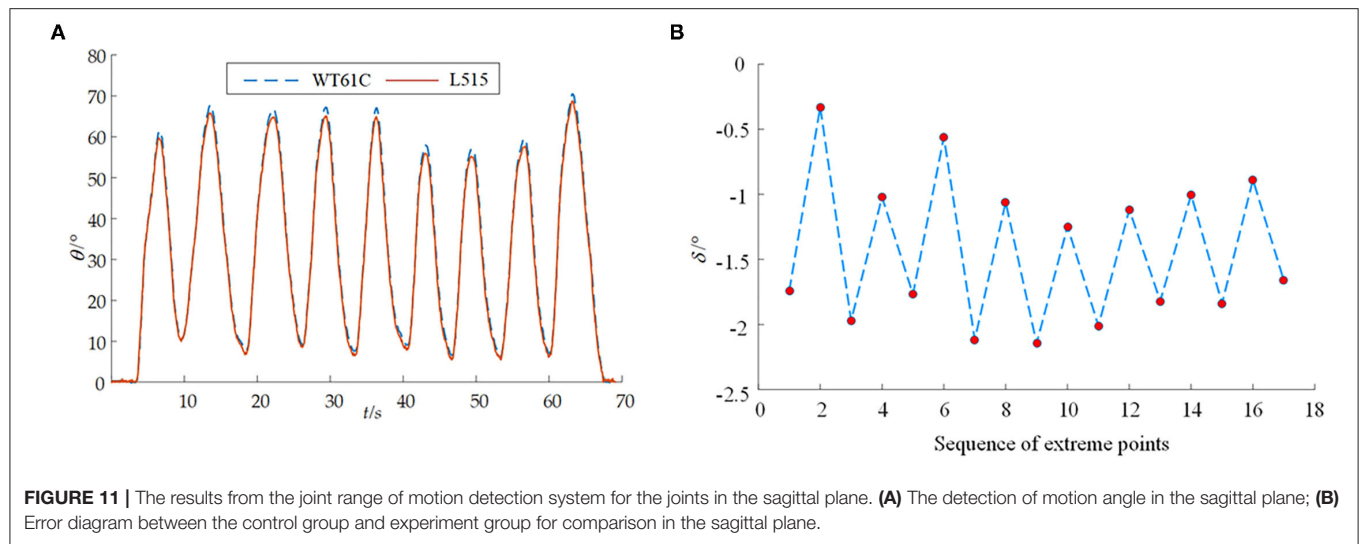
The designed joint range of motion detection system needs to realize the range of motion detection of the hip and knee joint in the sagittal plane. The thigh and calf of the mechanical leg can be regarded as two connecting rods, which are connected by the rotating pairs, and the markers are set up on the thigh and calf, respectively, on one side of the mechanical leg. The motion angles of both the hip and knee in the sagittal plane are represented by the angles between the lines fitted by the strips. Angle sensors WT61C are set on the thigh and calf on the mechanical leg for real-time angles acquisition, and the data from the angle sensors are used as the control group. The dynamic measurement accuracy of the angle sensor (WT61C) is 0.1 degrees, and the output data will be the time and angle.

The red strips are used for the color of the markers as shown in **Figure 10**. The angles between the line fitted by the marker on the mechanical calf and the line fitted by the marker on the mechanical thigh are analyzed and obtained. In order to verify the repetitive accuracy of the designed joint range of motion detection system in the sagittal plane, the calf is designed to move back and forth many times while the thigh is still, and the maximum and minimum values of the motion angle in each back and forth movement are randomly determined. The specific data are shown in **Figure 11A**. The corresponding peak values of angles obtained by the above two methods in time are analyzed here, and the analysis results are shown in **Figure 11B**.



The method of measuring the hip joint range of motion in the coronal plane is essentially based on the plane fitting of two line-markers with a certain angle. At first, the acquired fitting plane is used as the measurement base plane; as the measurement continues, the angle between the new fitting plane and the measurement base plane is obtained again, that is, the solution representing the hip joint range of motion in the coronal plane. The designed joint range of motion detection system also uses

the mechanical leg mentioned above to verify the joint range of motion in the coronal plane. The position arrangement of the markers is shown in **Figure 10B**. In the experiment, the knee axis of the mechanical leg is equivalent to the human hip joint axis in the coronal plane. The calf of the mechanical leg is equivalent to the human lower limb. The calf from the mechanical leg is designed to move round and forth around the rotation knee joint axis many times while the thigh is still, and



the data information of the angle sensors and the RGB-D camera are collected synchronously. To prevent the detection error of the maximum angle caused by the possible pulse interference, the median value average filtering processing is carried out for the obtained motion angles in the coronal plane, and the result is shown in **Figure 12A**. The corresponding peak values of the angles obtained by the above two methods in time are analyzed, and the analysis results are shown in **Figure 12B**.

DISCUSSION

In the precision verification experiment of the proposing detection system, the angle information obtained by the proposed detection system is highly consistent with the angle information obtained by the angle sensor (WT61C), which verifies the correctness of the joint range of motion detection system in the sagittal plane and the coronal plane. When measuring joint

range of motion in the sagittal plane, it is concerned with the maximum and minimum values of the joint angles being measured. Therefore, the corresponding peak values of angles obtained by the proposed method and method through the angle sensor (WT61C) in time are analyzed here, and the analysis results are shown in **Figure 11B**. It shows the difference δ between the angle at the peak obtained by the proposed detection system and the angle sensor. It can be seen from **Figure 11B** that the angle in the sagittal plane measured by the proposed detection system designed is relatively conservative, and the maximum measurement error is not more than 2.2 degrees. It also shows the difference δ in the coronal plane between the angle at the peak obtained by the proposed detection system and the angle sensor. It can be seen from **Figure 12B** that the maximum measurement error between the angle measured by the proposed detection system and the angle sensor is not more than 2.65 degrees.

To our knowledge, no studies have investigated the machine version to achieve the multi-joints spatial motion evaluation of

TABLE 1 | The mean differences through different detection methods.

Measurement method	Abduction/°	Flexion/°	Extension/°
MDCGK	0.33	−2.83	−0.10
MDDGK	1.1	−1.63	0.03
MDALC	−0.56	−1.87	−0.88

human lower limbs. Most studies focus on the gait parameters and their method of estimation using the OptiTrack and Kinect system, such as step length, step duration, cadence, and gait speed, whose messages are different from our study. The reliability and validity analyses of Kinect V2 based measurement system for shoulder motions has been researched in the literature (Çubukçu et al., 2020). The mean differences of the clinical goniometer from the Kinect V2 based measurement system (MDCGK), the mean differences of the digital goniometer from the Kinect V2 based measurement system (MDDGK), and the mean differences of the angle sensor from the proposed method based on the L515 camera (MDALC) are shown in **Table 1**. Compared with the measurement effectiveness of coronal abduction and adduction and sagittal flexion and extension of the shoulder, the proposed lower limb spatial motion measurement system based on the L515 camera also has good relative effectiveness.

Compared with the other methods through the inertial sensors, the proposed method is much easier to obtain the joint range of motion. In terms of operation, it is more convenient for rehabilitation physicians to operate. For the patients with mobility difficulties, only setting marks on the human thigh and calf will not make the patient have a big change in their posture. This paper provides an important parameter basis for the future lower limb rehabilitation robot to set the range of motion of each joint limited in the safe workspace of the patient.

CONCLUSION

This paper proposed a new detection system used for data acquisition before the patients participating in rehabilitation robot training, so as to ensure that the rehabilitation robot does not over-extend any joint of the stroke patients. A mapping between the camera coordinate system and pixel coordinate system in the RGB-D camera image is studied, where the range of motion of the hip and joint, knee joint in the sagittal plane, and hip joint in the coronal plane are modeled *via* least-square analysis. A scene-based experiment with the human in the loop has been carried out, and the results substantiate the

effectiveness of the proposed method. However, considering the complexity of human lower limb skeletal muscle, the regular rigid body of rehabilitation mechanical leg was used as the test object in this paper. Therefore, in practical clinical application, especially for patients with dysfunctional limbs, there are still high requirements for the pasting position and shape of the makers. As the location of the makers, the uniformity of its own shape, and the light intensity of the measurement progress will also affect the measurement results. In future, we will further study the subdivision directions, such as the uniformity of makers, the light intensity of the camera, and the clinical trials.

DATA AVAILABILITY STATEMENT

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

ETHICS STATEMENT

The studies involving human participants were reviewed and approved by the Ethics Committee of Faculty of Mechanical Engineering & Mechanics, Ningbo University. The patients/participants provided their written informed consent to participate in this study.

AUTHOR CONTRIBUTIONS

YF: conceptualization and formal analysis. XW: methodology. GL: software. JN and WL: validation. ZG: investigation, resources, visualization and supervision, and project administration. XW: writing—original draft preparation. YF and ZG: writing—review and editing and funding acquisition. All authors have read and agreed to the published version of the manuscript and agree to be accountable for the content of the work.

FUNDING

This research was funded by the Shanghai Municipal Science and Technology Major Project, grant number 2021SHZDZX0103; the Natural Science Foundation of Zhejiang Province, grant number LQ21E050008; the Educational Commission of Zhejiang Province, grant number Y201941335; the Natural Science Foundation of Ningbo City, grant number 2019A610110; the Major Scientific and Technological Projects in Ningbo City, grant number: 2020Z082; the Research Fund Project of Ningbo University, grant number XYL19029; and the K. C.Wong Magna Fund in Ningbo University, China.

REFERENCES

- Ajdaroski, M., Tadakala, R., Nichols, L., and Esquivel, A. (2020). Validation of a device to measure knee joint angles for a dynamic movement. *Sensors* 20:1747. doi: 10.3390/s20061747
- Akdogan, E., and Adli, M. A. (2011). The design and control of a therapeutic exercise robot for lower limb rehabilitation: physiotherabot. *Mechatronics* 21, 509–522. doi: 10.1016/j.mechatronics.2011.01.005
- Beshara, P., Chen, J. F., Read, A. C., Lagadec, P., Wang, T., and Walsh, W. R. (2020). The reliability and validity of wearable

- inertial sensors coupled with the Microsoft Kinect to measure shoulder range-of-motion. *Sensors* 20:7238. doi: 10.3390/s20247238
- Cai, L. S., Ma, Y., Xiong, S., and Zhang, Y. X. (2019). Validity and reliability of upper limb functional assessment using the Microsoft Kinect V2 sensor. *Appl. Bionics Biomech.* 2019, 1–14. doi: 10.1155/2019/7175240
- Cespedes, N., Raigoso, D., Munera, M., and Cifuentes, C. A. (2021). Long-term social human-robot interaction for Neurorehabilitation: robots as a tool to support gait therapy in the pandemic. *Front. Neurobot.* 15, 1–12. doi: 10.3389/fnbot.2021.612034
- Clark, R. A., Vernon, S., Mentiplay, B. F., Miller, K. J., McGinley, J. L., Pua, Y. H., et al. (2015). Instrumenting gait assessment using the Kinect in people living with stroke: reliability and association with balance tests. *J. Neuroeng. Rehabil.* 12, 1–9. doi: 10.1186/s12984-015-0006-8
- Coleman, E. R., Moudgal, R., Lang, K., Hyacinth, H. I., Awosika, O. O., Kissela, B. M., et al. (2017). Early rehabilitation after stroke: a narrative review. *Curr. Atheroscler. Rep.* 30, 48–54. doi: 10.1007/s11883-017-0686-6
- Çubukçu, B., Yüzgeç, U., Zileli, R., and Zileli, A. (2020). Reliability and validity analyzes of Kinect V2 based measurement system for shoulder motions. *Med. Eng. Phys.* 76, 20–31. doi: 10.1016/j.medengphys.2019.10.017
- Dahl, K. D., Dunford, K. M., Wilson, S. A., Turnbull, T. L. (2020). Wearable sensor validation of sports-related movements for the lower extremity and trunk. *Med. Eng. Phys.* 84, 144–150. doi: 10.1016/j.medengphys.2020.08.001
- Deng, S., Cai, Q. Y., Zhang, Z., and Wu, X. D. (2021a). User behavior analysis based on stacked autoencoder and clustering in complex power grid environment. *IEEE T Intell Transp.* 1–15. doi: 10.1109/TITS.2021.3076607
- Deng, S., Chen, F. L., Dong, X., Gao, G. W., and Wu, X. (2021b). Short-term load forecasting by using improved GEP and abnormal load recognition. *ACM T Internet Techn.* 21, 1–28. doi: 10.1145/3447513
- D'Onofrio, G., Fiorini, L., Hoshino, H., Matsumori, A., Okabe, Y., Tsukamoto, M., et al. (2019). Assistive robots for socialization in elderly people: results pertaining to the needs of the users. *Aging Clin. Exp. Res.* 31, 1313–1329. doi: 10.1007/s40520-018-1073-z
- Doost, M. Y., Herman, B., Denis, A., Spain, J., Galinski, D., Riga, A., et al. (2021). Bimanual motor skill learning and robotic assistance for chronic hemiparetic stroke: a randomized controlled trial. *Neural Regen. Res.* 16, 1566–1573. doi: 10.4103/1673-5374.301030
- Dubois, A., and Bresciani, J. P. (2018). Validation of an ambient system for the measurement of gait parameters. *J. Biomech.* 69, 175–180. doi: 10.1016/j.jbiomech.2018.01.024
- Ezaki, S., Kadone, H., Kubota, S., Abe, T., Shimizu, Y., Tan, C. K., et al. (2021). Analysis of gait motion changes by intervention using robot suit hybrid assistive limb (HAL) in myelopathy patients after decompression surgery for ossification of posterior longitudinal ligament. *Front. Neurobot.* 15, 1–13. doi: 10.3389/fnbot.2021.650118
- Feng, Y. F., Wang, H. B., Lu, T., Vladareanu, V., Li, Q., and Zhao, C. S. (2016). Teaching training method of a lower limb rehabilitation robot. *Int. J. Adv. Robot Syst.* 13, 1–10. doi: 10.5772/62058
- Foreman, M. H., and Engsborg, J. R. (2020). The validity and reliability of the Microsoft Kinect for measuring trunk compensation during reaching. *Sensors* 20:7073. doi: 10.3390/s20247073
- Gassert, R., and Dietz, V. (2018). Rehabilitation robots for the treatment of sensorimotor deficits: a neurophysiological perspective. *J. Neuroeng. Rehabil.* 15, 1–15. doi: 10.1186/s12984-018-0383-x
- Gherman, B., Birlescu, I., Plitea, N., Carbone, G., Tarnita, D., and Pisla, D. (2019). On the singularity-free workspace of a parallel robot for lower-limb rehabilitation. *Proc. Romanian Acad. Ser. A* 20, 383–391.
- Hobbs, B., and Artemiadis, P. (2020). A review of robot-assisted lower-limb stroke therapy: unexplored paths and future directions in gait rehabilitation. *Front. Neurobot.* 14, 1–16. doi: 10.3389/fnbot.2020.00019
- Lee, H. Y., Park, J. H., and Kim, T. W. (2021). Comparisons between Locomat and Walkbot robotic gait training regarding balance and lower extremity function among non-ambulatory chronic acquired brain injury survivors. *Medicine* 100:e25125. doi: 10.1097/MD.00000000000025125
- Liang, X., and Su, T. T. (2019). Quintic pythagorean-hodograph curves based trajectory planning for delta robot with a prescribed geometrical constraint. *Appl. Sci. Basel* 9:4491. doi: 10.3390/app9214491
- Linkel, A., Griskevicius, J., and Daunoraviciene, K. (2016). An objective evaluation of healthy human upper extremity motions. *J. Vibroeng.* 18, 5473–5480. doi: 10.21595/jve.2016.17679
- Maggio, M. G., Naro, A., Manuli, A., Maresca, G., Balletta, T., Latella, D., et al. (2021). Effects of robotic neurorehabilitation on body representation in individuals with stroke: a preliminary study focusing on an EEG-based approach. *Brain Topogr.* 34, 348–362. doi: 10.1007/s10548-021-00825-5
- Mavor, M. P., Ross, G. B., Clouthier, A. L., Karakolis, T., and Tashman, S. (2020). Validation of an IMU suit for Military-Based tasks. *Sensors* 20:4280. doi: 10.3390/s20154280
- Park, J. H., and Seo, T. B. (2020). Study on physical fitness factors affecting race-class of Korea racing cyclists. *J. Exerc. Rehabil.* 16, 96–100. doi: 10.12965/jer.1938738.369
- Su, T. T., Cheng, L., Wang, Y. K., Liang, X., Zheng, J., and Zhang, H. J. (2018). Time-optimal trajectory planning for delta robot based on quintic pythagorean-hodograph curves. *IEEE Access.* 6, 28530–28539. doi: 10.1109/ACCESS.2018.2831663
- Tak, I., Wiertz, W. P., Barendrecht, M., and Langhout, R. (2020). Validity of a new 3-D motion analysis tool for the assessment of knee, hip and spine joint angles during the single leg squat. *Sensors* 20:4539. doi: 10.3390/s20164539
- Tan, K., Koyama, S., Sakurai, H., Tanabe, S., Kanada, Y., and Tanabe, S. (2020). Wearable robotic exoskeleton for gait reconstruction in patients with spinal cord injury: a literature review. *J. Orthop. Transl.* 28, 55–64. doi: 10.1016/j.jot.2021.01.001
- Teufl, W., Miezal, M., Taetz, B., Fröhlich, M., and Bleser, G. (2019). Validity of inertial sensor based 3D joint kinematics of static and dynamic sport and physiotherapy specific movements. *PLoS ONE* 14:e0213064. doi: 10.1371/journal.pone.0213064
- Tian, Y., Wang, H. B., Zhang, Y. S., Su, B. W., Wang, L. P., Wang, X. S., et al. (2021). Design and evaluation of a novel person transfer assist system. *IEEE Access.* 9, 14306–14318. doi: 10.1109/ACCESS.2021.3051677
- United Nations (2019). *World Population Prospects 2019: Highlights*. Available online at: <https://www.un.org/development/desa/publications/world-population-prospects-2019-highlights.html> (accessed June 17, 2019).
- van Kammen, K., Reinders-Messelink, H. A., Elsinghorst, A. L., Wesselink, C. F., Meeuwisse-de Vries, B., van der Woude, L. H. V., et al. (2021). Amplitude and stride-to-stride variability of muscle activity during Lokomat guided walking and treadmill walking in children with cerebral palsy. *Eur. J. Paediatr. Neuro.* 29, 108–117. doi: 10.1016/j.ejpn.2020.08.003
- Wang, L. D., Liu, J. M., Yang, Y., Peng, B., and Wang, Y. L. (2019). The prevention and treatment of stroke still face huge challenges —brief report on stroke prevention and treatment in China, 2018. *Chin. Circ. J.* 34, 105–119. doi: 10.3969/j.issn.1000-3614.2019.02.001
- Wu, D., Luo, X., Shang, M. S., He, Y., Wang, G. Y., and Wu, X. D. (2020). A data-characteristic-aware latent factor model for Web service QoS prediction. *IEEE T Knowl. Data En.* 1–12. doi: 10.1109/TKDE.2020.3014302
- Wu, D., Luo, X., Shang, M. S., He, Y., Wang, G. Y., and Zhou, M. C. (2021a). A deep latent factor model for high-dimensional and sparse matrices in recommender systems. *IEEE Transac. Syst. Man Cybernet. Syst.* 51, 4285–4296. doi: 10.1109/TSMC.2019.2931393
- Wu, D., Luo, X., Wang, G. Y., Shang, M. S., Yuan, Y., and Yan, H. Y. (2018). A highly-accurate framework for self-labeled semi-supervised classification in industrial applications. *IEEE T Ind. Inform.* 14, 909–920. doi: 10.1109/TII.2017.2737827
- Wu, D., Shang, M. S., Luo, X., and Wang, Z. D. (2021b). An L1-and-L2-norm-oriented latent factor model for recommender systems. *IEEE T Neur. Net Lear.* 1–14. doi: 10.1109/TNNLS.2021.3071392
- Zeng, D. Z., Qu, C. X., Ma, T., Qu, S., Yin, P., Zhao, N., et al. (2021). Research on a gait detection system and recognition algorithm for lower limb

exoskeleton robot. *J. Braz. Soc. Mech. Sci.* 43:298. doi: 10.1007/s40430-021-03016-2

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

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

this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

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



Dynamical Conventional Neural Network Channel Pruning by Genetic Wavelet Channel Search for Image Classification

Lin Chen¹, Saijun Gong², Xiaoyu Shi^{1*} and Mingsheng Shang¹

¹ Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences (CAS), Chongqing, China, ² School of Information Science and Technology, Tibet University, Lhasa, China

OPEN ACCESS

Edited by:

Song Deng,
Nanjing University of Posts and
Telecommunications, China

Reviewed by:

Yi He,
Old Dominion University, United States
Ji Xu,
Guizhou University, China

*Correspondence:

Xiaoyu Shi
xiaoyushi@cigit.ac.cn

Received: 18 August 2021

Accepted: 28 September 2021

Published: 27 October 2021

Citation:

Chen L, Gong S, Shi X and Shang M
(2021) Dynamical Conventional Neural
Network Channel Pruning by Genetic
Wavelet Channel Search for Image
Classification.
Front. Comput. Neurosci. 15:760554.
doi: 10.3389/fncom.2021.760554

Neural network pruning is critical to alleviating the high computational cost of deep neural networks on resource-limited devices. Conventional network pruning methods compress the network based on the hand-crafted rules with a pre-defined pruning ratio (PR), which fails to consider the variety of channels among different layers, thus, resulting in a sub-optimal pruned model. To alleviate this issue, this study proposes a genetic wavelet channel search (GWCS) based pruning framework, where the pruning process is modeled as a multi-stage genetic optimization procedure. Its main ideas are 2-fold: (1) it encodes all the channels of the pertained network and divide them into multiple searching spaces according to the different functional convolutional layers from concrete to abstract. (2) it develops a wavelet channel aggregation based fitness function to explore the most representative and discriminative channels at each layer and prune the network dynamically. In the experiments, the proposed GWCS is evaluated on CIFAR-10, CIFAR-100, and ImageNet datasets with two kinds of popular deep convolutional neural networks (CNNs) (ResNet and VGGNet). The results demonstrate that GNAS outperforms state-of-the-art pruning algorithms in both accuracy and compression rate. Notably, GNAS reduces more than 73.1% FLOPs by pruning ResNet-32 with even 0.79% accuracy improvement on CIFAR-100.

Keywords: neural network pruning, neural architecture search, wavelet features, neural network compression, image classification

1. INTRODUCTION

Deep convolutional neural networks (CNNs) have achieved substantial progress in many research fields, such as computer vision (Wang et al., 2019), natural language processing (Giménez et al., 2020), and information recommendation (Wu et al., 2021a,b). However, the number of parameters in deep CNN-based models (e.g., ResNet-50 He et al., 2016) generally exceeds hundreds of megabytes. It needs billions of floating number operations (FLOPs) to run these deep models, bringing a significant challenge to deploy large networks on devices with limited resources (e.g., mobile phone, robot, drone). Thus, the huge storage and the expensive computational costs have become significant problems to hinder practical applications of deep CNNs in complex real-world scenarios.

Neural network compression (Renda et al., 2020; Xu et al., 2020) has been proposed to accelerate the deep CNNs computation. Network pruning is one of the most intuitive methods to create a small-scale network by reducing redundant and non-informative weights (Li et al., 2016; Yang et al., 2017). The critical point in network pruning is finding a proper metric to measure the importance of the pruned parts. One solution is deleting the weights with small absolute values (Liu et al., 2017) under the presumption that the smaller value of a weight parameter is, the less impact it has on the final result. But this intuitive assumption has been proved invalid in some cases (Ye et al., 2018). On the other hand, many other pruning algorithms have been developed, such as judging the influence of parameter clipping on training loss (Molchanov et al., 2016) or the reconstruction errors of feature outputs (He et al., 2017). However, such algorithms mainly rely on human expert knowledge and hand-crafted pruning rules.

In addition, prevailing methods usually ignore the variety of channels among layers (He et al., 2018, 2019). The candidates of sub-networks are chosen according to various evaluation criteria with the pre-defined pruning ratio (PR) for each layer or block. In this case, no matter which specific channels are pruned, the compressed network architecture remains the same. As mentioned in Gu et al. (2018) and He et al. (2020), the channels of different layers have various functions. Thus, the truly informative (or discriminative) channels might be wrongly removed if the PR is fixed (Yang et al., 2018; Liu et al., 2019), resulting in a decrease in the test accuracy of the pruned network. Furthermore, these manually-set pruning parameters may be the sub-optimal trade-off between the model size and prune accuracy.

Recently, automatic pruning algorithms with neural architecture search (NAS) approaches (Chen et al., 2021; Jia et al., 2021; Liang et al., 2021; Wang et al., 2021a; Xu et al., 2021; Yang et al., 2021) are identified as a promising way to automate network compression. It casts the network pruning problem into the NAS framework, i.e., the search space of NAS is the parameters of the pre-trained network to be pruned. A typical NAS-based pruning model (Dong and Yang, 2019; Jiahui and S., 2019; Liu et al., 2019) explores the potential sub-network architectures from the pre-trained network. Then the intermediate compressed model is evaluated and fine-tuned sequentially to construct the final output. However, prevailing NAS-based algorithms (Jiahui and S., 2019; Liu et al., 2019) usually simplify the network at a coarse-grained level while ignoring the critical specific channels.

This study proposes a novel NAS-based pruning model named GWCS. It can dynamically prune a pre-trained network at the channel level while maintaining the model accuracy. First, we formulate the network compression task as a combinatorial optimization problem. Specifically, we genetically encode each channel in the pre-trained network and prune it adaptively using a dynamic selection operation in multiple stages with a wavelet channel aggregation (WCA) based fitness function. As shown in **Figure 1**, our dynamic network pruning model yields much higher prune accuracy than the hand-crafted pruning method for ResNet series models on CIFIA-100. Notably, our model even

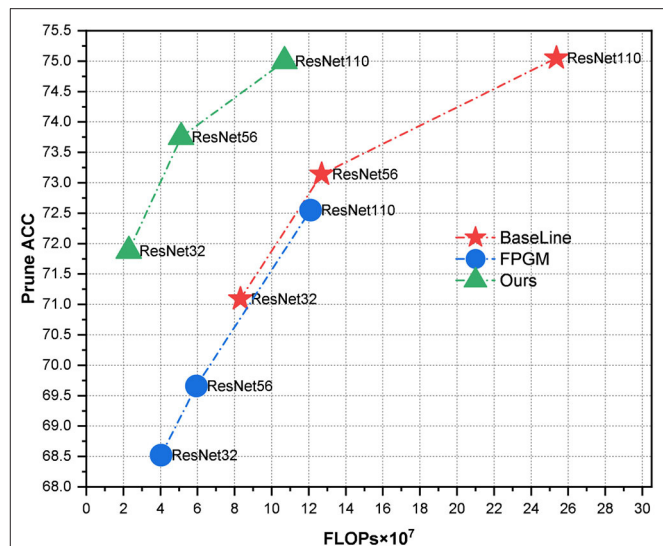


FIGURE 1 | We compare classification accuracy vs. computational complexity (FLOPs) with ResNet series models on CIFIA-100. Our pruning method with a more flexible optimization procedure obtains more promising results than filter pruning algorithm based on geometric median (FPGM) (He et al., 2019) with the fixed pruning rate.

accelerates ResNet32 and ResNet56 three times, along with the improved classification results.

This study makes innovative contributions in the automatic network pruning process for image classification as follows:

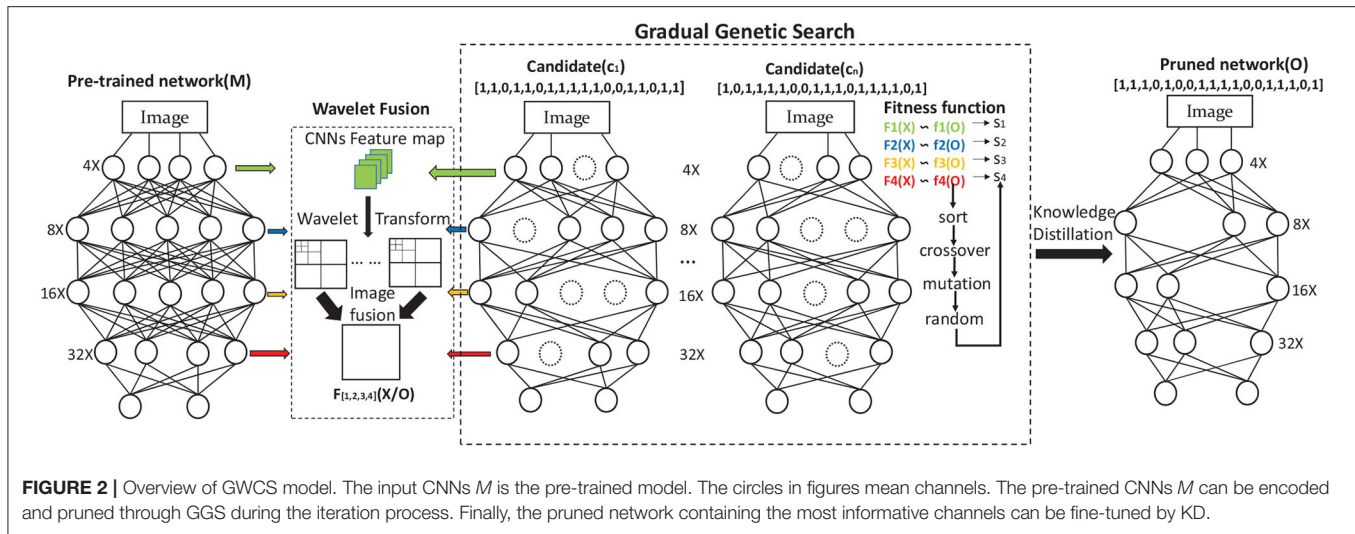
- (1) We develop a GWCS pipeline to prune the pre-trained network dynamically. It models the channel-wise network pruning task as a multi-stage genetic optimization procedure.
- (2) We introduce a WCA based fitness function to evaluate and exploit the most informative channels.
- (3) Extensive experiments are conducted to demonstrate the effectiveness of the proposed dynamic channel pruning model on some popular benchmark datasets, including CIFAR-10, CIFAR-100, and ImageNet. Our GWCS outperforms the tested state-of-the-art models regarding pruning accuracy and network compression rate.

The rest of the study is organized as follows: Section 2 presents the proposed genetic wavelet channel search scheme. The experimental results are provided in section 3, following the discussion in section 4.

2. METHODS

2.1. Overview of GWCS

This study aims to remove the redundant channels from the pre-trained network M for generating a pruned output O with reliable classification results. We approach the problem of compressing the network with flexible pruning layers as a genetic search framework. It contains three steps which are shown in **Figure 2**: (1) Training a large CNNs (the pre-trained network M), (2) Using GWCS to prune the channels in pre-trained network M layer by layer, (3) Knowledge distilling (KD) the pruned network to recover the model accuracy. In the search process, the most



critical part is to effectively and flexibly remove the inadequate channels in the pre-trained network M without significantly compromising accuracy. Next, we will introduce our GWCS model to address this problem.

2.2. Genetic Wavelet Channel Search

2.2.1. Gradual Genetic Search (GGS)

Initialization. Our proposed GWCS strategy is an iterative process in which the initial network is made gradually better as a group called a population. At first, all the channels of pre-trained network M can be encoded into random binary genotypes to generate the population \mathcal{A} , in which we denote the candidate compressed network $\mathbf{X}_i \in \mathcal{A}$ standing for the i th instance in \mathcal{A} :

$$\mathbf{X}_i = \{c_i^1, c_i^2, \dots, c_i^N\} \quad (1)$$

where $i \in \{1, 2, \dots, NP\}$, and NP and N is the total number of population individuals. N is the total number of the channels in \mathbf{X}_i , and c_i^j means the j -th channel code of \mathbf{X}_i , while $c_i^j = 0$ represents the corresponding channel to be pruned; otherwise, $c_i^j = 1$ means the channel will be reserved.

All the individuals of \mathbf{X}_i are grouped into the population set \mathcal{A} , defined in Equation (2):

$$\mathcal{A} = \begin{cases} X_1 = [1, 0, 1, 1, 0, 1, 0, \dots, 0, 1, 1, 1] \\ X_2 = [0, 1, 1, 0, 0, 1, 0, \dots, 0, 1, 0, 1] \\ \vdots \\ X_{NP} = [1, 0, 1, 0, 0, 1, 1, \dots, 0, 0, 1, 0] \end{cases} \quad (2)$$

channels' code

Gradual Genetic Search. Searching the entire space with millions of channels in \mathbf{X}_i is intractable. In this study, we proposed a new strategy, named GGS, to examine the valuable channels hierarchically, rather than directly inspecting all the c_i^j in \mathbf{X}_i as a whole.

The success of CNN mainly attributes to its hierarchical structures from the concrete level to the abstract level, i.e., the convolutions in shallow layers extract coarse features such as color and edges. In contrast, those in deep layers acquire more abstract or semantic features related to the concept of category. The proposed GGS is consistent with this theory. As shown in **Figure 2**, we divide the neural network searching process into multiple stages according to the down-sampling sizes in CNNs, i.e., we can divide the whole search space into several sub-spaces with multi-scale feature sizes down-sampling from $4\times$ to $32\times$, e.g., an individual network \mathbf{X}_i can also be divided as:

$$\mathbf{X}_i = [\mathbf{X}_i^{(1)}, \mathbf{X}_i^{(2)}, \mathbf{X}_i^{(3)}, \mathbf{X}_i^{(4)}] \quad (3)$$

where the sub-network $\mathbf{X}_i^{(st)} \in \mathbf{X}_i$ and $st \in [1, 4]$. Note that the maximum iteration number of $T^{(st)}$ is set variously in each stage due to the total number of channels in $\mathbf{X}_i^{(st)}$ is different.

Crossover. In every iteration, we can produce a new group of offspring (i.e., new codes of the pruned network) using variations through the crossover operator. First, we randomly selected two chromosomes as parents, e.g., $\mathbf{X}_{r1}^{(st)}$ and $\mathbf{X}_{r2}^{(st)}$ are chosen to exchange channel bits at certain points. After that, a new offspring $\mathbf{X}_{cr}^{(st)}$ can be generated by using the multipoint crossing strategy based on the selected parents $\mathbf{X}_{r1}^{(st)}$ and $\mathbf{X}_{r2}^{(st)}$, which can be formulated as:

$$\mathbf{X}_{cr}^{(st)} = \mathbf{G} \circ (\mathbf{X}_{r1}^{(st)}) + |1 - \mathbf{G}| \circ (\mathbf{X}_{r2}^{(st)}) \quad (4)$$

where \mathbf{G} is a random vector of bits (0 or 1) to disrupt the codes of selected parents.

Mutation. The mutation operator is applied to further enhance the diversity of offspring and the ability of the model to escape from local optimization. We use the binary mutation strategy by flipping the bit randomly in $\mathbf{X}_{cr}^{(st)}$ to produce a new individual $\mathbf{X}_m^{(st)}$, defined as follows.

$$\mathbf{X}_m^{(st)} = H(\mathbf{X}_{cr}^{(st)}) \quad (5)$$

where $H(\cdot)$ means that a total of $p_m\%$ of binary codes in randomly selected channels will be flipped. In our study, the bit flip in the genotype space could potentially create a different pruned network.

Selection. Every candidate of the lightweight network in the population (including both parents and offspring) will be evaluated for survival and reproduction (becoming a parent) in each iteration. For each network at stage $X_i^{(st)}$, the top K individuals with the highest fitness are selected based on the Roulette Wheel algorithm with a survival probability of $p_s\%$, to form the next generation. In this study, the $P_i^{(st)}$ of each $X_i^{(st)}$ at the st_{th} stage can be denoted as follows:

$$P_i^{(st)} = \frac{Ft(X_i^{(st)})}{\sum_{i=1}^{NP} Ft(X_i^{(st)})} \quad (6)$$

where the $Ft(\cdot)$ is the fitness function, which determines whether a potential pruned network could survive and will be introduced in detail below.

2.2.2. Fitness Function

In GWCS, we aim to find the best individual network after removing the redundancy channels through the fitness evaluation. Considering that the most informative channels should have the minimal reconstruction error of feature maps, our fitness function $Ft(\cdot)$ is designed based on the similarity of feature maps between the pre-trained network and pruned network. The output Ft can be used as a pruning criterion to identify the best-pruned networks.

Wavelet transform has been successfully applied in image processing. Its primary purpose is to extract the specific properties of the image with the wavelet basis function, which can be formulated as:

$$F^* = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} F * \psi\left(\frac{t-\tau}{a}\right) dt \quad (7)$$

where a is the scale that controls the stretching of the wavelet, and τ is the translation that affects the translation of the wavelet. F represents the feature maps of the last CNN layer in the input network. In our GWCS algorithm, we adapt the Haar wavelet function (Porwik and Lisowska, 2005) to extract the frequency features due to its simplicity and effectiveness.

To calculate the similarity between the networks with different sizes of features maps (i.e., the total number of channels of the network is variable after the dynamic pruning), we aggregate all the wavelet feature maps into one vector, which is formulated in Equation (8).

$$F^* = \max(F_{HH}^*) \oplus \text{Avg}(F_{LL}^*) \quad (8)$$

where HH and LL represent high-frequency and low-frequency information. \oplus is the element-wise addition. The final fused feature vectors F^* are generated by the maximum values of HH and the average values of LL using \oplus operation. Comparing to conventional aggregate functions, including global average pooling (GAP) or global max pooling (GMP),

Algorithm 1: Algorithm of the gradual genetic search.

Input: The original network M

Output: A pruned network O

- 1: Randomly initialize the binary codes in networks $\{X_1, X_2, \dots, X_{NP}\}$ to form the initial population \mathcal{A}_0 by Equation (2).
- 2: Set the maximum iteration number $T=[T^1, \dots, T^4]$ for each searching stage.
- 3: **for** $st = 1$ to 4 **do**
- 4: **while** t in T^{st} **do**
- 5: Calculate $F^{*(st)}$ and $\{f_i^{*(st)}\}_{i=1}^{NP}$ by Equation (7, 8).
- 6: Calculate the fitness $\{s_i^{(st)}\}_{i=1}^{NP}$ by Equation (9).
- 7: Select top K individuals from $\mathcal{A}_t^{(st)}$ by Eq. (6).
- 8: Crossover and mutate the top K individuals using (Equation 4, 5).
- 9: Generate $\mathcal{A}_{t+1}^{(st)}$
- 10: Update $t = t + 1$
- 11: **end while**
- 12: **end for**
- 13: Select the best individual as the pruned network O
- 14: **return** O

more rich information contained in both high- and low-frequency components are more helpful for improving the classification (Qin et al., 2020), i.e., it can further boost the feature similarity estimation.

As is illustrated in **Figure 3**, both $F^{(st)}$ and $f_i^{(st)}$ can be transformed and aggregated by wavelet operation using (Equations 7, 8), denoted as $F^{*(st)}$ and $f_i^{*(st)}$, respectively. We can obtain the similarity $s_i^{(st)}$ based on the cosine distance, which can be formulated as:

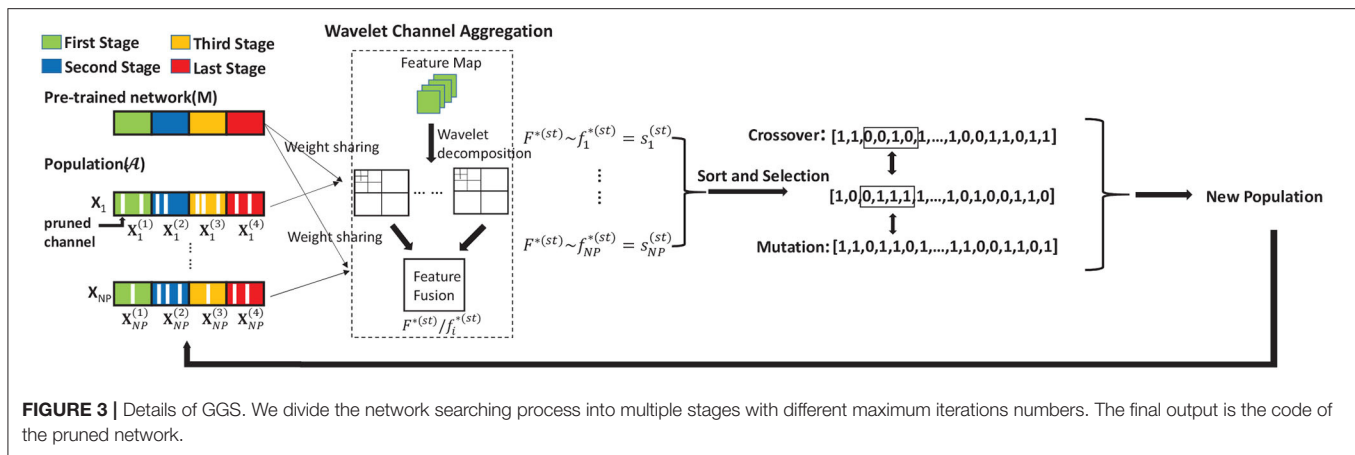
$$s_i^{(st)} = \frac{F^{*(st)} \cdot f_i^{*(st)}}{\|F^{*(st)}\| \|f_i^{*(st)}\|} \quad (9)$$

The best-pruned network O can be achieved by selecting the best individual with the highest fitness from population \mathcal{A} after maximum iterations. The detailed steps of GGS are shown in **Algorithm 1**, from which we can observe that the time complexity of our GWCS is $O[st * T^{st} * (NP * N * \text{size}(F) + NP * N + N)]$.

2.3. Knowledge Distillation

The fine-tuning (FT) process is crucial for recovering the original performance (Dong and Yang, 2019). In this study, knowledge distillation (KD) (Hinton et al., 2015) is applied to improve the performance of the pruned network. In our model, the pruned network derives from the pre-trained network. Thus, we take the pre-trained network as the teacher network and transfer its knowledge into the pruned network (i.e., the student network).

In the classification task with CNNs, the softmax layer is adopted as the classifier. The softmax output is a one-hot vector, i.e., the classification result is the label with the largest value. However, such logit outputs contain very little information as



we cannot learn the relationship between classes except the prediction labels. The output results can be further softened as:

$$q_k = \frac{\exp(z_k/T)}{\sum_j \exp(z_j/T)} \quad (10)$$

where z is the softmax vector from the pre-trained network. T stands for temperature. When T tends to zero, the output q_k is degraded into the one-hot vector. The pruned network can take the soft target output q_k as the training loss to transfer the knowledge from the original unpruned network.

Following the prevailing study in Dong and Yang (2019), we use the middle layer transfer of KD to optimize the searched network *via* (Equation 11).

$$L = \rho_1 L_1 + \dots + \rho_n L_n + (1 - \rho_1 - \dots - \rho_n) L_{hard} \quad (11)$$

where L is the total loss function of KD and L_n is the loss function of each training stage.

3. RESULTS

3.1. Experimental Setting

3.1.1. Datasets

In our experiment, we evaluated the tested models on CIFAR-10, CIFAR-100 (Krizhevsky, 2009), and ImageNet (Russakovsky et al., 2015) for image classification tasks. CIFAR-10 consists of 50 k training and 10 k testing 32×32 images in 10 classes. Similar to CIFAR-10, the CIFAR-100 dataset has 100 categories. There are 500 training images and 100 verification images for each class. The ImageNet dataset (ISLVR 2012) (Russakovsky et al., 2015) is a large visual database collected from the real world. it consists of 1,281,167 training images and 50,000 validation images in 1,000 classes. Data augmentation techniques, including random resize, crop, brightness changing, and horizontal flipping are also employed to improve accuracy.

3.1.2. Implementation Details

Following the previous studies (He et al., 2018, 2019; Dong and Yang, 2019), ResNet series networks (He et al., 2016), and VGGNet-16 (Simonyan and Zisserman, 2015) are chosen as

the baseline networks in our pruning experiment. We trained them using the standard stochastic gradient descent (SGD) optimization with batch size 128. Our initial learning rate is set to 0.1, which is gradually reduced with a weight decay of 0.0005. For CIFAR-10, we train the Resnet for 150 epochs and train the VGGNet-16 for 200 epochs, respectively. For CIFAR-100, we train the Resnet for 200 epochs and train the VGGNet-16 for 300 epochs, respectively. For ImageNet, we train the Resnet for 150 epochs and train the VGGNet-16 for 300 epochs, respectively. All models are implemented on dual NVIDIA GTX1080ti GPUs in PyTorch.

3.1.3. Specific Searching and Training Setting

We take the unpruned network as the initial input for our algorithm in the process of pruning. First, the codes of 50 individuals (which have the same number of channels as the unpruned network) are randomly initialized. Each individual will be evaluated as a candidate pruned network. Then, we search the optimal channels using GGS in multiple stages, i.e., we divided the searching procedure into four stages with the maximum number of iterations in [10, 10, 5, 5]. The top 20 individuals are chosen for crossover and mutating based on the fitness values. Specifically, in crossover operation, two individuals are randomly selected for exchanging 50% of codes with each other. In mutation, 10% of codes of individuals are chosen for mutating, i.e., 0 and 1 interchange. Finally, a new population can be generated by selecting the top 30 individuals for the next iteration.

3.2. Comparison With State-of-the-Art Methods

We compare several state-of-the-art network pruning models published in most recent years in our experiments.

Soft Filter Pruning (SFP): He et al. (2018) proposes a SFP method. After training the model at each epoch, the L2 norm of the corresponding channel is calculated. Meanwhile, the lower-ranked channel is set to zero according to a manual pruning rate. Still, the pruned ones will also participate in the next round of iterations instead of deleting them directly.

TABLE 1 | Comparison results on CIFAR-10 with ResNet-32, 56, and 110.

Network	Method	Baseline Acc (%)	Prune Acc (%)	Drop (%)	FLOPs(PR)
ResNet-32	FPGM	92.63	92.31	<u>0.32%</u>	4.03E7(41.5%)
	SFP	92.63	92.08	0.55	4.03E7(41.5%)
	TAS	93.88	92.92	0.96	3.78E7(45.4%)
	LFPC	92.63	92.12	0.51	3.27E7(52.6%)
	ManiDP	92.66	92.15	0.51	2.54E7(63.2%)
	Ours	93.08	92.97	0.11	1.82E7(73.6%)
ResNet-56	HRank	94.46	93.52	0.94	6.58E7(37.9%)
	JST	94.41	93.68	0.73	6.32E7(49.7%)
	FPGM	93.59	92.89	<u>0.70</u>	5.94E7(52.6%)
	SFP	93.59	92.26	1.33	5.94E7(52.6%)
	TAS	94.46	93.69	0.77	5.95E7(52.7%)
	Ours	94.23	93.75	0.48	5.05E7(60.3%)
ResNet-110	SFP	93.67	92.97	0.70	1.21E8(52.3%)
	TAS	94.97	94.33	0.64	1.19E8(53.0%)
	LFPC	93.68	93.07	<u>0.61</u>	1.01E8(60.3%)
	Ours	95.03	94.78	0.25	1.12E8(56.0%)

The best results are highlighted in bold and the second-best results are underlined. "Drop" means accuracy drop, "FLOPs (PR)" represents FLOPs of the compressed model with the corresponding pruning ratio (PR).

Discrimination aware channel pruning (DCP): Zhuang et al. (2018) implements a pruning method called DCP, which adds discriminative losses into the network and obtains pruned network after a greedy algorithm for channel selection.

Genetic channel pruning (GCP): Hu et al. (2018) also uses a genetic algorithm to code and prune the network. However, the GCP searches the entire pre-trained network as a whole and prunes it with a group of manually assigned compression rates and the layer-wise error is estimated with the Hessian matrix.

Filter pruning algorithm based on geometric median (FPGM): He et al. (2019) proposes a filter pruning algorithm based on geometric median. FPGM deletes the redundant filters instead of the relatively less important ones with a manual setting of pruning rate.

Transformable architecture search (TAS): Dong and Yang (2019) proposes a TAS approach for compressing CNNs by channel-wise probability distribution and knowledge transfer. TAS aimed to search for the appropriate width and depth of the pruned network.

High-rank pruning (HRank): Lin et al. (2020) reveals a rule of CNNs even if the input image is different, there is always a large rank in the same part of the feature graph. The results suggest that the latent rank information is essential in the network so that the redundancy weights can be compressed with low-rank feature maps.

Joint search-and-training (JST): Lu et al. (2020) implements an automatic search algorithm by training and pruning simultaneously. It saves the pre-training time in the automatic pruning algorithm with competitive classification accuracy.

Discrete model compression (DMC): Gao et al. (2020) proposes a discrete compression model, which attaches a gate for each channel to control whether the channel is opened or

not. Then the pruned network is obtained by gradient descent to optimize the gate parameters.

Learning filter pruning criteria (LFPC): He et al. (2020) introduces a LFPC to select a set of suitable measures for different layers adaptively. LFPC evaluates the importance of the filters based on the proposed differentiable criteria sampler with Gumbel-softmax.

Structural redundancy reduction with graph redundancy (SRR-GR): Wang et al. (2021b) assumes that the performance of the pruning filter in the more redundant layer is better than that of pruning the least important filter in all layers. Based on this assumption, this method establishes an undirected graph for each layer, in which each vertex represents a filter and edge denotes the distance between filter weights. The quotient space size and covering number are calculated according to the redundancy rates of each graph.

Manifold regularized dynamic pruning (ManiDP): Tang et al. (2021) develops a (ManiDP) strategy that identifies the complexity and feature similarity of the training data set. The network is pruned dynamically by exploiting the manifold regularization, and the appropriate sub-network is allocated for each instance.

3.3. Main Results With ResNet

3.3.1. Results on CIFAR-10 and CIFAR-100

The pruning result of ResNet series networks on CIFAR-10 and CIFAR-100 are shown in Tables 1, 2. Among all the tested pruning algorithms, our GWCS model consistently reduces the largest number of channels to generate the minimum FLOPs among all the tested pruning models. Notably, our model produced the highest pruning rates of 73.6 and 73.1% by pruning ResNet-32 on CIFAR-10 and CIFAR-100, respectively.

TABLE 2 | Comparison results on CIFAR-100 with ResNet-32, 56, and 110.

Network	Method	Baseline Acc (%)	Prune Acc (%)	Drop (%)	FLOPs(PR)
ResNet-32	FPGM	69.77	68.52	1.25	4.03E7(41.5%)
	TAS	70.62	71.74	-1.12	3.80E7(45.0%)
	Ours	71.09	71.88	<u>-0.79</u>	2.29E7(73.1%)
ResNet-56	FPGM	71.41	69.66	1.75	5.94E7(52.6%)
	JST	72.89	70.63	2.26	6.72E7(51.1%)
	TAS	73.18	72.25	<u>0.93</u>	6.12E7(51.3%)
	Ours	73.14	73.75	-0.61	5.12E7(59.7%)
ResNet-110	FPGM	74.14	72.55	<u>1.59</u>	1.21E8(52.3%)
	JST	74.42	72.26	2.16	1.08E8(58.0%)
	TAS	75.06	73.16	1.90	1.20E8(52.6%)
	Ours	75.05	75.00	0.05	1.07E8(58.2%)

The best results are highlighted in bold and the second-best results are underlined.

TABLE 3 | Comparison results on ImageNet with ResNet-50 and ResNet-101.

Network	Method	Top-1 Prune Acc (%)	Top-5 Prune Acc (%)	Top-1 Drop (%)	Top-5 Drop (%)	FLOPs(PR)
ResNet-50	HRank	74.98	92.33	2.48	1.22	2.62E9(40.8%)
	TAS	76.20	93.07	1.26	0.48	2.31E9(43.5%)
	JST	75.51	92.43	<u>1.01</u>	0.66	2.25E9(44.9%)
	FPGM	74.83	92.32	1.32	0.55	2.58E9(53.5%)
	DMC	75.35	92.49	0.80	<u>0.38</u>	2.01E9(55.0%)
	SRR-GR	75.76	92.67	1.02	0.51	2.01E9(55.1%)
	DCP	74.95	92.32	1.06	0.61	1.99E9(55.6%)
	Our	76.64	93.78	1.09	0.36	1.83E9(59.1%)
ResNet-101	SFP	77.51	93.71	-0.14	-0.20	6.43E9(30.0%)
	FPGM	77.37	93.56	0.05	<u>0.00</u>	6.43E9(30.0%)
	Our	77.65	93.65	<u>-0.13</u>	0.33	4.36E9(58.7%)

The best results are highlighted in bold, and the second-best results are underlined.

It saved more than half of GPU computational cost compared to FPGM. Turning to the pruning accuracy, our model achieves the lowest accuracy drops by pruning the ResNet networks on CIFAR-10 and obtains the best prune accuracy with ResNet-56 and ResNet-110 on CIFAR-100. For example, when pruning ResNet-110, our model achieves the highest pruning accuracy of 75%, outperforming the second-best model (FPGM) by more than 1.54% in terms of accuracy drop, along with much fewer computations. Note that the proposed GWCS model also achieves a very close result (only 0.05% of the drop of accuracy) to the original ResNet-110 with the highest FLOPs reduction (nearly 2.39× compression rate). These results suggest that the proposed GWCS is an effective and reliable network pruning model, achieving a better trade-off between pruning accuracy and model size.

3.3.2. Results on ImageNet

The effectiveness of GWCS is further validated by the transferred performance on ImageNet using ResNet-50 and ResNet-101. As shown in **Table 3**, The proposed approach can produce a promising test accuracy (0.36 and 0.33% Top-5 accuracy drop

on ResNet-50 and ResNet-101, respectively) with the largest compression rates. For example, GWCS outperforms TAS by 0.12% Top-5 accuracy drop with a significant FLOPs reduction (less than nearly 15.6%). When pruning ResNet-101, our model obtains a comparable test accuracy rate but removes nearly 2× FLOPs than the SFP model.

We also visualize the pruned channels of ResNet-50 on ImageNet in **Figure 4**. As can be seen from **Figure 4**, the pruning rates are various in each layer, which could be more suitable for channel searching as the information contained in layers may be different, and the truly useful channels can be preserved with a flexible pruning strategy.

3.4. Main Results With VGGNet

In **Table 4**, we show the comparison results in terms of Prune ACC and FLOPs on CIFAR-10 and CIFAR-100 with VGGNet-16. Among all the tested models, our proposed GWCS still achieves the highest pruning rate and yields the lowest FLOPs. In particular, we can see that the pruning reductions of our model are 64.18 and 66.61% on CIFAR-10 and CIFAR-100, respectively, which are much higher than HRank, GCP, and JST. Furthermore,

our model produces a comparable accuracy with much fewer FLOPs. For instance, compared with GCP, our model obtains a very close test accuracy (0.58 vs. 0.20% accuracy drop) but prunes more than $1.8\times$ channels on CIFAR-100.

4. DISCUSSION

4.1. GGS vs. Overall Genetic Search

In our model, we proposed a hierarchical search method named GGS algorithm to prune the network in multiple stages instead of searching the whole space of all channels [i.e., Overall Genetic Search (OGS)] at each iteration. We conduct the ablation experiment for studying the effect of GGS comparing to the

overall search method on the CIFAR-10 dataset using ResNet-32. The channels of the pre-trained network are divided into four stages by using GGS, and the maximal number of iterations in each stage is set to 10, 10, 5, and 5, respectively. Thus, the total number of iterations in the pruning process is 30, which

TABLE 5 | Comparison of gradual search and overall search on CIFAR-10 with ResNet-32.

Method	Prune Acc(Drop)	FLOPs(PR)
Overall Genetic Search	92.19%(0.89%)	2.31E7(72.7%)
Gradual Genetic Search	92.97%(0.11%)	2.23E7(73.6%)

The best results are highlighted in bold, and the second-best results are underline.

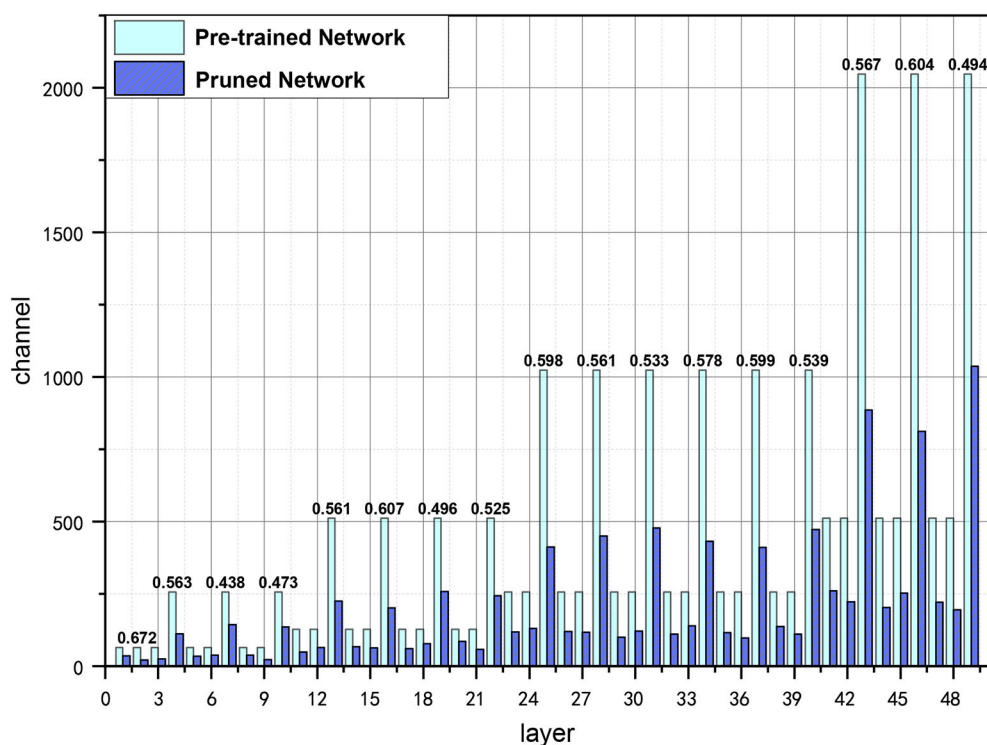


FIGURE 4 | Visualization of pruned channels in ResNet-50 with GWCS on ImageNet, where *layer* on the x-axis represents the number of layers in ResNet-50, *channel* means the number of channels in each layer. The bars in light blue indicate the number of channels in the pre-trained network. The purple ones indicate the number of channels after pruning. The PR are shown on the top of the bars.

TABLE 4 | Comparing our model and other methods with VGGNet-16 on CIFAR-10 and CIFAR-100.

Datasets	Method	Baseline Acc (%)	Prune Acc (%)	Drop (%)	FLOPs(PR)
CIFAR-10	GCP	92.71	92.74	-0.03	2.74E8(52.0%)
	HRank	93.96	93.43	0.53	2.71E8(53.5%)
	Ours	94.13	93.76	<u>0.37</u>	2.29E8(64.18%)
CIFAR-100	GCP	72.21	72.01	0.20	3.82E8(37.0%)
	JST	75.75	74.63	1.12	3.22E8(45.0%)
	Ours	73.75	73.17	<u>0.58</u>	2.21E8(66.61%)

We highlight the best and second-best results in bold face and underline, respectively. "Drop" means accuracy drop, "FLOPs (PR)" means FLOPs and pruning rate.

is also set as the maximum iteration number for OGS. The same FT operation with KD is applied in the OGS method to recover the accuracy of the pruned network. We can inform from **Table 5** that, GGS generates a more accurate classification result with more than 0.9% FLOPs reduction, compared to the OGS method when pruning ResNet-32 on CIFAR-10, suggesting that the proposed GGS proves a more optimal solution for identifying the critical channels in a large search space.

4.2. Effect of WCA

Wavelet channel aggregation in the proposed GWCS model is used to evaluate the performance of the pruned network based on the fused wavelet transformed features. Comparing to conventional feature aggregation methods used in deep CNNs, including GAP, GMP, and GAP+GMP, we investigated the utility of the fitness function based on the WCA method on CIFAR-10 and CIFAR-100 in terms of prune accuracy and FLOPs. The comparison results are reported in **Table 6**. We observed that GAP prunes much more channels while producing much worse

accuracy values. However, WCA achieves the best classification accuracy with the comparable FLOPs. As mentioned in Qin et al. (2020), GAP extracts the low-frequency information (e.g., the contour of an object) of the image, while GMP takes the high-frequency information (e.g., edge or texture). Nevertheless, both the contour and texture features are essential for image classification. Therefore, considering both high- and low-frequency information in our WCA method could help identify the best-compressed network.

As is shown in **Figure 5**, the feature maps of the pruned network with selected channels based on the WCA contains more information for categorization than those of the selected ones using other conventional feature aggregation methods. The results further prove that the proposed WCA can adequately choose the channels with the most representational power for the network.

4.3. Effect of FT Strategies

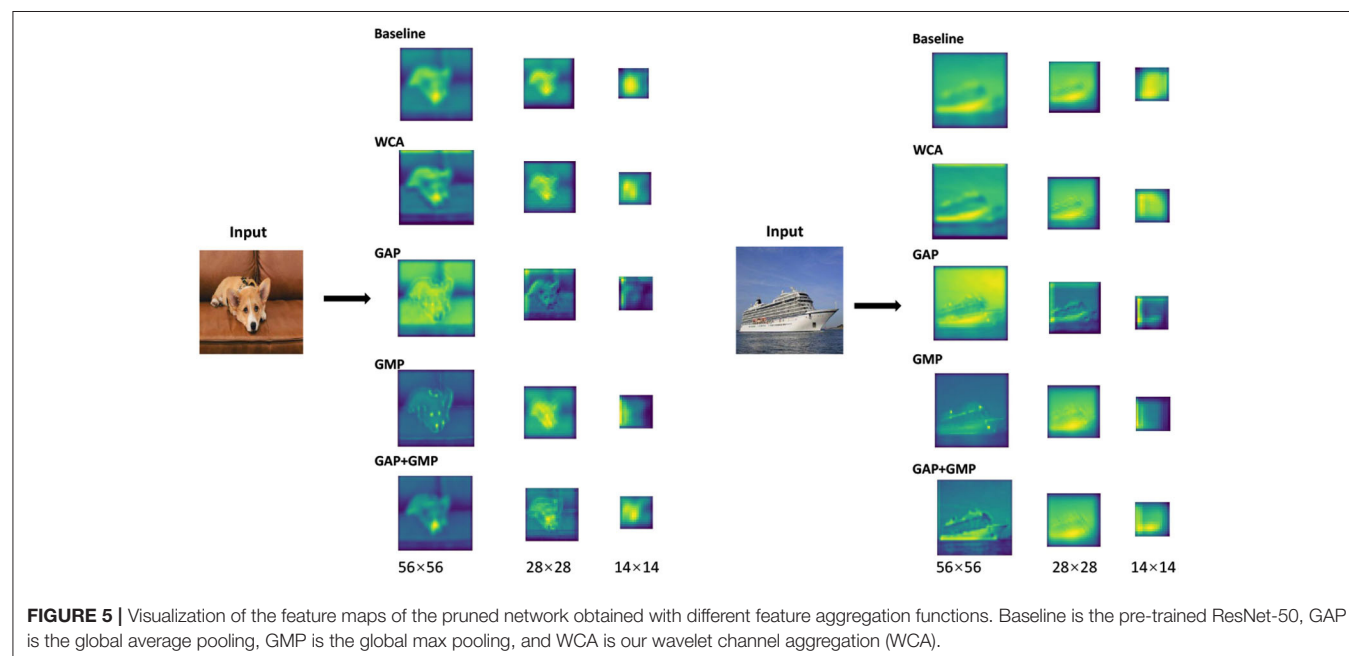
Knowledge Distillation (KD) is the last step in our GWCS model for regaining the lost performance. In this sub-section, we try to investigate the effectiveness of the GWCS for reducing the redundancy channels instead of relying on KD technology alone. Thus, two ablation experiments, i.e., the GWCS based on KD (GWCS+KD), are conducted to compare with: (1) the conventional FT technique by retraining the compact network from scratch, named GWCS+FT. (2) The channels pruning strategy with random selection and KD, named RS+KD.

As shown in **Figure 6**, we can observe that KD and FT result in very similar classification accuracy but are various in speeds of convergence. Specifically, GWCS+KD can always reach the highest accuracy after about 2,000 iterations, while FT needs more than 3,500 iterations. This suggests that GWCS+KD is more efficient for network pruning.

TABLE 6 | Comparison of different feature aggregation methods applied in fitness functions on CIFAR-10 with ResNet-32.

Fitness function	Prune Acc(Drop)	FLOPs(PR)
GAP	91.41%(1.67%)	2.15E7(75.4%)
GMP	91.25%(1.83%)	2.21E7(74.0%)
GAP+GMP	91.84%(1.24%)	2.23E7(73.5%)
WCA	92.97%(0.11%)	2.23E7(73.6%)

The best results are highlighted in bold, and the second-best results are underlined. GAP is global average pooling, GMP is global max pooling, WCA is our proposed wavelet channel aggregation method.



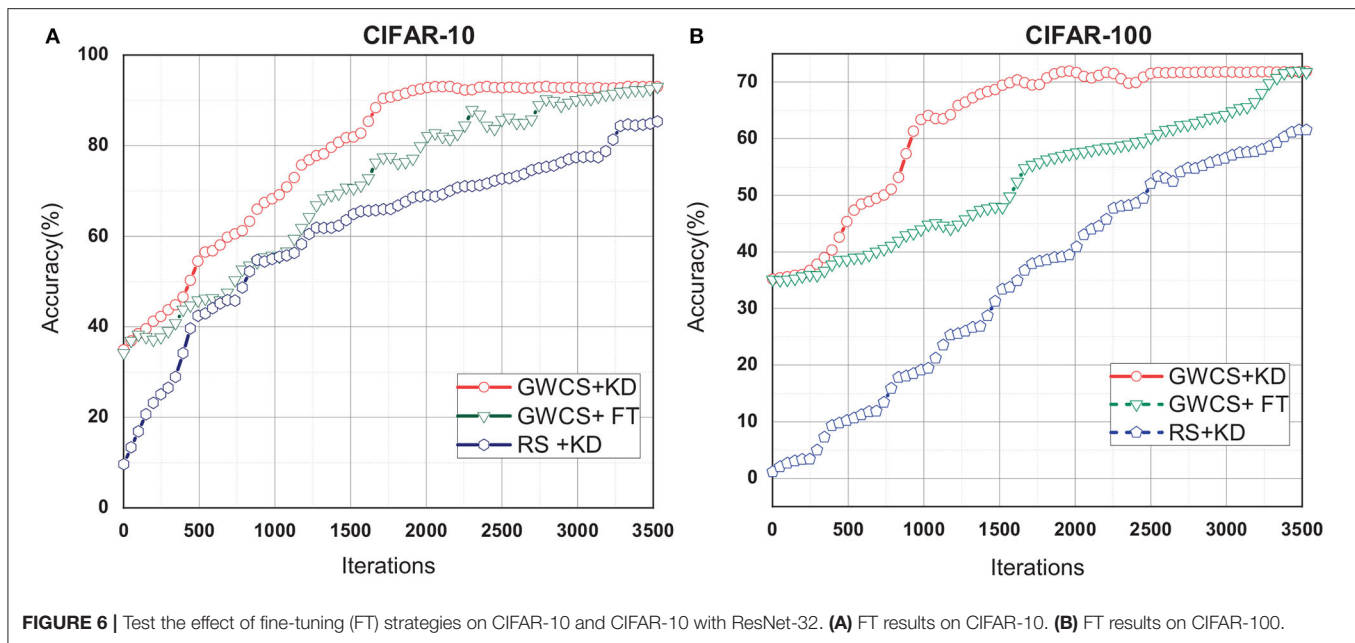


FIGURE 6 | Test the effect of fine-tuning (FT) strategies on CIFAR-10 and CIFAR-100 with ResNet-32. **(A)** FT results on CIFAR-10. **(B)** FT results on CIFAR-100.

However, RS+KD does not outperform GWCS+KD and GWCS+FT in the iterations. All of these demonstrate that our GWCS algorithm with KD indeed obtains a promise pruning result.

To summarize, in this study, we propose a novel genetic NAS-based network pruning method to automate the channel-wise network pruning. The main idea is to dynamically select the most informative channels in each layer from the pre-trained network using a multi-stage genetic optimization algorithm. Furthermore, we presented a novel fitness function based on the WCA to evaluate the performance of the pruned network. We conduct large-scale experiments using several public datasets to verify the performance of tested pruning models. The results demonstrate that the proposed GWCS model achieves a more compressed network with a promise classification accuracy than other tested SOTA pruning methods. In the future, we will further evaluate the effectiveness of our model on some mobile devices and employ the proposed model to compress the CNNs for other tasks, such as object detection or image segmentation.

DATA AVAILABILITY STATEMENT

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

AUTHOR CONTRIBUTIONS

LC, SG, and XS implemented and optimized the methods and wrote the manuscript. LC, XS, and MS designed the experiment and algorithm. All authors contributed to the article and approved the submitted version.

FUNDING

Publication costs are funded by the National Nature Science Foundation of China under grant nos. 61902370, 61802360, and in part by the Chongqing Research Program of Technology Innovation and Application under grants cstc2019jscx-zdztzxX0019.

REFERENCES

- Chen, Y., Guo, Y., Chen, Q., Li, M., Zeng, W., Wang, Y., et al. (2021). "Contrastive neural architecture search with neural architecture comparators," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE), 9502–9511.
- Dong, X., and Yang, Y. (2019). "Network pruning via transformable architecture search," in *The Conference on Neural Information Processing Systems (NeurIPS)* (Vancouver, BC), 760–771.
- Gao, S., Huang, F., Pei, J., and Huang, H. (2020). "Discrete model compression with resource constraint for deep neural networks," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Seattle, WA: IEEE), 1899–1908.

- Giménez, M., Palanca, J., and Botti, V. (2020). Semantic-based padding in convolutional neural networks for improving the performance in natural language processing. a case of study in sentiment analysis. *Neurocomputing* 378, 315–323. doi: 10.1016/j.neucom.2019.08.096
- Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., et al. (2018). Recent advances in convolutional neural networks. *Pattern Recognit.* 77, 354–377. doi: 10.1016/j.patcog.2017.10.013
- He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Las Vegas, NV: IEEE), 770–778.
- He, Y., Ding, Y., Liu, P., Zhu, L., Zhang, H., and Yang, Y. (2020). "Learning filter pruning criteria for deep convolutional neural networks acceleration," in *2020*

- IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (Seattle, WA: IEEE), 2006–2015.
- He, Y., Kang, G., Dong, X., Fu, Y., and Yang, Y. (2018). “Soft filter pruning for accelerating deep convolutional neural networks,” in *Proceedings of the 27th International Joint Conference on Artificial Intelligence* (Stockholm), 2234–2240.
- He, Y., Liu, P., Wang, Z., Hu, Z., and Yang, Y. (2019). “Filter pruning via geometric median for deep convolutional neural networks acceleration,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Long Beach, CA: IEEE), 4340–4349.
- He, Y., Zhang, X., and Sun, J. (2017). “Channel pruning for accelerating very deep neural networks,” in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (Venice, FL: IEEE), 1389–1397.
- Hinton, G., Vinyals, O., and Dean, J. (2015). Distilling the knowledge in a neural network. *arXiv arXiv:1503.02531*.
- Hu, Y., Sun, S., Li, J., Wang, X., and Gu, Q. (2018). A novel channel pruning method for deep neural network compression. *arXiv arXiv:1805.11394*.
- Jia, F., Wang, X., Guan, J., Li, H., Qiu, C., and Qi, S. (2021). Arank: Toward specific model pruning via advantage rank for multiple salient objects detection. *Image Vis. Comput.* 111:104192. doi: 10.1016/j.imavis.2021.104192
- Jiahui, Y., and Huang, T. S. (2019). Network slimming by slimmable networks: Towards one-shot architecture search for channel numbers. *CoRR, abs/1903.11728*.
- Krizhevsky, A. (2009). *Learning Multiple Layers of features From Tiny Images*. Technical report.
- Li, H., Kadav, A., Durdanovic, I., Samet, H., and Graf, H. P. (2016). Pruning filters for efficient convnets. *CoRR, abs/1608.08710*.
- Liang, T., Wang, Y., Tang, Z., Hu, G., and Ling, H. (2021). “Opanas: One-shot path aggregation network architecture search for object detection,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE), 10195–10203.
- Lin, M., Ji, R., Wang, Y., Zhang, Y., Zhang, B., Tian, Y., et al. (2020). “Hrank: Filter pruning using high-rank feature map,” in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020*, June 13–19, 2020 (Seattle, WA: IEEE), 1526–1535.
- Liu, Z., Li, J., Shen, Z., Huang, G., Yan, S., and Zhang, C. (2017). “Learning efficient convolutional networks through network slimming,” in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (Venice: IEEE), 2736–2744.
- Liu, Z., Mu, H., Zhang, X., Guo, Z., Yang, X., Cheng, K.-T., et al. (2019). Metapruning: Meta learning for automatic neural network channel pruning. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (Seoul: IEEE), 3295–3304.
- Lu, X., Huang, H., Dong, W., Li, X., and Shi, G. (2020). “Beyond network pruning: a joint search-and-training approach,” in *Twenty-Ninth International Joint Conference on Artificial Intelligence and Seventeenth Pacific Rim International Conference on Artificial Intelligence IJCAI-PRICAI-20* (Yokohama), 2583–2590.
- Molchanov, P., Tyree, S., Karras, T., Aila, T., and Kautz, J. (2016). Pruning convolutional neural networks for resource efficient transfer learning. *CoRR, abs/1611.06440*.
- Porwik, P., and Lisowska, A. (2005). The haar-wavelet transform in digital image processing: Its status and achievements. *Mach. Graph. Vis.* 13, 79–98. doi: 10.1007/978-3-540-25944-2_1
- Qin, Z., Zhang, P., Wu, F., and Li, X. (2020). Fcanet: Frequency channel attention networks. *arXiv arXiv:2012.11879*.
- Renda, A., Frankle, J., and Carbin, M. (2020). “Comparing rewinding and fine-tuning in neural network pruning,” in *8th International Conference on Learning Representations, ICLR 2020*, April 26–30, 2020 (Addis Ababa).
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., et al. (2015). Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* 115, 211–252. doi: 10.1007/s11263-015-0816-y
- Simonyan, K., and Zisserman, A. (2015). “Very deep convolutional networks for large-scale image recognition,” in *3rd International Conference on Learning Representations, ICLR 2015*, May 7–9, 2015 (San Diego, CA).
- Tang, Y., Wang, Y., Xu, Y., Deng, Y., Xu, C., Tao, D., et al. (2021). “Manifold regularized dynamic network pruning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE), 5018–5028.
- Wang, D., Li, M., Gong, C., and Chandra, V. (2021a). Attentivenas: Improving neural architecture search via attentive sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE), 6418–6427.
- Wang, G., Li, W., Ourselin, S., and Vercauteren, T. (2019). Automatic brain tumor segmentation based on cascaded convolutional neural networks with uncertainty estimation. *Front. Comput. Neurosci.* 13:56. doi: 10.3389/fncom.2019.00056
- Wang, Z., Li, C., and Wang, X. (2021b). “Convolutional neural network pruning with structural redundancy reduction,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE), 14913–14922.
- Wu, D., He, Y., Luo, X., and Zhou, M. (2021a). A latent factor analysis-based approach to online sparse streaming feature selection. *IEEE Trans. Syst. Man Cybern. Syst.* 1–15. doi: 10.1109/TSMC.2021.3096065
- Wu, D., Luo, X., Shang, M., He, Y., Wang, G., and Zhou, M. (2021b). A deep latent factor model for high-dimensional and sparse matrices in recommender systems. *IEEE Trans. Syst. Man Cybern. Syst.* 51, 4285–4296. doi: 10.1109/TSMC.2019.2931393
- Xu, X., Feng, W., Jiang, Y., Xie, X., Sun, Z., and Deng, Z. (2020). “Dynamically pruned message passing networks for large-scale knowledge graph reasoning,” in *8th International Conference on Learning Representations, ICLR 2020*, April 26–30, 2020 (Addis Ababa).
- Xu, Y., Wang, Y., Han, K., Tang, Y., Jui, S., Xu, C., et al. (2021). “Renas: Relativistic evaluation of neural architecture search,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE), 4411–4420.
- Yang, T.-J., Chen, Y.-H., and Sze, V. (2017). “Designing energy-efficient convolutional neural networks using energy-aware pruning,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Honolulu, HI), 5687–5695.
- Yang, T.-J., Howard, A., Chen, B., Zhang, X., Go, A., Sandler, M., et al. (2018). “Netadapt: Platform-aware neural network adaptation for mobile applications,” in *Proceedings of the European Conference on Computer Vision (ECCV)* (Munich), 285–300.
- Yang, Z., Wang, Y., Chen, X., Guo, J., Zhang, W., Xu, C., et al. (2021). “Hournas: Extremely fast neural architecture search through an hourglass lens,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE), 10896–10906.
- Ye, J., Lu, X., Lin, Z. L., and Wang, J. Z. (2018). Rethinking the smaller-norm-less-informative assumption in channel pruning of convolution layers. *CoRR, abs/1802.00124*.
- Zhuang, Z., Tan, M., Zhuang, B., Liu, J., Guo, Y., Wu, Q., et al. (2018). “Discrimination-aware channel pruning for deep neural networks,” in *Proceedings of the 32nd International Conference on Neural Information Processing Systems (NIPS)* (Montreal, QC), 883–894.

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

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

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



Finger Gesture Recognition Using Sensing and Classification of Surface Electromyography Signals With High-Precision Wireless Surface Electromyography Sensors

Jianting Fu¹, Shizhou Cao², Linqin Cai² and Lechan Yang^{3*}

¹ Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Chongqing, China, ² School of Automation, Chongqing University of Posts and Telecommunications, Chongqing, China, ³ Department of Soft Engineering, Jinling Institute of Technology, Nanjing, China

OPEN ACCESS

Edited by:

Yujie Li,
Fukuoka University, Japan

Reviewed by:

Dianlong You,
Yanshan University, China
Wu Bin,
Zhengzhou University, China

*Correspondence:

Lechan Yang
yanglc@jit.edu.cn

Received: 04 September 2021

Accepted: 11 October 2021

Published: 11 November 2021

Citation:

Fu J, Cao S, Cai L and Yang L (2021) Finger Gesture Recognition Using Sensing and Classification of Surface Electromyography Signals With High-Precision Wireless Surface Electromyography Sensors. *Front. Comput. Neurosci.* 15:770692. doi: 10.3389/fncom.2021.770692

Finger gesture recognition (FGR) plays a crucial role in achieving, for example, artificial limb control and human-computer interaction. Currently, the most common methods of FGR are visual-based, voice-based, and surface electromyography (EMG)-based ones. Among them, surface EMG-based FGR is very popular and successful because surface EMG is a cumulative bioelectric signal from the surface of the skin that can accurately and intuitively represent the force of the fingers. However, existing surface EMG-based methods still cannot fully satisfy the required recognition accuracy for artificial limb control as the lack of high-precision sensor and high-accurate recognition model. To address this issue, this study proposes a novel FGR model that consists of sensing and classification of surface EMG signals (SC-FGR). In the proposed SC-FGR model, wireless sensors with high-precision surface EMG are first developed for acquiring multichannel surface EMG signals from the forearm. Its resolution is 16 Bits, the sampling rate is 2 kHz, the common-mode rejection ratio (CMRR) is less than 70 dB, and the short-circuit noise (SCN) is less than 1.5 μ V. In addition, a convolution neural network (CNN)-based classification algorithm is proposed to achieve FGR based on acquired surface EMG signals. The CNN is trained on a spectrum map transformed from the time-domain surface EMG by continuous wavelet transform (CWT). To evaluate the proposed SC-FGR model, we compared it with seven state-of-the-art models. The experimental results demonstrate that SC-FGR achieves 97.5% recognition accuracy on eight kinds of finger gestures with five subjects, which is much higher than that of comparable models.

Keywords: surface EMG, EMG sensor, finger gesture recognition, convolution neural network, artificial limb

INTRODUCTION

Comparing to traditional peripheral devices such as a mouse or a keyboard, finger gesture recognition (FGR) is much more convenient and natural for users to control an artificial limb and to interact with a computer (Rechy-Ramirez and Hu, 2015). As a result, FGR becomes more and more important during the past few years (Rechy-Ramirez and Hu, 2015). Currently, the most common methods of FGR are visual-based, voice-based, and surface electromyography (EMG)-based ones.

Among them, surface EMG is the comprehensive photoelectrical signal of potential muscle action on the surface of the skin (Botros et al., 2020). It is a kind of non-stationary signal, and its strength is sensitively proportional to the degree of muscle activity, which makes it can accurately represent the gesture of fingers (Botros et al., 2020). Therefore, surface EMG-based is widely adopted to achieve FGR.

Surface EMG-based FGR has been researched for many years. Among existing approaches, machine learning-based approach is very popular and successful (Qi et al., 2020; Wong et al., 2021). For example, Phinyomark et al. (2011) applied the critical index analysis and fractal dimension to extract the characteristics of surface EMG signals, and seven kinds of gestures were recognized from eight-channel EMG signals. Ishii et al. (2012) divided hand motions into six movements and classified finger motions using two types of characteristics. Khushaba et al. (2016) proposed the mutual component analysis (MCA) by improving the principal component analysis (PCA) to deduct the noise and redundant features. The recognition accuracy reached 95% for 15 kinds of gestures by combining the feature selection and MCA from eight channels of the surface EMG signals. Nge0 et al. (2014) used the multi-output convolution Gaussian process to analyze the dependence of multi-joint gesture and to estimate the finger joint motion. Through the correlation between knuckles, the regression model was modified to improve the recognition rate of finger posture. AlOmari and Liu (2015) constructed a model by combining genetic algorithm, particle swarm optimization, and support vector machine (SVM). Arozi et al. (2020) identified the hand gesture through the single channel of the surface EMG signal with the time-domain feature extraction, PCA, feature dimensionality reduction, and neural network. The recognition accuracy is 86.7% for nine kinds of gestures.

Recently, since convolution neural network (CNN) was proposed by Krizhevsky et al. in 2012 (Atzori et al., 2016), it has achieved great success in many fields of image recognition, natural language processing, and language translation (Wu et al., 2019b; Yao et al., 2019). As it has much better performance of feature extraction and non-linear fitting than traditional machine learning models, many researchers employed CNN to classify hand gestures from surface EMG signals. For example, Atzori et al. (2016) and Geng et al. (2016) selected CNN to classify hand gestures using the original surface EMG signals as the input signal. A spectral map that was obtained by the short-time Fourier transform (STFT) from the original surface EMG signal was put into the convolution network (Du et al., 2017; Côté-Allard et al., 2019a). Zia Ur Rehman et al. (2018) constructed a simple network model consisting of one convolutional layer, one pooling layer, and two fully connected layers. Then, the original surface EMG was directly used as the input of the CNN. Wu et al. (2018) proposed a model based on long short-term memory (LSTM) and CNN, where LSTM reserves time information and CNN extract features. Its performance was better than the model proposed in the study by Santello et al. (2016). Chen L. et al. (2020) designed a compact CNN with a small number of parameters to improve the classification accuracy of EMG signals. However, all these approaches mainly focus on developing a CNN-based recognition model while ignoring to acquire the

high-precision surface EMG. Hence, they still cannot fully satisfy the required recognition accuracy for real applications of artificial limb control and human-computer interaction.

To address this issue, this study proposes a novel FGR model that consists of two parts, namely, sensing and classification of surface EMG signal (SC-FGR). First, wireless sensors with high-precision surface EMG are developed for acquiring multichannel surface EMG signals from the forearm. Second, a CNN-based classification algorithm is proposed to classify the acquired surface EMG signals for FGR, where we named it CNN-FGR. A general chart of FGR with the proposed SC-FGR model is shown in **Figure 1**. The surface EMG signals of each channel are segmented by a moving window. A spectrum map is generated by continuous wavelet transform (CWT) from the segmented signals of each channel. Then, the spectrum maps of multiple channels are put into the CNN-FGR for classifying.

The main research contents and contributions of this study are as follows:

- (1) The wireless sensors are specially developed to acquire surface EMG from the forearm with high precision. Its resolution is 16 Bits, the sampling rate is 2 kHz, the common-mode rejection ratio (CMRR) is less than 70 dB, and the short-circuit noise (SCN) is less than 1.5 μ V.
- (2) A new CNN-FGR algorithm is proposed to accurately classify the surface EMG signals acquired by the developed wireless sensors. It consists of a 5-layer CNN that is trained on a spectrum map transformed from the time-domain signals of surface EMG by CWT.
- (3) A novel SC-FGR model is proposed for highly accurate FGR. It comprises two parts of the developed wireless sensors and the proposed CNN-FGR algorithm.
- (4) A surface EMG dataset is collected and shared online. It contains eight kinds of finger gestures with five subjects collected by the developed wireless sensors.

In the experiments, we evaluated the proposed SC-FGR model on the collected surface EMG dataset. The results demonstrate that the proposed SC-FGR model achieves 97.5% recognition accuracy, which is much higher than that of comparable models.

The rest of this article is organized as follows: A wireless surface EMG acquisition system is designed in section “A Wireless Surface EMG Acquisition System”; The data processing and CNN-FGR algorithm are described in detail in section “Data Processing and Network Architecture”; The proposed SC-FGR model is compared with several related models in Section “Experiment and Results”; and finally, section “Conclusion” concludes this study.

A WIRELESS SURFACE EMG ACQUISITION SYSTEM

The EMG is a weak electrophysiological signal of a muscle fiber group. It can be detected by sensors placed on the surface of skin or needle sensors implanted in muscle tissue (De Luca et al., 2006). The EMG signal is closely related to neuron muscular

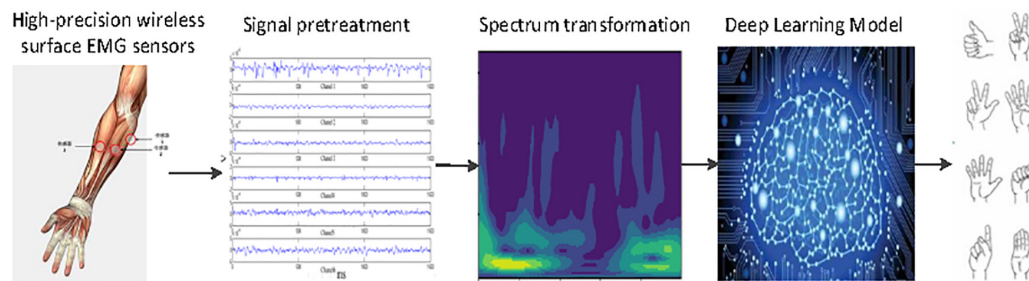


FIGURE 1 | General chart of finger gesture recognition (FGR) using sensing and classification of surface electromyography (EMG) signals (SC-FGR).

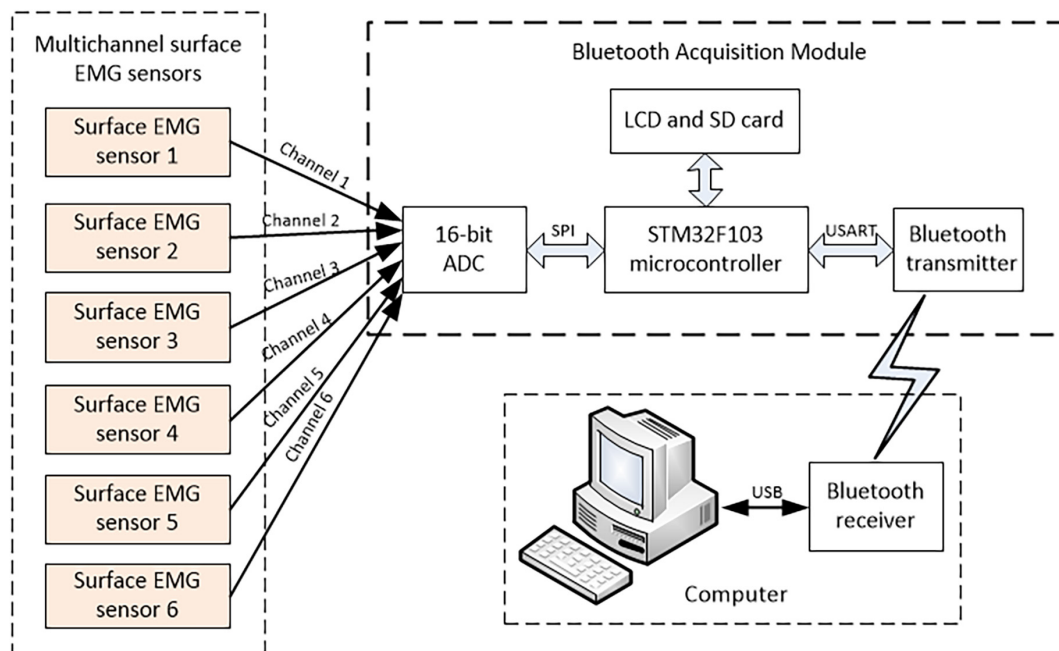


FIGURE 2 | Multichannel surface EMG acquisition.

activity information so that the surface EMG signals of the forearm can be used to analyze and recognize the finger gestures.

De Luca (1997) showed that the amplitude of the EMG signal was random and could be expressed by the arithmetic mean value of zero Gaussian distribution function. The surface EMG signal is a weak signal whose amplitude ranges from 0 to 10 mV (Peak-to-Peak) or 0 to 1.5 mV [root mean square (RMS)]. The frequency range of the available energy signal is limited from 0 to 1,000 Hz, and the dominant energy is distributed in the range from 50 to 150 Hz. In the same state of muscle motion, the amplitude-frequency characteristic curve of the EMG signal is similar, and the EMG signal has a certain regularity in the muscle motion state of different detection points. According to the characteristics of surface EMG, the frame of the acquisition module is designed as shown in **Figure 2**.

Inspired by the surface EMG sensor on the market, the surface EMG sensor consists of the surface EMG electrode and the signal conditioning circuit. This surface EMG sensor uses three

parallel silver electrodes with a spacing of 10 mm, including two measuring electrodes and one reference electrode, which prevent saturation caused by the common-mode signals. The silver electrode is put close to the skin for complete polarization, forming a capacitor by surface skin and electrode. To improve the accuracy, the front analog amplifier circuit is designed as close as possible to the silver electrode. This measure is beneficial to weaken the disturbance of white noise for the acquisition of surface EMG signals. Then, the potential difference between the two measuring electrodes is detected by the differential amplifier circuit and converted into a digital signal for signal preprocessing. Finally, the digital signal is transformed into a computer by the Bluetooth data acquisition module.

The signal conditioning circuit plays a key role in amplifying the weak signal to improve the performance of the whole acquisition system. The expected conditioning circuit is with high input impedance, high gain, wide frequency band, low noise, and high CMRR. It should amplify surface EMG signals while

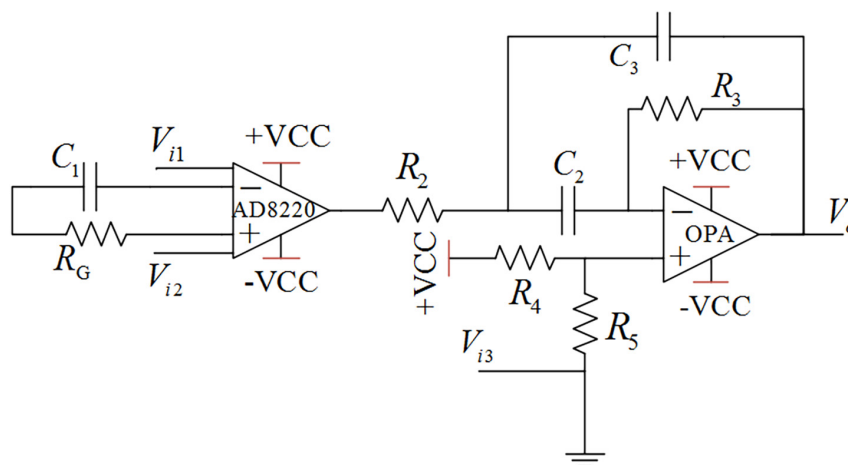


FIGURE 3 | Conditioning circuit for the analog signal.

suppressing other noise signals (Khokhar et al., 2010). The signal conditioning circuit uses instrument amplifier AD8220 with the JFET as the input of the preamplifier. The rail-to-rail amplifier

OPA364 constitutes the band-pass amplifier. The instrument amplifier AD8220 plays the role of first-order high-pass filtering, while the amplifier OPA364 plays the role of second-order band-pass filtering. All in all, the function of the analog conditioning circuit is to amplify the original EMG signal 1,000 times and then signal processing by the second-order band-pass filtering with the range of 5–1,000 Hz. The schematic diagram of the signal conditioning circuit (Fu et al., 2013) is shown in **Figure 3**. The theoretical gain of the signal conditioning circuit is shown as follows:

$$G = \frac{V_o}{V_{i2} - V_{i1}} = \left(\frac{49.4e^3}{R_G + R_{c1}} + 1 \right) \left(\frac{R_3 R_{c3}}{R_{c2} R_{c3} + R_2 R_{c2} + R_2 R_{c3} + R_2 R_3} \right) \quad (1)$$

where G represents amplifier gain; R_{c1} , R_{c2} , and R_{c3} represent the impedance of the capacitance C_1 , C_2 , and C_3 , respectively; V_{i1} , V_{i2} , and V_{i3} represent the input of the detection points; and V_o is the output of the signal conditioning circuit. The core design principles of the surface EMG acquisition system are anti-noise treatment, such as co-ground and anti-electromagnetic interference. This EMG acquisition system uses a Bluetooth module for physical isolation and anti-interference, avoiding 50-Hz interference from a wired connection with the computer. This data acquisition system contains a 16-bit AD conversion, an ARM processor, and a Bluetooth communication module, as shown in **Figure 2**. The output of the surface EMG sensor is connected to the input port of the AD converter by shielding line. It adopts the common ground technology between the analog signal and the digital signal. There is photoelectric isolation between the AD converter and the ARM microprocessor to reduce the crosstalk from digital signals to analog signals. On the one hand, the ARM controller stores the eigenvalues of the collected signal and stresses it in the local SD card. On the other hand, it transfers the collected signal to the HC-05 Bluetooth module through the USRT serial communication protocol.



FIGURE 4 | Multichannel wireless surface EMG acquisition device.

TABLE 1 | Characterization of different surface electromyography (EMG) acquisition systems.

	Delsys Trigno Wireless EMG (ADInstruments, 2020)	Biometrics DataLITE sEMG (Côté-Allard et al., 2019b)	Thalmic Labs MYO Armhand (Côté-Allard et al., 2019b)	This design
Number of channels	16	16	8	4–8
sEMG ADC	16 bits	13 bits	8 bits	16 bits
Sampling rate	2,000 Hz	2,000 Hz	1,000 Hz	2,000 Hz
Bandwidth	10–850 Hz	10–490 Hz	5–100 Hz	5–1,000 Hz
Contact material	Sliver	Stainless Steel	Stainless Steel	Sliver
Common-mode rejection ratio	>80 dB	N.A	N.A	>70 dB
Short-circuit noise	<0.75 μ V	<5 μ V	N.A	< 1.5 μV
Transfer protocol	BLE 4.2	WiFi	BLE 4.0	BLE 4.2

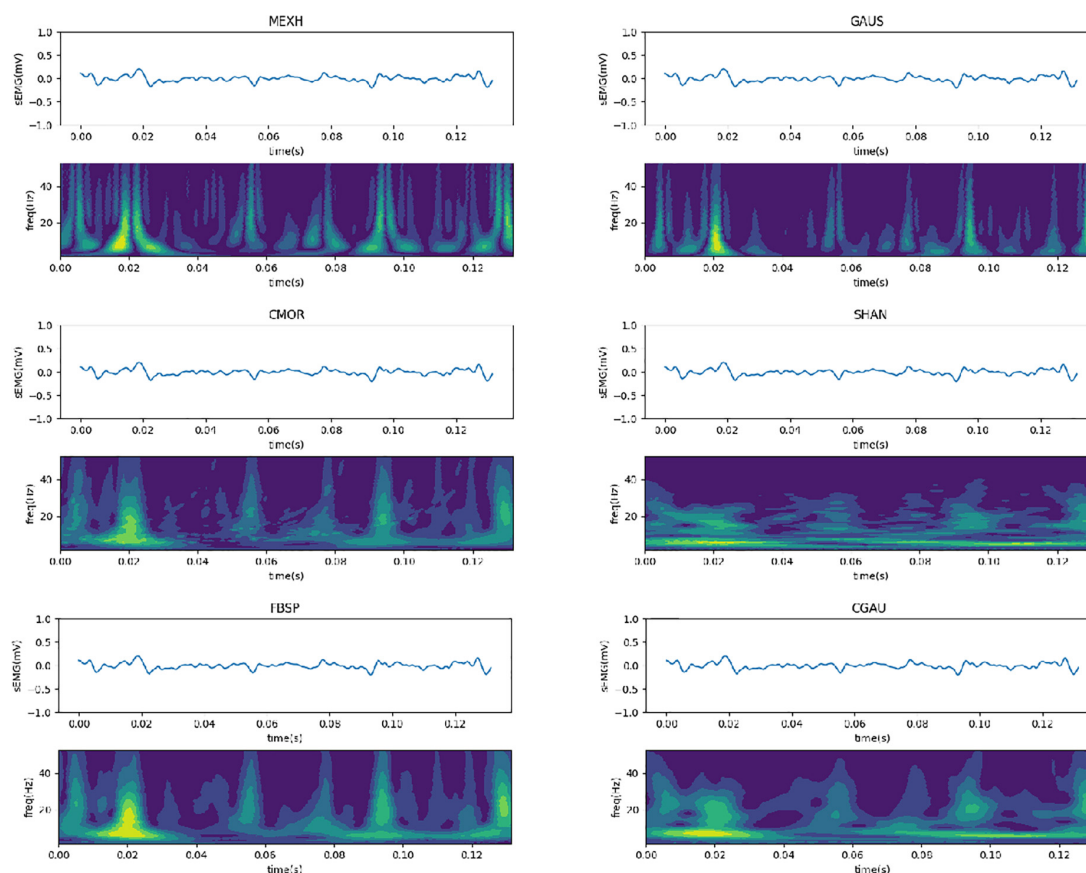


FIGURE 5 | Spectrum maps transformed from surface EMG with different kinds of parent wavelet functions.

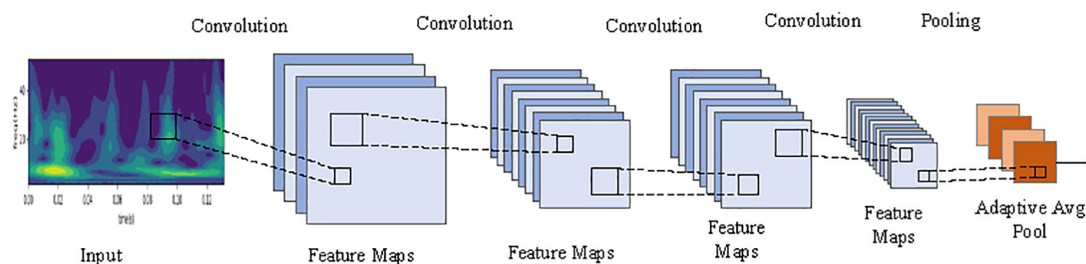


FIGURE 6 | The block diagram of the CNN-FGR algorithm.

Bluetooth communication realizes the information interaction function between sensors and the computer. The Bluetooth communication module uses low-energy radio communication technology to realize data transmission, with the maximum rate of 1 Mb/s (Song et al., 2020) and the effective communication of 15 m. The multichannel wireless surface EMG module is designed with a highly extending function and could be extended to 4–8 channels. The surface EMG device is shown in **Figure 4**.

The parameter comparison between the high-precision wireless surface EMG acquisition system and the other surface EMG acquisition systems on the market is shown in **Table 1**.

DATA PROCESSING AND NETWORK ARCHITECTURE

Signal Feature Extraction of Surface EMG

Since the surface EMG signal is non-stationary, it is limited to analysis the signal with Fourier transform. The STFT, which divides the signal into smaller segments by sliding windows and calculates the Fourier transform of each segment separately, is an effective method to solve that problem. A frequency spectrogram can be obtained from the transformation of STFT. When the

signal $x(t)$ and window function $w(t)$ are designed, the spectra can be calculated as follows:

$$\text{spectrogram}(x(t), w(t)) = |STFT_x(t, f)|^2 \quad (2)$$

$$STFT_x(t, f) = \int_{-\infty}^{+\infty} [x(u)w(u-t)] e^{-j2\pi fu} du \quad (3)$$

where f represents the frequency. The wavelet transform (WT) is similar to STFT, while it overcomes the disadvantage that the window does not change with frequency in STFT. By adjusting the width of the window, the WT adapts to the frequency changes in the signal. When the frequency of the processed signal increases, the WT improves the resolution by narrowing the time window. Furthermore, WT is an ideal analysis tool, which can obtain the amplitude and frequency of mutations in the signal.

$$X(a, b) = \frac{1}{\sqrt{b}} \int_{-\infty}^{+\infty} x(t) \phi\left(\frac{t-a}{b}\right) dt \quad (4)$$

$$\int_{-\infty}^{+\infty} \frac{|\phi(\omega)|^2}{\omega} d\omega < \infty \quad (5)$$

where the Fourier transform $\phi(\omega)$ must satisfy Equation 5. $\phi(t)$ is named as the parent wavelet function, which is a signal with limited duration, frequency change, and zero mean value. The scaling factor b and the translation factor a control the scaling and transform of the wavelet function, respectively. There are many kinds of parent wavelet functions for the transform, such as Mexican hat wavelet (MEXH), Gaussian wavelet (GAUS), complex Morlet wavelet (CMOR), Shannon wavelet (SHAN), frequency B-spline wavelet (FBSP), and complex Gaussian wavelet (CGAU). MEXH function is defined by Equation 6 as follows:

$$\psi(t) = c(1-t^2)e^{-t^2/2} \quad (6)$$

where $c = \frac{2}{\sqrt{3}}\pi^{1/4}$. GAUS is the differential form derived from the Gaussian function. It is defined by Equation 7 as follows:

$$\psi(t) = C_{p1}te^{-t^2} \quad (7)$$

where $C_{p1} = \sqrt{2/\pi}$. CMOR is defined by Equation 8 in the time-domain and by Equation 9 in the frequency domain as follows:

$$\psi(t) = \frac{1}{\sqrt{\pi f_b}} \bullet e^{j2\pi f_c t - (t^2/f_b)} \quad (8)$$

$$\Psi(f) = e^{\pi^2 f_b (f-f_c)^2} \quad (9)$$

where f_c is the center frequency and f_b is the bandwidth. SHAN is defined by Equation 10 as follows:

$$\psi(t) = \sqrt{f_b} \sin c(f_b x) e^{2j\pi f_c x} \quad (10)$$

where f_c is the center frequency and f_b is the bandwidth. FBSP is defined by Equation 11 as follows:

$$\psi(t) = \sqrt{f_b} \left[\sin\left(\frac{f_b t}{m}\right) \right]^m e^{2j\pi f_c t} \quad (11)$$

where m is an integer parameter, f_c is the center frequency, and f_b is the bandwidth. CGAU is defined by Equation 12 as follows:

$$\psi(t) = C_p e^{-it} e^{-x^2} \quad (12)$$

where C_p is constant.

After the CWT of the surface EMG signals, the corresponding spectrum map is similar to the image on the scale and also contains the frequency domain information of the timing sequence data. The six-channel surface EMG signals of the forearm were collected by the high-precision wireless surface EMG sensors, and the data of each channel were separated by applying a sliding window of 264 samples (132 ms). The parent wavelet of the CWT adopts the optimal wavelet function, calculating the CWTs with 64 scales to obtain the 64×264 matrix of spectral information. The matrix is set as input to the CNN-FGR algorithm. Thus, the input of the CNN-FGR algorithm has six channels, each consisting of a matrix with the size of 64×264 . **Figure 5** is the spectrum maps of the spectral information transformed from 264 EMG data with different kinds of parent wavelet functions, such as MEXH, GAUS, CMOR, SHAN, FBSP, and CGAU.

CNN-FGR Algorithm

Chen L. et al. (2020) used a compact CNN to improve the hand gesture recognition by surface EMG. Inspired from that model, the CNN-FGR algorithm consists of four convolutional layers and one mean pool layer as shown in **Figure 6**, and its design details are listed in **Table 2**.

The loss function is calculated as follows:

$$\text{Loss} = - \sum_{i=1}^n y_i \log(y'_i) \quad (13)$$

where y_i is the true value of the first class, n is the number of categories, y'_i is the first-class prediction value of the output. Since one-hot coding was adopted, the true value of one class is 1, while the true value of the other classes is 0.

The three quantities where accuracy rate (AR) is used to evaluate the performance of the SC-FGR model, such as AR, the mean AR (MAR), and the SD of AR (SD-AR), are, respectively,

TABLE 2 | Configuration of CNN of CNN-FGR algorithm.

Layers of Network	Parameters of each layer
Convolutional layer 1 (Activation Function: ReLU)	kernel_size = 3, stride = 1 Number of feature graphs:16
Convolutional layer 2 (Activation Function: ReLU)	kernel_size = 3, stride = 2 Number of feature graphs:32
Dropout	$P = 0.5$
Convolutional layer 3 (Activation Function: ReLU)	kernel_size = 3, stride = 1 Number of feature graphs:32
Convolutional layer 4 (Activation Function: ReLU)	kernel_size = 3, stride = 2 Number of feature graphs: 64
Convolutional layer 5 (Activation Function: ReLU)	adaptive_avg_pool2d

computed as Equations 14–16. A test set composed of t number of instances x_i with ω known is used for the test stage.

$$AR = \frac{1}{t} \sum_{i=1}^t \Psi(w, f(x_i)), \Psi(w, f(x_i)) = \begin{cases} 1, & \text{if } w = f(x_i) \\ 0, & \text{else} \end{cases} \quad (14)$$

$$MAR = \frac{1}{n} \sum_{k=1}^n AR_k \quad (15)$$

$$SD - AR = \sqrt{\frac{1}{n} \sum_{k=1}^n (AR_k - MAR)^2} \quad (16)$$

where $f(x_i)$ represents the calculated label of x_i , and n is the repeated times of computing AR. MAR represents the classification ability of the algorithm, and SD-AR represents the robustness of the algorithm.

Advanced optimization methods were used for the backpropagation of the CNN-FGR algorithm with the ultimate goal to minimize the function loss. In the field of image recognition, the common size of the convolutional kernel is selected as 3×3 , 5×5 , or 7×7 (Krizhevsky et al., 2012; Simonyan and Zisserman, 2014). Therefore, the different sizes of the convolutional kernel in the CNN-FGR algorithm model are evaluated to get a better experimental result. Meanwhile, the various layer feature maps of the model are also set smaller to minimize the parameters of the model. The step length of the convolution is set to 2, for reducing the feature parameters

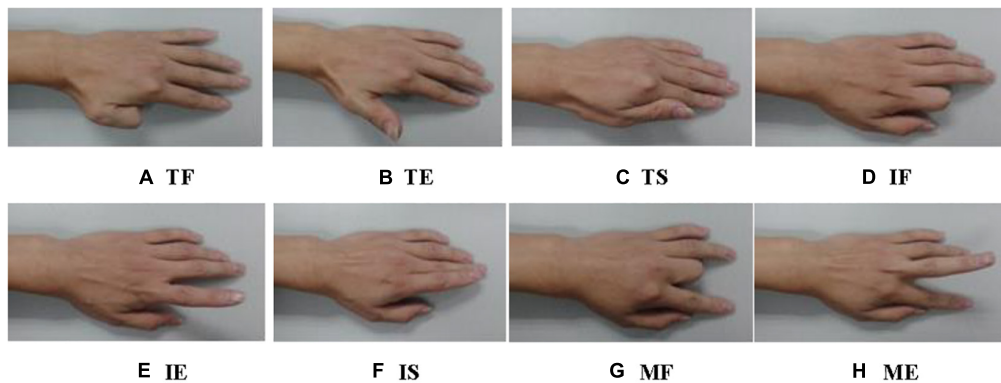


FIGURE 7 | Eight kinds of finger gestures: (A) Thumb Flexion (TF), (B) Thumb Extension (TE), (C) Thumb Swing (TS), (D) Index-finger Flexion (IF), (E) Index-finger Extension (IE), (F) Index-finger Swing (IS), (G) Middle-finger Flexion (MF), and (H) Middle-finger Extension (ME).

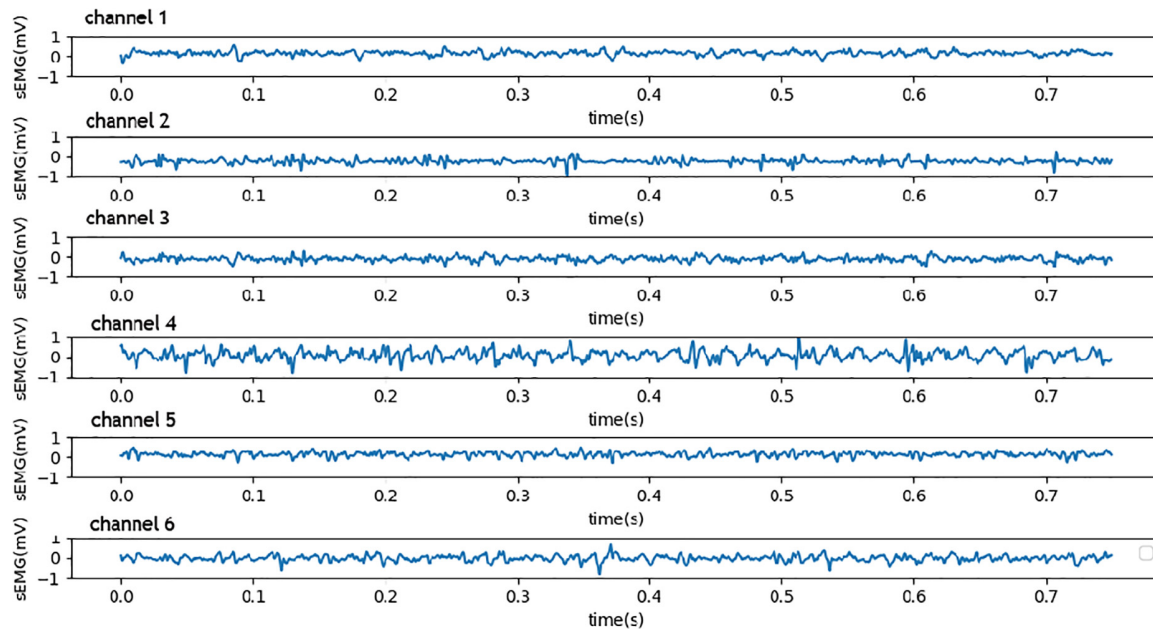


FIGURE 8 | Six channels of the raw EMG signals from the high-precision sensors.

by half. To further reduce the number of network parameters, the output of the model used the convolutional layer with adaptive mean sampling for classification, instead of the full connection layer.

EXPERIMENT AND RESULTS

Finger Gestures

Before the experiment, the collection points of the surface EMG from the forearm must be disinfected and cleaned to reduce skin contact interference. In the experiment, the subject sat on a chair with his left arm lying flat on the table and relaxed. In each group of experiments, as shown in **Figure 7**, each subject completed eight types of gestures, namely, Thumb Flexion (TF), Thumb Extension (TE), Thumb Swing (TS), Index-finger Flexion (IF), Index-finger Extension (IE), Index-finger Swing (IS), Middle-finger Flexion (MF), and Middle-finger Extension (ME).

Number of Sensors and Layout of Detection Points

The surface EMG signal is closely related not only to the objective factors such as human physical state and movement state but also to the form and location of the detection electrode. The number of electrodes also has a great impact on the accuracy of surface

EMG signal recognition. Extensive research and experiments showed that the acquisition of surface EMG signals with six channels can not only effectively identify single and multi-finger movement information but also avoid the waste of resources with over-channel detection. It was found that the electrodes were placed on the nerve-dominated region, and the EMG signals collected in the 10-tendon head or muscle edge area were usually weak. When sensors were placed vertically on the muscle fibers, the surface EMG signals were strongest. Since the front group muscles of the forearm cover the flexor, it mainly controls the bending movement of the elbow, wrist, and knuckles. The muscles of the back group cover the stretched muscles, which mainly control the stretching movement of each joint. In this experiment, six surface electrodes were placed on the corresponding muscle abs, and the electrodes were radially perpendicular to the muscle fibers. The sensors were fixed on the forearm with a bandage in moderate tension. Three sensors were placed on the corresponding muscle abs at the front of the forearm, mainly for detecting the bending movement of the finger, while the other sensors were placed at the back of the forearm for detecting the stretching movement of the fingers. The raw EMG signals detected by six sensors on the forearm are shown in **Figure 8**.

Classification Results

This experiment used the high-precision wireless surface EMG sensors and DELSYS data acquisition system to collect six channels of the surface EMG signal, with a frequency of 2 kHz. Before classification, the collected surface EMG signal must be pretreated and feature extracted. The original EMG signal is preprocessed with a 264-sample-point (132 ms) sliding window and a 100-sample-point incremental step. After the data segment processing, each experiment of each gesture obtains 12 samples, and 300 samples are collated after 25 repeating times. The total datasets of eight gestures of five subjects (i.e., S1, S2, S3, S4, and S5) are 12,000 samples. Each subject has 2,400 samples, where 1,920 samples are adopted as training set and 480 samples are adopted as testing set.

To evaluate the effects of CWT in transforming the surface EMG from time-domain to spectrum map, we, respectively, trained the CNN-FGR algorithm on the time-domain and the spectrum map of surface EMG. The comparison results on the testing set are shown in **Table 3**, where we observed that the CNN-FGR algorithm trained on the spectrum map of surface EMG achieves much higher accuracy than that trained on the time-domain of surface EMG. This observation demonstrates that transforming the surface EMG from time-domain to

TABLE 3 | The effects of continuous wavelet transform (CWT) on the accuracy of gesture classification of five subjects (S1, S2, S3, S4, and S5).

	S1	S2	S3	S4	S5	MAR	SD-AR
Time-domain (%)	92.3	94.37	97.5	95.4	98.54	95.62	2.22
Spectrum map (%)	92.50	97.50	97.50	100.00	100.00	97.50	2.74

TABLE 4 | The accuracy of the CNN-FGR algorithm with the convolutional kernel size of 3×3 on five subjects.

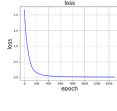
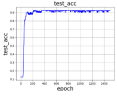
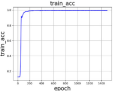
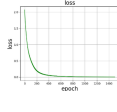
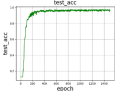
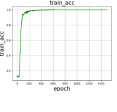
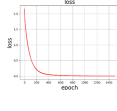
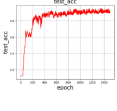
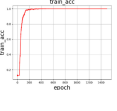
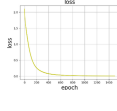
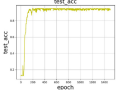
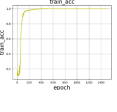
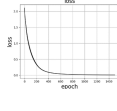
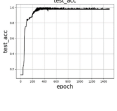
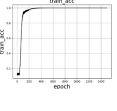
	Loss	Test_acc	Train_acc
S1			
S2			
S3			
S4			
S5			

TABLE 5 | The comparison of accuracy with different convolution kernel sizes.

	S1	S2	S3	S4	S5	MAR	SD-AR
3×3 (%)	92.5	97.5	97.5	96.67	100	96.83	2.44
5×5 (%)	92.5	97.5	96.875	98.58	100	97.09	2.53
7×7 (%)	92.5	100	97.5	97.29	100	97.46	2.74
9×9 (%)	92.5	100	97.71	98.125	100	97.67	2.75

spectrum map by CWT is beneficial for the CNN-FGR algorithm to achieve a better performance of FGR.

There are two factors affecting the identification accuracy in the SC-FGR algorithm model. One is the size of the convolutional kernel, and the other is the parent wavelet function. Using the same parent wavelet function “CGAU” for CWT transform, the different sizes of the convolutional kernel are compared to get a better recognition accuracy. The training accuracy curve, loss curve during training, and testing accuracy curve are used to analyze the results of FGR. The accuracy of the CNN-FGR algorithm with the convolutional kernel size of 3×3 is shown in Table 4.

From Table 4, we found that the training accuracy keeps increasing and loss keeps decreasing with more epochs until reaching convergence. Similarly, testing accuracy also keeps increasing with more epochs until reaching convergence. These findings verify that the CNN-FGR algorithm can be well applied

to classify these samples for FGR. In the experiment, we compared the accuracy of the CNN-FGR algorithm with the kernel size of 3×3 , 5×5 , 7×7 , and 9×9 on collected datasets.

From Table 5, it can be observed that the classification ability of the algorithm is improved, but the robustness of the algorithm becomes worse, while the size of the convolution kernel increases. The size of 5×5 is a better selection as the convolution kernel, because not only the accuracy is high, but also the robustness performed well.

To choose the suitable parent wavelet function for CNN-FGR, the experiments are carried out on different parent wavelet functions, such as MEXH, SHAN, GAUS, FBSP, CGAU, and CMOR. For dataset S3, the comparison results of accuracy of various parent wavelet functions with the same convolutional kernel size of 5×5 are shown in Table 6.

On all collected datasets, the comparison results of accuracy of various parent wavelet functions with the same convolutional kernel size of 5×5 are shown in Table 7.

From Table 7, it is easy to get the results that the accuracy of GAUS is higher than that of other wavelet functions, but the robustness is worse. Considering the classification ability and the robustness, the algorithm with the parent wavelet MEXH performs better.

Finally, to evaluate the proposed SC-FGR model, we compared it with several related models. Especially, enhanced time-domain (EnhancedTD) (Khushaba et al., 2016; Fournelle and Bost, 2019), time-domain cycle (TDC) (Tang et al., 2010), autoregression (AR) (Soares et al., 2003), sample entropy (SampEn) (Delgado-Bonal and Marshak, 2019), and wavelet package coefficient (WPC) (Zhao et al., 2006) are selected as feature extractors. The classical classifiers [e.g., probabilistic neural network (PNN) (Zeinali and Story, 2017), linear discriminant analysis (LDA) (Zhang et al., 2012), and SVM (Varatharajan et al., 2018)], CNN (Chen H.F. et al., 2020), and CWT-EMGNet

TABLE 6 | The comparison results of accuracy of various parent wavelet functions on the dataset S3.

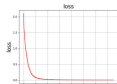

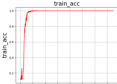
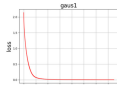
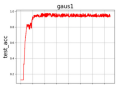
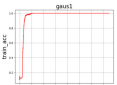
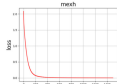
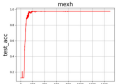
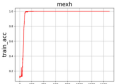
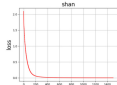
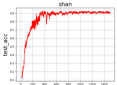
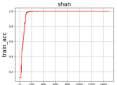
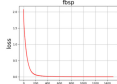
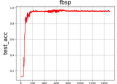
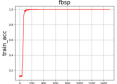
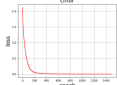
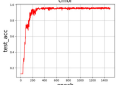
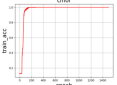
	Loss	Test_acc	Train_acc
CGAU			
GAUS			
MEXH			
SHAN			
FBSP			
CMOR			

TABLE 7 | The comparison results of accuracy of various parent wavelet functions on the collected datasets.

	S1	S2	S3	S4	S5	MAR	SD-AR
MEXH (%)	92.50	98.75	97.50	97.71	98.75	97.04	2.33
SHAN (%)	92.50	98.54	92.50	98.38	96.67	95.72	2.71
FBSP (%)	92.50	98.13	97.91	96.67	100.00	97.04	2.51
CMOR (%)	92.50	98.33	97.50	97.5	100.00	97.17	2.51
CGAU (%)	92.50	97.50	96.88	98.54	100.00	97.08	2.52
GAUS (%)	92.50	97.50	97.50	100.00	100.00	97.50	2.74

TABLE 8 | The comparison results of accuracy of various models on the collected datasets.

	S1	S2	S3	S4	S5	MAR	SD-AR
TDC-AR+PCA+PNN (%) (Fu et al., 2017)	90.84	89.39	93.88	95.8	95.7	93.12	2.59
TDC-WPC+PCA+PNN (%)	90.67	91.45	95.78	98.17	97.54	94.72	3.10
EnhancedTD+LDA (%) (Zhang et al., 2012)	89.58	92.41	94.13	95.81	97.48	93.88	2.73
EnhancedTD+SVM (%)	88.29	90.57	94.06	93.59	95.03	92.31	2.50
SampEn+LDA (%)	89.67	92.75	96.22	95.77	94.07	93.70	2.36
SampEn+SVM (%)	87.44	90.77	93.18	94.44	93.82	91.93	2.57
CNN (%) (Chen H.F. et al., 2020)	92.3	94.37	97.5	95.4	98.54	95.62	2.22
CWT+EMGNet (%) (Chen L. et al., 2020)	92.5	94.58	96.875	99.17	99.58	96.54	2.70
SC-FGR (%)	92.50	97.50	97.50	100.00	100.00	97.50	2.74

(Chen L. et al., 2020) are adopted as classifiers. The comparison results are recorded in **Table 8**, where we clearly observed that the SC-FGR model achieves 97.5% accuracy, which is the best among all the models. Hence, we concluded that the proposed SC-FGR model is powerful for FGR.

CONCLUSION

This study proposes a novel SC-FGR model that consists of two parts, namely, sensing and classification of the surface EMG signal. First, wireless sensors are developed for acquiring multichannel surface EMG signals from the forearm according to the characteristics of the surface EMG signal. These sensors can provide a high-precision signal source of surface EMG for FGR. In addition, a CNN-based classification algorithm, i.e., CNN-FGR, is proposed for FGR based on the acquired surface EMG by the developed wireless sensors. The CNN-FGR is trained on a spectrum map transformed from the time-domain of surface EMG by CWT. The experimental results demonstrate that the proposed SC-FGR model achieves 97.5% recognition accuracy on eight kinds of finger gestures with five subjects, which is much higher than that of comparable models. In the future, we plan to adopt the techniques of latent factor analysis (Wu et al., 2019a, 2020, 2021a,b), cognitive computing (Wu et al., 2021c), and attention mechanism

(Zheng and Chen, 2021) to simultaneously recognize the gesture and strength of the fingers based on the surface EMG of the forearm.

DATA AVAILABILITY STATEMENT

The datasets presented in this study can be found in online repositories. The name of the repository and accession number can be found below: Baidu Netdisk, https://pan.baidu.com/s/1wXT_i2kPMRALvfl17bP1YA (access code: f6wu).

AUTHOR CONTRIBUTIONS

JF contributed to writing—original draft, conceptualization, and methodology. SC contributed to the experiment design. LC contributed to the data curation. LY contributed to writing—review and editing. All authors contributed to the article and approved the submitted version.

FUNDING

This work was supported by the Surface Project of Chongqing Natural Science Fund, Grant No. cstc2021jcyj-msxmX0144.

REFERENCES

- ADInstruments (2020). *Delsys for research available from ADInstruments*. Available Online at: <https://www.adinstruments.com/partners/delsys> (accessed September 04, 2021).
- AlOmari, F., and Liu, G. (2015). Novel hybrid soft computing pattern recognition system SVM–GAPSO for classification of eight different hand motions. *Optik Int. J. Light Electron Opt.* 126, 4757–4762. doi: 10.1016/j.ijleo.2015.08.170
- Arozi, M., Caesarendra, W., Ariyanto, M., Munadi, M., Setiawan, J. D., and Glowacz, A. (2020). Pattern Recognition of Single-Channel sEMG Signal Using PCA and ANN Method to Classify Nine Hand Movements. *Symmetry* 12, 541–549. doi: 10.3390/sym12040541
- Atzori, M., Cognolato, M., and Müller, H. (2016). Deep learning with convolutional neural networks applied to electromyography data: a resource for the classification of movements for prosthetic hands. *Front. Neurobot.* 10:9. doi: 10.3389/fnbot.2016.00009
- Botros, F., Phinyomark, A., and Scheme, E. (2020). EMG-Based Gesture Recognition: is It Time to Change Focus from the Forearm to the Wrist? *IEEE Trans. Industr. Inform.* 99:1. doi: 10.1109/TMC.2020.3045635
- Chen, H. F., Zhang, Y., Li, G. F., Fang, Y., and Liu, H. (2020). Surface electromyography feature extraction via convolutional neural network. *Int. J. Mach. Learn. Cybern.* 11, 185–196. doi: 10.1007/s13042-019-00966-x
- Chen, L., Fu, J., Wu, Y., Li, H., and Zheng, B. (2020). Hand Gesture Recognition Using Compact CNN Via Surface Electromyography Signals. *Sensors* 20, 672–680. doi: 10.3390/s20030672
- Côté-Allard, U., Fall, C. L., Drouin, A., Campeau-Lecours, A., Gosselin, C., Glette, K., et al. (2019a). Deep learning for electromyographic hand gesture signal classification using transfer learning. *IEEE Trans. Neural Syst. Rehabil. Eng.* 27, 760–771. doi: 10.1109/TNSRE.2019.2896269
- Côté-Allard, U., Gagnon-Turcotte, G., Laviolette, F., and Gosselin, B. (2019b). A low-cost, wireless, 3-d-printed custom armband for semg hand gesture recognition. *Sensors* 19, 2811–2820. doi: 10.3390/s19122811
- De Luca, C. J. (1997). The use of surface electromyography in biomechanics. *J. Appl. Biomech.* 13, 135–163. doi: 10.1123/jab.13.2.135
- De Luca, C. J., Adam, A., Wotiz, R., Gilmore, L. D., and Nawab, S. H. (2006). Decomposition of surface EMG signals. *J. Neurophysiol.* 9, 1646–1657. doi: 10.1152/jn.00009.2006
- Delgado-Bonal, A., and Marshak, A. (2019). Approximate entropy and sample entropy: a comprehensive tutorial. *Entropy* 21:541. doi: 10.3390/e21060541
- Du, Y., Jin, W., Wei, W., Hu, Y., and Geng, W. (2017). Surface EMG-based inter-session gesture recognition enhanced by deep domain adaptation. *Sensors* 17, 458–466. doi: 10.3390/s17030458
- Fournelle, M., and Bost, W. (2019). Wave front analysis for enhanced time-domain beamforming of point-like targets in optoacoustic imaging using a linear array. *Photoacoustics* 14, 67–76. doi: 10.1016/j.pacs.2019.04.002
- Fu, J., Jian, C., and Shi, Y. (2013). “Design of a low-cost wireless surface EMG acquisition system,” in *The 6th International IEEE EMBS Conference on Neural Engineering?* California, 699–702.
- Fu, J., Xiong, L., Song, X., Yan, Z., and Xie, Y. (2017). “Identification of finger movements from forearm surface EMG using an augmented probabilistic neural network,” in *2017 IEEE/SICE International Symposium on System Integration (SII)*, (Taipei, Taiwan: IEEE). doi: 10.1109/SII.2017.8279278
- Geng, W., Du, Y., Jin, W., Wei, W., Hu, Y., and Li, J. (2016). Gesture recognition by instantaneous surface EMG images. *Sci. Rep.* 6:36571. doi: 10.1038/srep36571
- Ishii, C., Saitou, S., Sasaki, A., and Hashimoto, H. (2012). “Distinction of finger operation for myoelectric prosthetic hand on the basis of surface EMG,” in *16th Csi International Symposium on Artificial Intelligence & Signal Processing*, (Shiraz, Iran: IEEE). doi: 10.1109/AISP.2012.6313786
- Khokhar, Z. O., Xiao, Z. G., and Menon, C. (2010). Surface EMG pattern recognition for real-time control of a wrist exoskeleton. *Biomed. Eng. Online* 9, 41–47. doi: 10.1186/1475-925X-9-41
- Khushaba, R. N., Al-Timemy, A., Kodagoda, S., and Nazarpour, K. (2016). Combined influence of forearm orientation and muscular contraction on EMG pattern recognition. *Expert Syst. Appl.* 61, 154–161. doi: 10.1016/j.eswa.2016.05.031
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Adv. Neural Inform. Process. Syst.* 03, 1097–1105.

- Ngeo, J., Tamei, T., and Shibata, T. (2014). Estimation of continuous multi-DOF finger joint kinematics from surface EMG using a multi-output Gaussian process. *Eng. Med. Biol. Soc.* 08, 3537–3540. doi: 10.1109/EMBC.2014.6944386
- Phinyomark, A., Phothisonothai, M., Phukpattaranont, P., and Limsakul, C. (2011). Critical Exponent Analysis Applied to Surface EMG Signals for Gesture Recognition. *Metrol. Meas. Syst.* 18, 645–658. doi: 10.2478/v10178-011-0061-9
- Qi, J., Jiang, G., Li, G., Sun, Y., and Tao, B. (2020). Surface EMG hand gesture recognition system based on PCA and GRNN. *Neural Comput. Appl.* 32, 6343–6351. doi: 10.1007/s00521-019-04142-8
- Rechy-Ramirez, E. J., and Hu, H. (2015). Bio-signal based control in assistive robots: a survey. *Digit. Commun. Netw.* 1, 85–101. doi: 10.1016/j.dcan.2015.02.004
- Santello, M., Bianchi, M., Gabiccini, M., Ricciardi, E., Salvietti, G., Prattichizzo, D., et al. (2016). Hand synergies: integration of robotics and neuroscience for understanding the control of biological and artificial hands. *Phys. Life Rev.* 06, 1–23. doi: 10.1016/j.plev.2016.02.001
- Simonyan, K., and Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv*. Available online at: <https://arxiv.org/abs/1409.1556> (accessed September 04, 2021).
- Soares, A., Andrade, A., Lamounier, E., and Carrijo, R. (2003). The development of a virtual myoelectric prosthesis controlled by an EMG pattern recognition system based on neural networks. *J. Intell. Inf. Syst.* 21, 127–141. doi: 10.1023/A:1024758415877
- Song, E., Kim, S., Han, J., and Kwon, K. (2020). A 2.4-GHz Quadrature Local Oscillator Buffer Insensitive to Frequency-Dependent Loads for Bluetooth Low Energy Applications. *IEEE Microw. Wirel. Compon. Lett.* 30, 961–964. doi: 10.1109/LMWC.2020.3016733
- Tang, M., Li, K., and Ma, X. T. (2010). Research on pulse wave signal and time-domain feature extraction algorithm. *Comput. Modernization* 176, 16–22.
- Varatharajan, R., Manogaran, G., and Priyan, M. K. (2018). A big data classification approach using LDA with an enhanced SVM method for ECG signals in cloud computing. *Multimed. Tools Appl.* 77, 10195–10215. doi: 10.1007/s11042-017-5318-1
- Wong, W. K., Juwono, F. H., and Khoo, B. T. T. (2021). Multi-features capacitive hand gesture recognition sensor: a machine learning approach. *IEEE Sens. J.* 21, 8441–8450. doi: 10.1109/JSEN.2021.3049273
- Wu, D., Luo, X., Shang, M. S., He, Y., Wang, G., Zhou, M., et al. (2019b). A Deep Latent Factor Model for High-Dimensional and Sparse Matrices in Recommender Systems. *IEEE Trans. Syst. Man Cybern. Syst.* 99, 1–12.
- Wu, D., He, Q., Luo, X., Shang, M., He, Y., Wang, G., et al. (2019a). A posteriorneighborhood-regularized latent factor model for highly accurate web service QoS prediction. *IEEE Trans. Serv. Comput.* 99:1. doi: 10.1109/TSC.2019.2961895
- Wu, D., Luo, X., Shang, M., He, Y., Wang, G., Wu, X., et al. (2020). A Data-Characteristic-Aware Latent Factor Model for Web Service QoS Prediction. *IEEE Trans. Knowl. Data Eng.* 32:1.
- Wu, D., He, Y., Luo, X., and Zhou, M. (2021a). A Latent Factor Analysis-Based Approach to Online Sparse Streaming Feature Selection. *IEEE Trans. Syst. Man Cybern. Syst.* 1–15. doi: 10.1109/TSMC.2021.3096065
- Wu, D., Shang, M., Luo, X., and Wang, Z. (2021b). An L1-and-L2-Norm-Oriented Latent Factor Model for Recommender Systems. *IEEE Trans. Neural Netw. Learn. Syst.* 1–14. doi: 10.1109/TNNLS.2021.3071392
- Wu, E. Q., Lin, C. T., Zhu, L. M., Tang, Z. R., Jie, Y. W., Zhou, G. R., et al. (2021c). Fatigue Detection of Pilots' Brain Through Brain Cognitive Map and Multi-Layer Latent Incremental Learning Model. *IEEE Trans. Cybern.* [Epub Online ahead of print].
- Wu, Y., Zheng, B., and Zhao, Y. (2018). "Dynamic gesture recognition based on LSTM-CNN," in *2018 Chinese Automation Congress (CAC)*, (Xi'an, China: IEEE). doi: 10.1109/TNNLS.2021.3071392
- Yao, G., Lei, T., and Zhong, J. (2019). A review of convolutional-neural-network-based action recognition. *Pattern Recognit. Lett.* 118, 14–22. doi: 10.1109/TCYB.2021.3068300
- Zeinali, Y., and Story, B. A. (2017). Competitive probabilistic neural network. *Integr. Comput. Aided Eng.* 24, 105–118. doi: 10.1109/CAC.2018.8623035
- Zhang, D., Xiong, A., Zhao, X., and Han, J. (2012). "PCA and LDA for EMG-based control of bionic mechanical hand," in *2012 IEEE International Conference on Information and Automation*, (Shenyang, China: IEEE). doi: 10.1016/j.patrec.2018.05.018
- Zhao, J., Xie, Z., Jiang, L., Cai, H., Liu, H., Hirzinger, G., et al. (2006). "EMG control for a five-fingered underactuated prosthetic hand based on wavelet transform and sample entropy," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, (Beijing, China: IEEE). doi: 10.3233/ICA-170540
- Zheng, X., and Chen, W. (2021). An Attention-based Bi-LSTM Method for Visual Object Classification via EEG. *Biomed. Signal Process. Control* 63:102174. doi: 10.1109/ICInfA.2012.6246955
- Zia Ur Rehman, M., Waris, A., Gilani, S. O., Jochumsen, M., Niazi, I. K., Jamil, M., et al. (2018). Multiday EMG-based classification of hand motions with deep learning techniques. *Sensors* 18, 2497–2505. doi: 10.1109/IROS.2006.282425

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

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

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



cuSCNN: A Secure and Batch-Processing Framework for Privacy-Preserving Convolutional Neural Network Prediction on GPU

Yanan Bai^{1,2}, Quanliang Liu^{2,3}, Wenyuan Wu^{1*} and Yong Feng¹

¹ Chongqing Key Laboratory of Automated Reasoning and Cognition, Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Chongqing, China, ² University of Chinese Academy of Sciences, Beijing, China, ³ Chongqing School, University of Chinese Academy of Sciences, Chongqing, China

OPEN ACCESS

Edited by:

Song Deng,
Nanjing University of Posts and
Telecommunications, China

Reviewed by:

Huyong Yan,
Chongqing Technology and Business
University, China

Jin Dong,
Oak Ridge National Laboratory (DOE),
United States

Tianchen Wang,
University of Notre Dame,
United States

*Correspondence:

Wenyuan Wu
wuwenyuan@cigit.ac.cn

Received: 22 October 2021

Accepted: 22 November 2021

Published: 23 December 2021

Citation:

Bai Y, Liu Q, Wu W and Feng Y (2021)
cuSCNN: A Secure and
Batch-Processing Framework for
Privacy-Preserving Convolutional
Neural Network Prediction on GPU.
Front. Comput. Neurosci. 15:799977.
doi: 10.3389/fncom.2021.799977

The emerging topic of privacy-preserving deep learning as a service has attracted increasing attention in recent years, which focuses on building an efficient and practical neural network prediction framework to secure client and model-holder data privately on the cloud. In such a task, the time cost of performing the secure linear layers is expensive, where matrix multiplication is the atomic operation. Most existing mix-based solutions heavily emphasized employing BGV-based homomorphic encryption schemes to secure the linear layer on the CPU platform. However, they suffer an efficiency and energy loss when dealing with a larger-scale dataset, due to the complicated encoded methods and intractable ciphertext operations. To address it, we propose cuSCNN, a secure and efficient framework to perform the privacy prediction task of a convolutional neural network (CNN), which can flexibly perform on the GPU platform. Its main idea is 2-fold: (1) To avoid the trivial and complicated homomorphic matrix computations brought by BGV-based solutions, it adopts GSW-based homomorphic matrix encryption to efficiently enable the linear layers of CNN, which is a naive method to secure matrix computation operations. (2) To improve the computation efficiency on GPU, a hybrid optimization approach based on CUDA (Compute Unified Device Architecture) has been proposed to improve the parallelism level and memory access speed when performing the matrix multiplication on GPU. Extensive experiments are conducted on industrial datasets and have shown the superior performance of the proposed cuSCNN framework in terms of runtime and power consumption compared to the other frameworks.

Keywords: privacy-preserving, convolutional neural network, homomorphic encryption, GPU computation, deep learning, cloud computing

1. INTRODUCTION

Deep learning (DL) has been applied to lots of fields [e.g., visual recognition (He et al., 2016), medical diagnosis (Shen et al., 2017), risk assessment (Deng et al., 2021a,b), and a recommender system (Shi et al., 2020; Wu et al., 2021a,b)], which achieves a superior performance in comparison with human cognition. The DL with a complex neural network (DNN) structure usually requires massive data for training a high-accuracy model. To alleviate the cost of using DL models, cloud providers (e.g., Amazon, Alibaba, Microsoft) are now providing Deep Learning as a Service (DLaaS) that offers DL model training and inference APIs for clients. For example,

Google AI¹ provides a series of APIs for AI services (e.g., image classification, personalization recommendation, etc.). By calling these APIs, the client can upload their plaintext data to the cloud, then receive the analysis results (e.g., predication or classification task) by paying certain fees, as shown in **Figure 1**. Due to the fact that users' queries often involve personal privacy information, such as X-ray images or user's behavior trajectory data (Wu et al., 2020), a natural yet essential question about the protection of privacy has been raised: *if massive personal data are collected for model training and prediction, will the disclosing of user-sensitive information increase?* (Riazi et al., 2019; Liu et al., 2020).

Although those cloud providers claim that they will never leak or use users' data for commercial purposes, the increasing number of user data leaks tell us that there is no guarantee on what they promised (Abadi et al., 2016). An intuitive solution to protect user's privacy during DL inference is to give users propriety to download the model from the server and run the model on their platform locally. Nevertheless, this is an undesirable result for the model-holder (e.g., company or hospital) for at least two reasons: (1) The well-trained DL model is considered as the core intellectual property for companies, which is built on the massive collection of data. To avoid the loss of profits, companies require confidentiality to preserve their competitive advantage. (2) The well-trained DL model is known to reveal information about the underlying data used for training. In the case of medical data, this reveals sensitive information about other patients, violating their privacy and perhaps even HIPAA regulations (Assistance, 2003).

Therefore, the target of our work is to design a privacy-preserving service framework where both the model-holder and client can use the well-trained DL model and private data without worries. Two important requirements should be considered:

1. For protecting the privacy of the data owner, their sensitive queries should not be revealed to the model-holder;
2. For the propriety of the model-holder, the DL model should not be revealed to users, in order to preserve their competitive advantage.

Following this mainstream, several solutions based on various secure computing technologies have been proposed, such as homomorphic encryption (HE)-based (Dowlin et al., 2016), multi-party computing (MPC)-based (Rouhani et al., 2018), and mixed-based solutions (Juvekar et al., 2018). Among them, HE (Gentry and Craig, 2009) is an intuitive yet promising way to evaluate it, which considers the whole neural network as a function and evaluates it in the ciphertext domain thoroughly, such as CryptoNet (Dowlin et al., 2016). Secure multi-party computing is another option for secure function evaluation. Secret sharing (SS) (Shamir, 1979) and garbled circuits (GC) (Yao, 1986) are two representational methods. They can transform a neural network model into an oblivious form and evaluate it with secure two-party computation, such as MinONN (Liu et al., 2017). Besides, mixed-based solutions have been proposed to

obtain better performance with trade-off for each advantage, such as Gazzle (Juvekar et al., 2018).

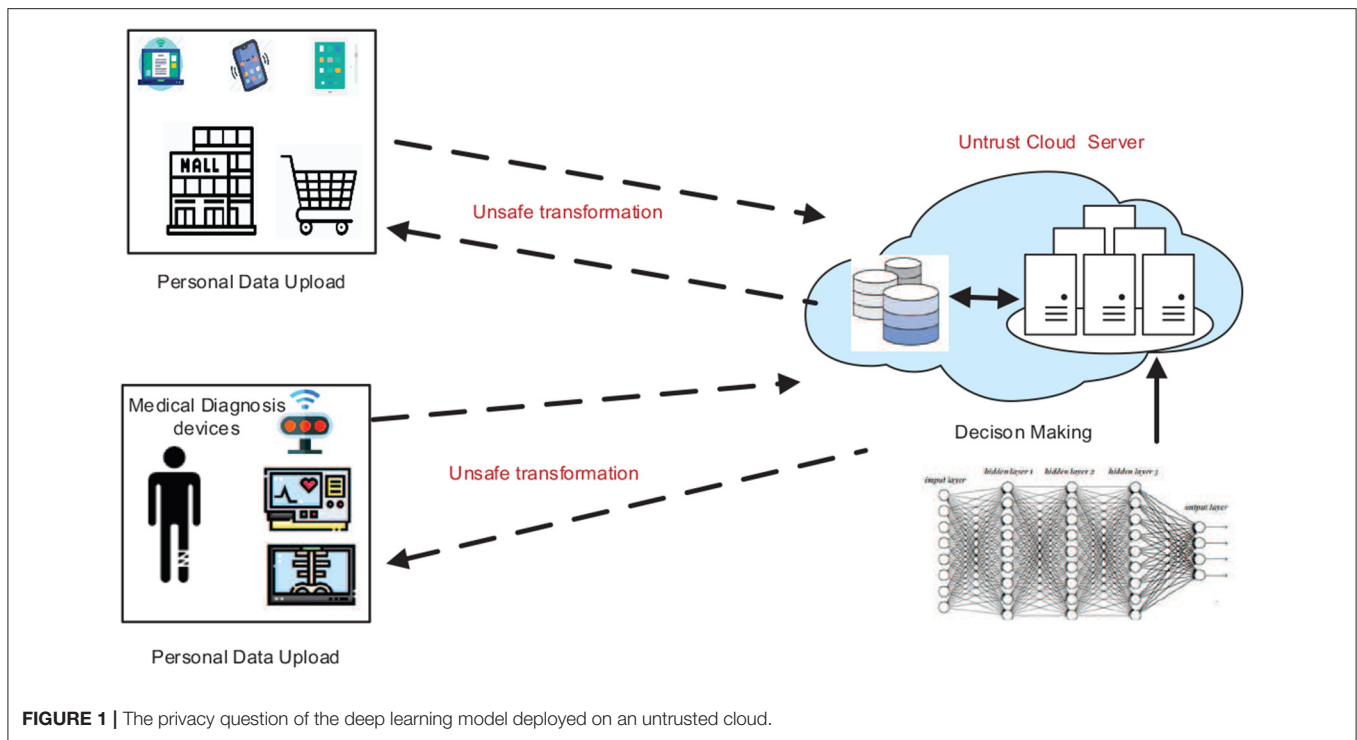
We notice that the CNN inference task requires a lot of inner product operations to finish the convolutional layer. The existing mix-based methods usually adopt the Chinese Remainder Theorem (CRT)-based Single Instruction Multiple Data (SIMD) schemes to execute inner product operations of privacy-preserving CNN. However, it is time-consuming, since rotating operations in privacy-preserving CNN are required to sum up the results among slots. Different with the above solutions, we adopt the GSW-based method to design the matrix multiplication method in the ciphertext space, which is the main motivation of this study. The advantage of the GSW-based solution is that the ciphertext operation is a natural matrix operation without the expensive rotate-and-add strategy. Furthermore, with the rapid development of graphics processing hardware, a GPU is becoming the standard for cloud providers, where CUDA programming makes it possible to harness the computation power of GPU efficiently. Therefore, the use of GPU technology to accelerate matrix multiplication is another important motivation of this study.

On this basis, we introduce cuSCNN, a practical realization of a mixed-based framework that supports the privacy-preserving prediction of convolutional neural networks (CNNs). CNN is one of the most popular neural network architectures in DL. Generally, a CNN model consists of convolutional layers, activation, pooling, and fully connected layers. Convolutional and fully connected layers have linear properties, while activation and pooling are non-linear layers. For cuSCNN, it employs HE to perform the linear operations (e.g., homomorphic addition and multiplication) in each layer, while conducting the non-linear activation functions and pooling operations collaboratively by employing HE and GC jointly. The main contribution of this paper is as follows:

- We propose cuSCNN, an efficient and privacy-preserving neural network prediction framework that keeps user and server data secure. We employ the optimized homomorphic matrix computations for the linear operations in CNN, while adopting GC technology to execute the non-linear operations. Our secure matrix-based computation implements linear operations in the batch mode when dealing with a large-scale dataset.
- We introduce an efficient and natural GSW-based homomorphic matrix encryption scheme to support secure matrix multiplication and addition operations. Furthermore, we propose a hybrid optimization approach to matrix multiplication on GPU to improve the computation efficiency, which combines dual-optimization for I/O and computation.
- We implement cuSCNN on real-world data with varied CNN models and evaluate its performance on the industrial dataset. The experimental results show the superiority and effectiveness of cuSCNN in terms of runtime and power consumption, compared with state-of-the-art works.

The rest of this paper is organized as follows. Section 2 gives the preliminaries. Section 3 overviews the cuSCNN framework.

¹<https://ai.google/>



Section 4 gives the implementation details of the cuSCNN framework. Section 5 evaluates the performance of cuSCNN. Finally, section 6 concludes this paper.

2. PRELIMINARIES

2.1. Related Work

2.1.1. Privacy-Preserving Neural Network Inference Framework

As the representative solution of homomorphic encryption-based solutions, CryptoNets (Dowlin et al., 2016) can evaluate the trained neural network in the ciphertext domain via utilizing leveled homomorphic encryption (LHE). However, the most critical limitation of CryptoNets is that the computational complexity drastically increases as the depth of layers in the NN model increases. Moreover, due to only adopting the LHE, non-linear functionalities such as the ReLU activation function in CryptoNets cannot be supported. To support the non-linear functionalities and pooling operations, DeepSecure (Rouhani et al., 2018) leverages GC as its backbone cryptographic engine. It can support various activations in the DL model. However, since multiplication is an atomic operation in the DL model and the number of Boolean gates in the multiplication circuit grows 2x times concerning the bit width of operands, together with multiple interactions between participants, DeepSecure requires an extensive communication overhead when performing secure privacy-preserving prediction. MiniONN (Liu et al., 2017) transforms a neural network model into an oblivious form and evaluates it with secure two-party computation. In detail, it utilizes the GC to compute the non-linear activation function

while incorporating SS and HE-based methods to run the linear operations in the DNN model. Moreover, GAZELLE (Juvekar et al., 2018) is another mixed-protocol solution that uses an intricate combination of HE and GC to carry out the inference phase of the DNN model, which utilizes the GC to perform the non-linear activation function and uses lattice-based HE with packing technology to execute linear operations. As a result, GAZELLE improves the runtime of private inference and reduces communication between the user and the cloud. To improve the efficiency of the ciphertext computations, FALCON (Li et al., 2020) exploits the Fast Fourier transform to accelerate the homomorphic computations in the convolutional and fully connected layers. Unlike the method mentioned above, we introduce GSW-based secure matrix computations to implement the linear layers and leverage the GPU to accelerate the computation efficiency of the proposed approach.

2.1.2. Matrix-Based Homomorphic Encryption Scheme

Matrix-based computations are the core yet time-consuming operations in the neural network. In this context, some matrix-based homomorphic encryption schemes have been proposed. Based on the SIMD technology, Wu and Haven et al. proposed a safety inner product method on packed ciphertexts (Wu and Haven, 2012). Lu et al. (2016) modified the matrix-vector multiplication for secure statistical analysis over HElib. Duong et al. (2016) proposed a homomorphic matrix multiplication scheme on the packed ciphertext over RLWE. Later, Mishra et al. (2017) designed an enhanced version of the matrix multiplication, but there were useless terms in the ciphertexts. Besides, it is only suitable for a one-depth homomorphic

TABLE 1 | Meaning of notation in the homomorphic encryption scheme.

Notations	The meaning
$\ \mathbf{x}\ _\infty$	The maximum norm of \mathbf{x}
$\ \mathbf{x}\ _2$	The Euclidean norm of \mathbf{x}
$\langle \mathbf{x}, \mathbf{y} \rangle$	The inner product of two vectors \mathbf{x} and \mathbf{y}
x_i	The i th element of vector \mathbf{x}
$[\mathbf{X} \mathbf{Y}] \in \mathbb{Z}^{m \times (n_1+n_2)}$	The column concatenation of \mathbf{X} with \mathbf{Y} , where $\mathbf{X} \in \mathbb{Z}^{m \times n_1}, \mathbf{Y} \in \mathbb{Z}^{m \times n_2}$
$\begin{bmatrix} \mathbf{Y} \\ \mathbf{X} \end{bmatrix} \in \mathbb{Z}^{(m_1+m_2) \times n}$	The row concatenation of $\mathbf{X} \in \mathbb{Z}^{m_1 \times n}$ with $\mathbf{Y} \in \mathbb{Z}^{m_2 \times n}$
\mathbf{X}_i	The i th column vector of \mathbf{X}
$\mathbf{X}(p:q, r:s)$	The submatrix consisting of rows p to q and columns r to s of the matrix \mathbf{X} .
$a \xleftarrow{U} \mathbf{D}$	a is chosen from set \mathbf{D} uniformly at random
\mathbf{I}_r	The identity matrix with size of $r \times r$
$\mathbf{X}_{ij} \in \{0, 1\}^{r \times r}$	The matrix with 1 in the position (i, j) and 0 in the others
λ	Security parameters, the scheme can resist 2^λ attacks
$\text{mod } q$	Modulus q with the range of values is $[-(q-1)/2, (q-1)/2]$
$\text{round}(x)$	Rounding $x \in \mathbb{R}$
$\lceil x \rceil$	Rounding up $x \in \mathbb{R}$
$\lfloor x \rfloor$	Rounding down $x \in \mathbb{R}$

multiplication scenario, due to the significant expansion rate of ciphertexts. Wang et al. (2017) modified Duong's methods for flexible matrix computation, but their modification was much less efficient for matrices of larger size. Jiang et al. (2018) presented a novel matrix encoding method that can encrypt more than one matrix in a single ciphertext and adapted an efficient evaluation strategy for generic matrix operations via linear transformations. However, the methods mentioned above were all constructed based on the second-generation HE scheme with unnecessary key switching, which suffers efficiency and precision loss when dealing with large-scale data. Hiromasa et al. (2016) first conducted a GSW-FHE scheme for matrix homomorphism computations (i.e., HAO). They optimized the bootstrapping technique proposed by Alperin-Sheriff and Peikert (2014). However, all these improvements target binary plaintext, which dramatically restricts its application in the real world.

2.2. Notations and Definitions

Assume that vectors are in column form and are written using bold lower-case letters e.g., \mathbf{x} , while bold capital letters are used to denote matrices, e.g., \mathbf{X} . We introduce gadget matrix \mathbf{G} and the function G^{-1} by lemma 1. In order to facilitate readers to understand, the meanings of the notations mentioned in the encryption scheme are shown in Table 1.

Lemma 1 (Micciancio and Peikert, 2012). Let matrix $\mathbf{C} \in \mathbb{Z}_q^{n \times m}$, there are a fixed and primitive matrix $\mathbf{G} \in \mathbb{Z}_q^{nl \times nl}$ and a deterministic, randomized function G^{-1} that can be calculated by: $\mathbb{Z}_q^{n \times m} \rightarrow \mathbb{Z}_q^{nl \times m}$ such that $\mathbf{X} \xleftarrow{R} G^{-1}(\mathbf{C})$ is sub-Gaussian with parameter $O(1)$ and always satisfies $\mathbf{GX} = \mathbf{C}$.

Let $l = \lceil \log q \rceil + 1$ and $\mathbf{g}^T = (2^0, 2^1, \dots, 2^{l-1})$, \mathbf{I}_n is the unit matrix with n rank, then the gadget matrix can be defined as $\mathbf{G} := \mathbf{I}_n \otimes \mathbf{g}^T \in \mathbb{Z}_q^{n \times nl}$.

2.3. GSW-Based Homomorphic Matrix Encryption Scheme

Generally, a HE scheme consists of four algorithms HE=(Keygen, Enc, Dec, Eval) and can be illustrated as follows:

- **KeyGen(params):** Given the security parameter λ , the main function of **KeyGen(params)** is to produce a secret key \mathbf{sk} , a public key \mathbf{pk} , and a public evaluation key \mathbf{evk} .
- **Enc_{pk}(m):** Based on the created public key \mathbf{pk} , the encryption algorithm encrypts a plaintext $m \in \mathbf{M}$ into a ciphertext $c \in \mathbf{C}$.
- **Dec_{sk}(c):** Using the created secret key \mathbf{sk} , it can recover the original plaintext m from the ciphertext c .
- **Eval_{evk}(f, c₁, ..., c_l):** Under the ciphertext space \mathbf{C} with the evaluation key \mathbf{evk} , the ciphertext c_f can be calculated by using the function $f: \mathcal{M}^l \rightarrow \mathcal{M}$ to c_1, \dots, c_l

The original GSW scheme is proposed by Gentry, Sahai, and Waters (Gentry and Craig, 2009). It adopts the approximate eigenvector method based on the plaintext space \mathbf{M} to construct the ciphertext space \mathbf{C} . Based on this scheme, Bai et al. proposed a homomorphic matrix encryption scheme (Bai et al., 2020), which can be described as follows:

Given the security parameter λ and the multiplication depth of circuit L , $l = \lceil \log q \rceil + 1$. The integer modulus is $q = q(\lambda, L) := 2^{l-1}$, the lattice dimension $n = n(\lambda, L)$, and the noise distribution $\chi = \chi(\lambda, L)$ follows a sub-Gaussian distribution over \mathbb{Z} . Meanwhile, let $m = m(\lambda, L) := O((n+r)l)$, and $N := (n+r)l$. $\mathbf{G} = \mathbf{I}_{n+r} \otimes \mathbf{g}^T \in \mathbb{Z}_q^{(n+r) \times N}$ can be calculated, where $\mathbf{g}^T = \{2^0, 2^1, \dots, 2^{l-1}\}$.

- **HE.KeyGen(n, q, χ, m):** The key generation method mainly includes two parts, i.e., the secret key \mathbf{sk} and public key \mathbf{pk} :
 - For \mathbf{sk} , it first samples a secret key matrix $\tilde{\mathbf{S}} \leftarrow \chi^{r \times n}$, then the secret key matrix can be obtained as follows:

$$\mathbf{S} := [\mathbf{I}_r || -\tilde{\mathbf{S}}] \in \mathbb{Z}_q^{r \times (n+r)} \quad (1)$$

- For \mathbf{pk} , it first generates a uniformly random matrix $\mathbf{A} \xleftarrow{U} \mathbb{Z}_q^{n \times m}$, noise matrix $\mathbf{E} \xleftarrow{R} \chi^{r \times m}$, and $R_{ij} \xleftarrow{U} \{0, 1\}^{m \times N}$ (for all $i, j = 1, \dots, r$), then the public key matrix \mathbf{B} is:

$$\mathbf{B} := \begin{bmatrix} \tilde{\mathbf{S}}\mathbf{A} + \mathbf{E} \\ \mathbf{A} \end{bmatrix} \in \mathbb{Z}_q^{(n+r) \times m} \quad (2)$$

$$\mathbf{P}_{ij} := \mathbf{B}R_{ij} + \begin{bmatrix} \mathbf{M}_{ij}\mathbf{S} \\ 0 \end{bmatrix} \mathbf{G} \in \mathbb{Z}_q^{(n+r) \times N} \quad (3)$$

Hence, the output of keygen(n, q, χ, m) is $\mathbf{sk} := \mathbf{S}$, $\mathbf{pk} := \{\mathbf{P}_{ij}, \mathbf{B} | 1 \leq i, j \leq r\}$.

- **HE.SecEnc(\mathbf{sk}, \mathbf{M}):** Sample the random matrix $\bar{\mathbf{A}} \xleftarrow{U} \mathbb{Z}_q^{n \times N}$ and $\mathbf{E} \xleftarrow{R} \chi^{r \times N}$, then the ciphertext \mathbf{C} can be computed by:

$$\mathbf{C} := \begin{bmatrix} \tilde{\mathbf{S}}\bar{\mathbf{A}} + \mathbf{E} \\ \bar{\mathbf{A}} \end{bmatrix} + \begin{bmatrix} \mathbf{M}\mathbf{S} \\ 0 \end{bmatrix} \mathbf{G} \in \mathbb{Z}_q^{(n+r) \times N} \quad (4)$$

- HE.PubEnc(\mathbf{pk}, \mathbf{M}): Sample a random matrix $R \xleftarrow{U} \{0, 1\}^{m \times N}$, and the ciphertext can be denoted by

$$\mathbf{C} := \mathbf{BR} + \sum_{i=0}^{r-1} \sum_{j=0}^{r-1} M_{i,j} \cdot \mathbf{P}_{i,j} \in \mathbb{Z}_q^{(n+r) \times N} \quad (5)$$

- HE.Dec(\mathbf{S}, \mathbf{C}): The processing of the decryption algorithm can be described as follows:

Step 1: Compute the matrix $\mathbf{H} = \mathbf{SC} \in \mathbb{Z}_q^{r \times N}$;

Step 2: Denote the matrix $\mathbf{H}'_{i,j} \in \mathbb{Z}_q^{r \times rl}$, where $i \in \{1, 2, \dots, r\}$, and $j \in \{1, 2, \dots, rl\}$. Meanwhile, the noise matrix \mathbf{E}' has the same size as \mathbf{H}' . Hence,

$$\mathbf{H}' = \mathbf{E}' + \begin{bmatrix} \mathbf{M}_{0,0} \cdots 2^l \mathbf{M}_{0,0} \cdots \mathbf{M}_{0,r} \cdots 2^l \mathbf{M}_{0,r} \\ \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \\ \mathbf{M}_{r,0} \cdots 2^l \mathbf{M}_{r,0} \cdots \mathbf{M}_{0,r} \cdots 2^l \mathbf{M}_{0,r} \end{bmatrix} \quad (6)$$

Step 3: Recover each element (i.e., $m_{i,j}$) in the plaintext matrix \mathbf{M} via the function $\text{Dec1Num}(\mathbf{H}'(i, jl : (j+1)l - 2))$, where $1 \leq i \leq r$ and $1 \leq j \leq r$. The implementation details of Dec1Num can be found in Bai et al. (2020).

- HE.MatAdd($\mathbf{C}_1, \mathbf{C}_2$): Given the two ciphertext matrices $\mathbf{C}_1 \in \mathbb{Z}_q^{(n+r) \times N}$ and $\mathbf{C}_2 \in \mathbb{Z}_q^{(n+r) \times N}$, the homomorphic matrix addition can be defined as:

$$\mathbf{C}_{add} = \mathbf{C}_1 + \mathbf{C}_2 \in \mathbb{Z}_q^{(n+r) \times N} \quad (7)$$

- HE.MatMult($\mathbf{C}_1, \mathbf{C}_2$): For $\mathbf{C}_1, \mathbf{C}_2 \in \mathbb{Z}_q^{(n+r) \times N}$, it first computes $G^{-1}(\mathbf{C}_2) \in 0, 1^{N \times N}$, then outputs:

$$\mathbf{C}_{mult} := \mathbf{C}_1 \cdot \mathbf{C}_2 = \mathbf{C}_1 G^{-1}(\mathbf{C}_2) \in \mathbb{Z}_q^{(n+r) \times N} \quad (8)$$

To implement the privacy-preserving linear operations in cuSCNN, two kinds of homomorphic computation should be supported: HE.MatAdd and HE.MatMul. HE.Mat means that we can encrypt the plaintext matrix as approximate eigenvalues of the ciphertext matrix correspondingly, where the secret key is the eigenvector. Since the ciphertext calculation of GSW is based on the matrix computation, which cannot cause the expansion of the ciphertext dimension, it can significantly eliminate the unnecessary key conversion brought by BGV-based solutions. HE.MatAdd represents the homomorphic addition between two matrices in the ciphertext domain, while HE.MatMul means the homomorphic multiplication between two matrices.

2.4. GPU-Based Computing

A graphics processing unit (GPU) is a specialized electronic circuit designed to rapidly manipulate and alter memory to accelerate the creation of images in a frame buffer intended for output to a display device. GPU adopts a large number of computing units and ultra-long pipelines, but it only has straightforward control logic and eliminates cache. Their highly parallel structure makes them more efficient than CPUs for algorithms that process large data blocks in parallel. CUDA (an acronym for Compute Unified Device Architecture) is

a parallel computing platform and application programming interface (API) model created by Nvidia, which allows GPU to be compatible with various programming languages (e.g., C++, Fortran, and Python) and applications. The CUDA platform is a software layer that gives direct access to the GPU's virtual instruction set and parallel computational elements to execute compute kernels. In CUDA, *kernels* are functions that are executed on GPU, which are executed by a batch of threads. Meanwhile, the batch of threads is organized as a grid of thread blocks. Thus, a GPU with more blocks can execute a CUDA program in less time than a GPU with fewer blocks. As shown in **Figure 2**, threads in a block are organized into small groups of 32 called *wraps* for execution on the processors, and *wraps* are implicitly synchronous; however, threads in different blocks are asynchronous. CUDA assumes that the CUDA kernel, i.e., CUDA program, is executed on a GPU (drive), and the rest of the C program is executed on the CPU (host). CUDA threads access data from multiple memory hierarchies. Each thread has a private register and local memory, and each thread block has shared memory visible to all threads within the same thread block. All threads can access global memory.

3. THE cuSCNN FRAMEWORK

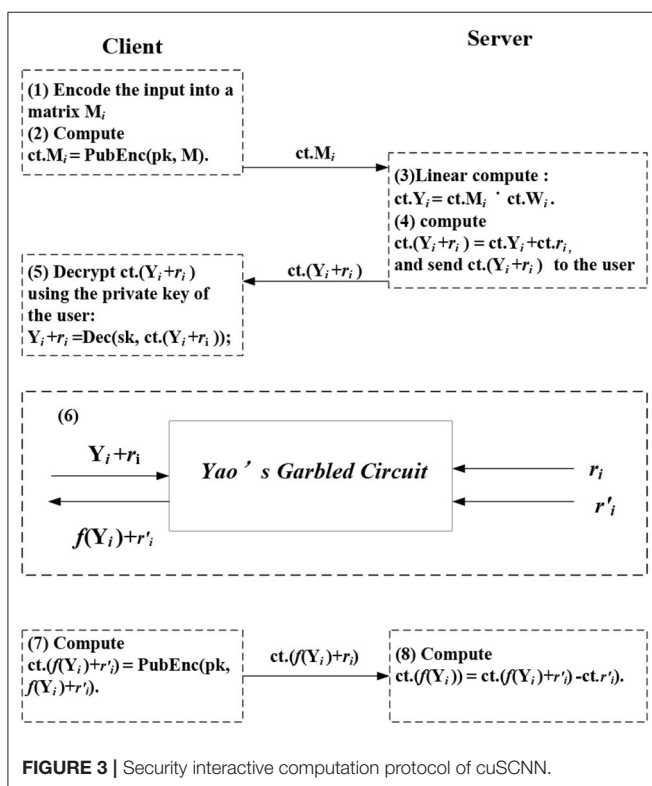
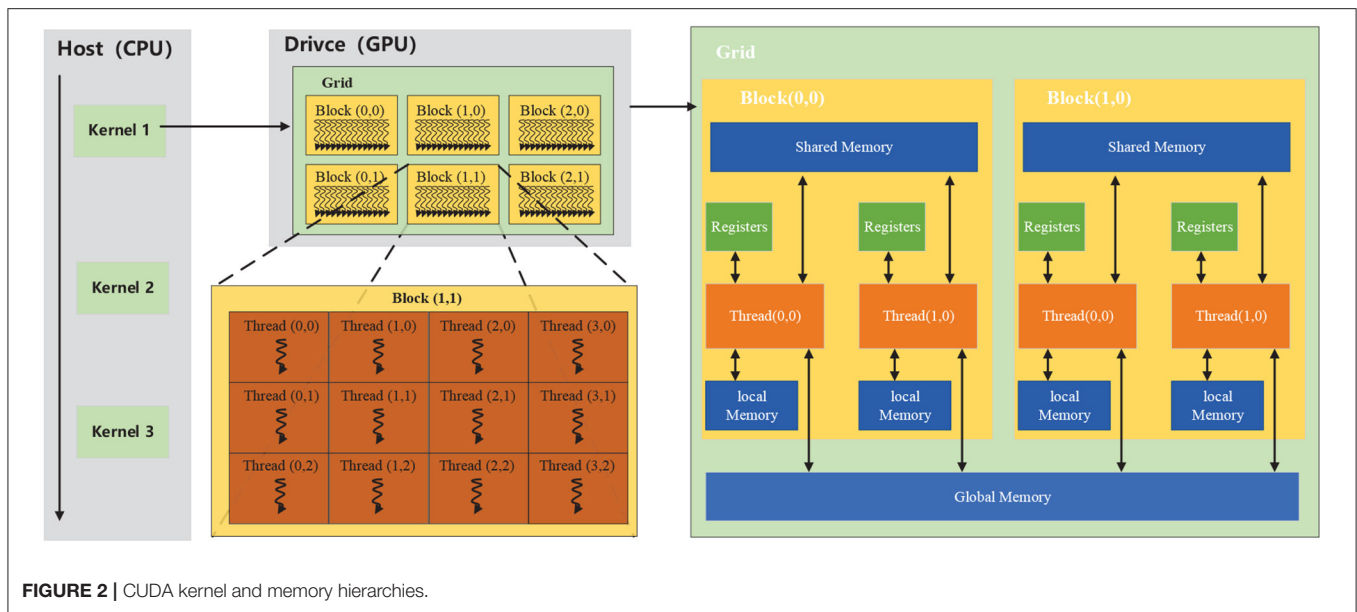
In this section, we design a privacy-preserving CNN prediction framework. Consider a cloud-based medical diagnosis scenario where a user wants to know his health status from an X-ray image. In our setting, we have two roles:

1. The cloud service provider (S) holds trained classifier models and has resources for storage and processing. He has a business interest in computation and making predictions on encrypted data from clients.
2. Clients (C) are the customers of the service provider. He uploads his private image to the cloud server via API interference and receives the result by paying specific service fees.

3.1. Overview

We introduce the execution flow of cuSCNN at a high level. Suppose that client C owns the input data (e.g., an X-ray image) and the server S holds a convolutional neural network model. For client C, the input is private. For server S, the details of the trained CNN model are also private, which includes the weights of convolutional and fully connected layers. The target of cuSCNN is to preserve privacy for the client and server when performing the CNN model.

Hypothesis: Both S and C are *semi-honest* Paverd et al. (2014), we presume they follow the cuSCNN protocol and never deviate from it, although they might attempt to infer more information based on the data they receive and transmit. Specifically, C leaks no information about the input contents, intermediate calculation result, and classified results to the cloud. The input data are factual, never using fake data. For the cloud server, the weights of the CNN model are kept secret from the client, but it does not hide the model architecture.



There are two phases of the framework, including off-line and online phases. In the off-line phase, the cloud generates shares \mathbf{r} and \mathbf{r}' used for Yao's garbled circuit. Besides, the cloud encrypts these shares and their weight matrices using the client's public key. In the online phase, the convolutional and fully connected operations are linear computations, while the

activation functions are non-linear. The execution flow of the proposed cuSCNN is indicated in Figure 3.

- **Evaluate linear layers (i.e., Conv and FC layer):** For $i \in [1, 2, \dots, l]$, l is the number of hidden layers, C firstly encodes the input data into matrix M_i , then it encrypts M_i via calling the public-key encryption algorithm, denoted by $ct.M_i = \text{PubEnc}(pk, M_i)$, and uploads $ct.M_i$ to the cloud server S . S utilizes the encryption scheme to execute matrix-matrix multiplication in convolutional layers and vector-matrix multiplication fully connected layers.

Take the Conv layer, for instance, C feeds the convolutional layer with an encrypted input matrix $ct.M_i$, S computes $Y_i = W_i \cdot ct.M_i$. W is the filter's matrices. The fully connected layer is similar except for homomorphic vector-matrix multiplication.

- **Evaluate non-linear functions (i.e., activation and pooling layer):** S and C perform designed secure computation protocols, i.e., Yao's garbled circuit to keep data secure. Concretely, for layer i , S homomorphically adds encryption of the share \mathbf{r}_i to obtain the encryption of $Y_i + \mathbf{r}_i$, and send it to the client. The client decrypts it using his private key to obtain the plaintext $Y_i + \mathbf{r}_i$. Next, $Y_i + \mathbf{r}_i$ is held by the client and \mathbf{r}_i and \mathbf{r}'_i are held by the server as inputs are conducted in garbled circuit evaluation. The output of it is $f(Y_i) + \mathbf{r}'_i$ (the activation function is denoted by f). Then the client encrypts it using their public key and transmits the ciphertext to the server, the server homomorphically adds the encryption of the share \mathbf{r}'_i to get the encryption of $f(Y)_i$.

3.2. Neural Networks Architecture

In DL, CNN is a popular category of neural network, most commonly applied to analyzing visual imagery. It usually consists of an input and output layer, as well as multiple hidden layers. In most cases, a CNN takes an input and processes it through a sequence of hidden layers to classify it into one of the potential

TABLE 2 | Layers description of CNN.

Layers	Description
Layer-1[Conv-1]	Input image: 28×28 , kernel size: 5×5 , stride: (1,1), number of output channels: 5, padding = VALID, activation = ReLU.
Layer-2[FC-1]	Fully connecting with $5 \times 3 \times 3 = 845$ inputs and 100 outputs, activation = ReLU.
Layer-3[FC-2]	Fully connecting with 100 inputs and 10 outputs activation = softmax.

classes. Hidden layers typically consist of a series of linear (e.g., convolutional, fully connected) layers and non-linear (e.g., activation function and pooling) layers.

For the Conv layer, the convolution operator forms the fundamental basis of the convolutional layer. It has convolutional kernels with size $k \times k$, a stride of (s, s) , and a mapcount of h . Given an image $I \in \mathbb{R}^{w \times w}$ and a convolution kernel $\mathbf{W} \in \mathbb{R}^{k \times k}$, the convolved image $\mathbf{Y} \in \mathbb{R}^{d_k \times d_k}$ can be computed as follows:

$$\mathbf{Y} = \text{Conv}(I, \mathbf{W})_{i',j'} = \sum_{0 \leq i,j \leq k} W_{i,j} \cdot I_{s \cdot i' + i, s \cdot j' + j} \quad (9)$$

where the range of (i', j') is $[0, \lceil \frac{(w-k)}{s} \rceil + 1]$, and $\lceil \cdot \rceil$ denotes the least integer greater than or equal to the input. For multiple kernel cases, it can be expressed as:

$$\mathbf{Y} = \text{Conv}(I, \mathcal{W}) = \left(\text{Conv}(I, \mathbf{W}^{(0)}), \dots, \text{Conv}(I, \mathbf{W}^{(h-1)}) \right) \in \mathbb{R}^{d_k \times d_k \times h} \quad (10)$$

For the FC layers, it connects n_I nodes to n_O nodes, which can form as the matrix-vector multiplication of an $n_O \times n_I$ matrix. Note that the output of the convolutional layer has a form of tensor, so it should be flattened before the FC layer.

4. cuSCNN DESIGN

We next utilize a commonly used CNN in privacy protection work (Dowlin et al., 2016; Rouhani et al., 2018) to describe the design details. The network topology contains one convolutional layer, one fully connected layer with ReLU activation function, and the second fully connected layer applying the softmax activation function for probabilistic classification. **Table 2** describes our neural networks to the MNIST dataset and summarizes the parameters.

4.1. Encryption of Images

We assume that a neural network is trained with the plaintext dataset in the clear. For the CNN architecture in **Table 2**, $w = 28$, $k = 5$, $d_k = 13$, $s = 2$, and $h = 5$. Suppose that the client has a two-dimensional image $I \in \mathbb{Z}^{w \times w}$. For $0 \leq i, j < 5$, $0 \leq i', j' < 13$, by taking the elements $I_{s \cdot i' + i, s \cdot j' + j}$, we extract the image feature to an extended matrix \mathbf{M} with the size of 25×169 . For bias, we add the vector $[1, \dots, 1]^{169}$ to the first

row. For a matrix \mathbf{M} with the size of 26×169 , it is blocked into $b_{num} = 7$ sub-matrices \mathbf{M}_b for parallel computation, where $b_{num} = \lceil N_i / (k^2 + 1) \rceil$, $N_i = d_k \times d_k$. Since this CNN can deal with 846 images in FC-1, we design the framework to compute 846 images at once to achieve this maximum throughput. At the encryption phase, the client C encrypts the \mathbf{M}_b using the public key of a HE scheme.

For $\text{PubEnc}(pk, \mathbf{M}_b)$, we first sample a random matrix $\mathbf{R} \leftarrow \{0, 1\}^{m \times N}$ uniformly, then the encrypted image can be computed by (11).

$$ct.\mathbf{M}_b = \text{PubEnc}(pk, \mathbf{M}_b) := \mathbf{BR} + \sum_{0 \leq i,j \leq r} M[i,j] \cdot \mathbf{P}_{(i,j)} \in \mathbb{Z}_q, \quad (11)$$

4.2. Encryption of Trained Model

The model provider encrypts the trained prediction model values such as multiple convolution kernel values \mathbf{W} and weights (matrices) of FC layers.

The provider begins with a procedure for encrypting the multiple convolutional kernels. Each kernel is extended into a one-row vector of size k^2 , and the bias is connected to the first column. Hence, h kernels are expanded into a matrix with a size of 5×26 . Then the provider pads $(k^2 + 1) - h$ (i.e., 21) rows with zeros to form a square matrix. Finally, the model provider encrypts the plaintext matrix into a ciphertext, denoted by $ct.\mathbf{W}_1$.

Next, the first FC and the second layer are specified by 100×846 and 10×101 matrices. They can pad 746 and 91 rows with zeros to become two square matrices. Then the model provider encrypts the two matrices respectively, and the ciphertexts are $ct.\mathbf{W}_2$ and $ct.\mathbf{W}_3$.

4.3. Homomorphic Evaluation of Neural Networks

The public cloud takes ciphertexts of the images from the data owner and the neural network prediction model from the model provider at the prediction phase. Since the data owner uses a batch of 864 different images, the FC-1 layer is specified as a matrix multiplication: $\mathbb{Z}^{100 \times 846} \times \mathbb{Z}^{846 \times 846} \rightarrow \mathbb{Z}^{100 \times 846}$, and the FC-2 layer is represented as a matrix multiplication: $\mathbb{Z}^{10 \times 101} \times \mathbb{Z}^{101 \times 101} \rightarrow \mathbb{Z}^{10 \times 101}$. The FC-1 layer inputs 846 computational image results to the FC-2 layer, and the FC-2 layer can deal with 101 images at once, so the FC-2 layer needs to execute nine times to finish the 846 image prediction task.

Homomorphic Conv-1 layer: For $0 \leq i < 846$, $0 \leq j < 7$, the public cloud takes the ciphertexts $ct.\mathbf{M}_b^{(i,j)}$ and $ct.\mathbf{W}_1$, and it performs the following computation on ciphertexts:

$$ct.\mathbf{C}_1 \leftarrow \sum_{0 \leq i < 846, 0 \leq j < 7} \text{Mult}(ct.\mathbf{M}_b^{(i,j)}, ct.\mathbf{W}_1) \in \mathbb{Z}_q^{(n+r) \times N}. \quad (12)$$

Secure activation layer: In order to protect the convolutional result \mathbf{Y} and safely compute the activation function, the framework adopts Yao's garbled circuits method similar to GAZALLE and FALCON. The ReLU function is defined by $f(\mathbf{x}) = \max(\mathbf{x}, 0)$, the cloud generates sharing r in the

preprocessing phase, C and S share the input \mathbf{x} additively, i.e., S holds r , while C holds $\max(\mathbf{x}, 0) - r$. The two parties jointly compute GT and MUX circuits to get $f(\mathbf{x}) + r'$, which is sent to C . C loads the 846 images to form a square matrix \mathbf{M}_2 , then we encrypt it into ciphertext $ct.\mathbf{M}_2$ and send it to the cloud.

The FC-1 layer: The cloud firstly performs a homomorphic addition operation to remove sharing, and then it carries out the homomorphic matrix multiplication:

$$ct.\mathbf{C}_2 \leftarrow Mult(ct.\mathbf{M}'_2, ct.\mathbf{W}_2) \in \mathbb{Z}_q^{(n+r) \times N}. \quad (13)$$

Next, the cloud and the user conduct the activation operation by the garbled circuit. Afterward, the user sends the ciphertext $ct.\mathbf{C}_3$ to the cloud.

The FC-2 layer The homomorphic evaluation in FC-2 is similar to FC-1, except for executing nine times to finish 846 image predictions.

$$ct.\mathbf{C}_3^{(i)} \leftarrow Mult(ct.\mathbf{M}'_3^{(i)}, ct.\mathbf{W}_3) \in \mathbb{Z}_q^{(n+r) \times N}, 0 \leq i < 9. \quad (14)$$

The activation operation of FC-2 is a softmax function, since $y_i = \frac{e^{z_i + r'}}{\sum_{j=1}^{num_out} e^{z_j + r'}} = \frac{e^{z_i}}{\sum_{j=1}^{num_out} e^{z_j}}$, where z_i is the i th $i \in [1, num_out]$ input elements of the last fully connected layer, D decrypts the ciphertext and gets the prediction result directly.

Please note that the plaintext of the scheme is a square matrix, and the length of the input vector is set to 846 ($5 \times 13 \times 13 + 1$) in the example. Thus, to maximize the use of plaintext space to improve operating efficiency, we need the number of input images to be 846. In the general case, the number of input images takes the max length of the fully connected layers input vectors in the proposed framework.

4.4. Hybrid Optimization Approach on GPU for Efficient Matrix-Based Computation

To improve homomorphic matrix multiplication efficiency and utilize the powerful computing ability of GPU, we propose a hybrid optimization approach to execute the homomorphic matrix multiplication on GPU.

Given two matrices \mathbf{A} and \mathbf{B} with the size of $r \times r$, the straightforward way is to open a thread for computing each element of its output matrix \mathbf{C} . For parallel matrix multiply operation, each thread loads a row of \mathbf{A} (i.e., $\mathbf{A}(\mathbf{i}, :)$) and a column of \mathbf{B} ($\mathbf{B}(:, \mathbf{j})$), then c_{ij} can be computed via making an inner product of these two vectors (i.e., $c_{ij} = \mathbf{A}(\mathbf{i}, :) \cdot \mathbf{B}(:, \mathbf{j})$). However, the delay in accessing the shared memory on the GPU is quite significant (almost 100 clock cycles). For example, suppose that the matrix elements are stored in the memory following the rows first way, then a row of \mathbf{A} can be saved in the memory continuously, and it can utilize the super large shared memory bandwidth of the GPU to load multiple elements with a short accessing delay. However, for the matrix \mathbf{B} with a large size r , the memory address of elements in a column is internal with r elements. It means that most of the data are useless except the required column of elements in a load time. As a result, the memory access efficiency of this parallel method is appalling, since it is almost impossible for this access mode to hit the cache line.

To address this problem, we introduce a partitioning algorithm for matrix multiplication computation on GPU. For the partition method as shown in **Figure 4A**, the key is to determine *how to maximize the use of limited shared memory space*. The shared memory (SM) is an on-chip cache located on the GPU, which can be as fast as the first level cache, and threads in the same thread block can exchange data through SM. The only disadvantage is that the capacity of SM is limited. To use this small piece of high-speed memory, we divide the matrix into a set of small pieces in each dimension. Suppose that the slice size is T , the output matrix \mathbf{C}_{00} can be written as:

$$\mathbf{C}_{00} = \sum_{i=0}^{bk-1} \mathbf{A}_{0,i} \cdot \mathbf{B}_{i,0} \quad (15)$$

where $bk = \lceil \frac{r}{T} \rceil$ is the block numbers of matrices \mathbf{A} and \mathbf{B} . Note that the small slice matrix will degenerate into a single element when the small slice size T becomes 1. If the small piece is regarded as an element, the size of the whole matrix is reduced by T times.

Each piece of the output matrix \mathbf{C} can assign a thread block with a group of threads to compute the result, where each thread corresponds to an element in the piece. In detail as shown in **Figure 4B**, each thread stores one element of block $\mathbf{B}(:, j)$ and one column of \mathbf{C}_{ij} in its register. $\mathbf{A}(\mathbf{i}, :)$ is stored in the shared memory of Block(0,0), which can be accessed by the threads in Block(0,0). Instead of using the inner product to perform matrix multiplication, we adopt the outer product to optimize the computation. For example, it first performs the outer product between the first column of $\mathbf{A}(:, 0)$ and the first row of $\mathbf{B}(0, :)$ and updates \mathbf{C}_{ij} . Then the \mathbf{C}_{ij} is updated via $\mathbf{A}(:, 1)$ and $\mathbf{B}(1, :)$. Executing the iterations in a similar way until T times, the updated \mathbf{C}_{ij} can be obtained. Finally, each thread stores one column of \mathbf{C}_{ij} from its register to global memory.

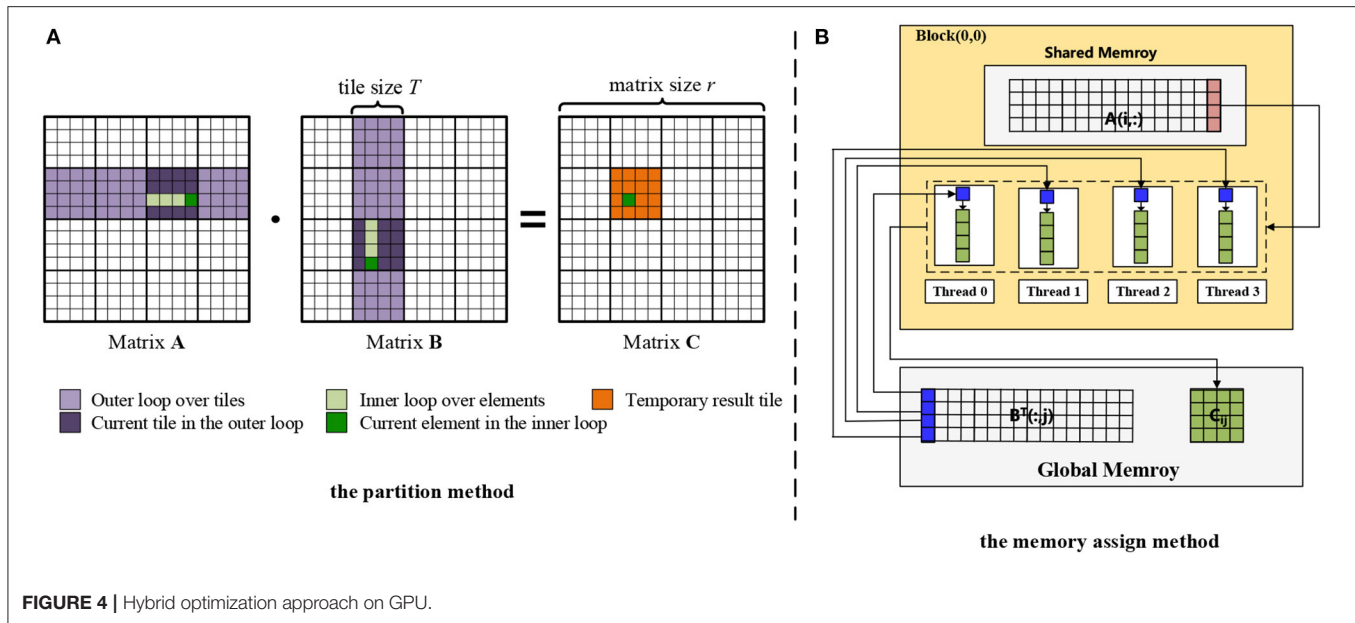
As we know, the time complexity of naive matrix multiplication is $O(r^3)$. Due to leveraging the proposed partition method, the big matrices \mathbf{A} and \mathbf{B} can be divided into bk blocks with slice size T . For each slice, the time complexity is $O(T)$ when calling T threads to perform it in parallel. Hence, the total time complexity of the proposed matrix multiplication on GPU is $O((bk)^2 \times T)$.

4.5. Security Analysis

We prove that the encryption scheme defined above is IND-CPA secure under the LWE hardness assumption.

Theorem 1. For any adversary \mathcal{A} there exists an adversary such that $Adv_{CPA}(\mathcal{A}) < 2Adv_{LWE}(\mathcal{B})$.

Proof: \mathbf{G}_0 : A challenger \mathcal{C} first initializes the encryption scheme and setup parameters, then generates a public key pk and a private key sk . The adversary \mathcal{A} obtains the public key and selects two challenge plaintexts m_0 and m_1 from the plaintext space, and sends them to the challenger \mathcal{C} . \mathcal{C} chooses $b \in [0, 1]$ at random, and encrypts m_b using the public key, then sends the ciphertext to adversary \mathcal{A} . The adversary guesses the plaintext



corresponding to the ciphertext and outputs b' . If $b = b$, the adversary attacks successfully, and the advantage of the adversary is $\text{AdvCPA}(\mathcal{A}) = |\Pr[b = b' \text{ in } G_0] - 1/2|$.

G₁: In G_1 , the public key $pk := P_{(i,j)}, B$ used in G_0 is substituted by a uniform random matrix $B \leftarrow \mathbb{Z}_q^{(n+r) \times m}$, and $P_{(i,j)}$ is substituted by a uniform random matrix $P'_{(i,j)} \leftarrow \mathbb{Z}_q^{(n+r) \times N}$. It is possible to verify that there exists an adversary \mathcal{B} with the same running time, such that $|\Pr[b = b' \text{ in } G_1] - \Pr[b = b' \text{ in } G_0]| \leq \text{AdvLWE}(\mathcal{B})$, since the circular security and LWE assumption, to distinguish B and B' , P and P' for \mathcal{B} is almost impossible.

G₂: In G_2 , the value in the generation of the challenge ciphertext C is substituted with uniform random elements to form matrix C' in G_1 . The adversary distinguishes C and C' which is as hard as solving the LWE problem, so there exists an adversary \mathcal{B} with the same running time as that of $|\Pr[b = b' \text{ in } G_2] \Pr[b = b' \text{ in } G_1]| \leq \text{AdvLWE}(\mathcal{B})$. Notice that in G_2 , the values in C from the challenge ciphertext are independent of bit b , hence, $\Pr[b = b' \text{ in } G_2] = 1/2$.

In summary, $\text{AdvCPA}(\mathcal{A}) < 2\text{AdvLWE}(\mathcal{B})$.

5. EXPERIMENTAL EVALUATION

In this section, we conduct extensive experiments on a real network to evaluate the effectiveness of the proposed cuSCNN. We mainly focus on the following questions (RQs):

- **RQ1:** How the performance of the proposed matrix multiplication method performs;
- **RQ2:** How the proposed homomorphic matrix encryption scheme performs compared to the existing methods;
- **RQ3:** How the performance of cuSCNN on each layer compares to the state-of-the-art networks.

5.1. Experimental Settings

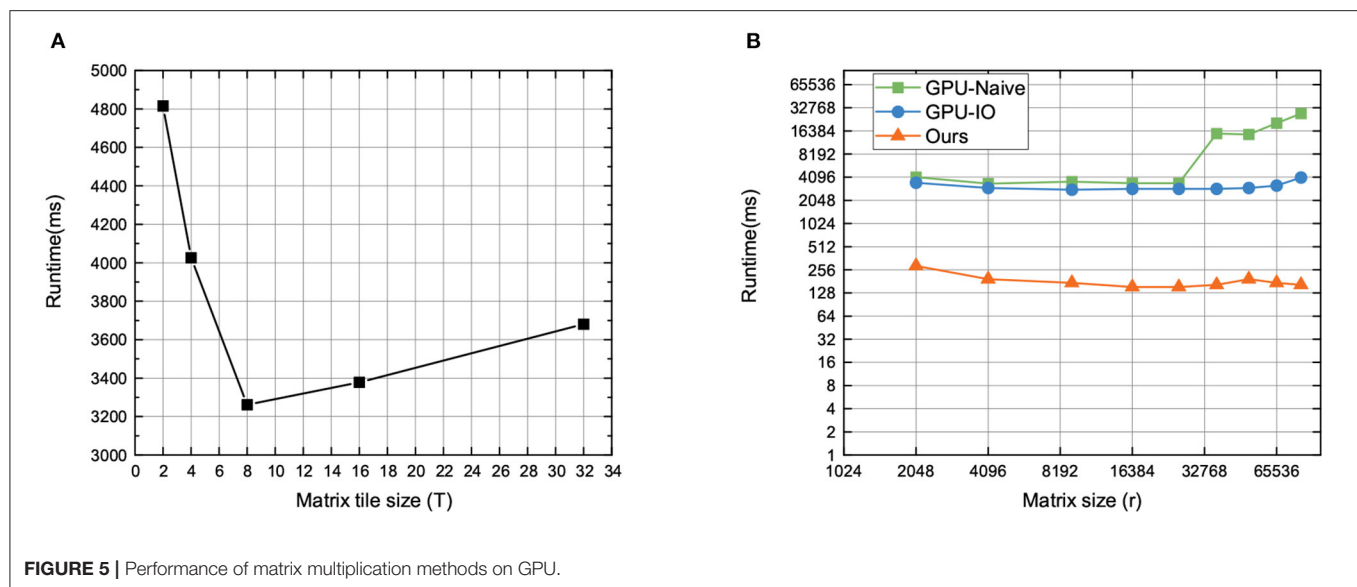
We implement cuSCNN in C++. Specifically, we use cuBLAS library to implement the matrix multiply operations on GPU, and utilize the ABT framework to implement Yao's garbled circuits. For the homomorphic matrix encryption scheme, we set the plaintext module $q = 2^{30}$ (i.e., $l = 30$), which has a 30-bit length and is enough for all the intermediate values. The generation noise follows sub-Gaussian distribution with variance $\text{var} = q/8m$, $n = 600$, the security level can achieve 128.

We tested cuSCNN on two computers, both of which are equipped with Intel Xeon(R) E5-2680 CPU with 4 2.40 Hz cores, and a GeForce GTX 1080Ti GPU. The operation system is CentOS 7.9. One of them worked as client C, and the other play as server S. We took experiments in the LAN setting similar to previous work (Juvekar et al., 2018; Li et al., 2020). Each experiment was repeated 100 times and we report the mean in this paper.

The MNIST database (Modified National Institute of Standards and Technology database) is a dataset of images representing handwritten digits by more than 500 different writers. It is commonly used as a benchmark for machine learning systems. The MNIST database contains 60,000 training images and 10,000 testing images. The format of the images is 28×28 and the integer value of each pixel represents a level of gray with a range 0 to 255. Moreover, each image is labeled with the digit it depicts.

5.2. Performance of Matrix Multiplication on GPU

In this part, we test the timing performance of proposed optimization methods on matrix multiplication, which is the core and time-consuming operation in DL-based applications. In our method, the matrix tile size (T) is a key factor. We set the matrix size to 1,024 (i.e., $r = 1,024$), and the range of



tile size is [2, 4, 8, 16, 24, 32]. The test results are shown in **Figure 5A**. We can observe that the runtime of the proposed method varies with different matrix tiles, and the optimized performance is achieved when the matrix tile size is 8. On the one hand, the inner reason is that the number of threads in each block is decreasing, but the amount of shared memory required in each block is not decreasing, after continuously increasing the matrix tile size. As a result, it will reduce the number of active threads in a streaming multiprocessor (SMP) due to the limited total number of blocks. That is, the occupancy will be reduced. In addition, calculating more elements per thread uses more registers. The number of registers in each thread will in turn affect the number of active threads in SMP, and then affect occupancy.

Then, we evaluate the proposed matrix multiplication method with two baselines on GPU. In detail, we adopt three different methods to execute matrix multiplication, including the naive way (i.e., GPU-Naive), I/O optimization (i.e., GPU-IO) method, and our optimization method. The GPU-Naive method only adopts the straightforward method to perform matrix multiplication, without considering the effect of matrix split and reunion in memory, while the GPU-IO method adopts the block matrix multiplication with matrix split, without considering the matrix reunion in memory. **Figure 5B** is the running time of HE.MatMult with different methods. We find that: (1) Our proposed optimization method has the best effectiveness with varying matrix size, since the running time of our methods is the lowest compared to the other methods; (2) with increasing matrix size, our method can maintain stable execution efficiency with little running time increased. That is because our method can effectively reduce the influence of IO bandwidth on performance by jointly using shared memory and registers. Furthermore, it has a higher computation efficiency via the fine-grained blocking method. Therefore, it can make more efficient use of GPU hardware computing resources.

TABLE 3 | The comparison result of homomorphic matrix encryption schemes.

Matrix size	Method	Enc(s)	HE.MatAdd(s)	HE.MatMult(s)	Dec. (s)
32 × 32	seIMC	6.998	7.345	10.639	0.0768
	Jiang's	0.09	0.01	15.592	0.0543
	Ours	0.679	0.204	0.946	0.067
64 × 64	seIMC	7.82	8.21	12.287	0.312
	Jiang's	0.196	0.01	37.793	0.705
	Ours	0.8	0.233	1.24	0.222
128 × 128	seIMC	9.843	10.402	15.824	1.305
	Jiang's	—	—	—	—
	Ours	1.127	0.291	1.525	0.862

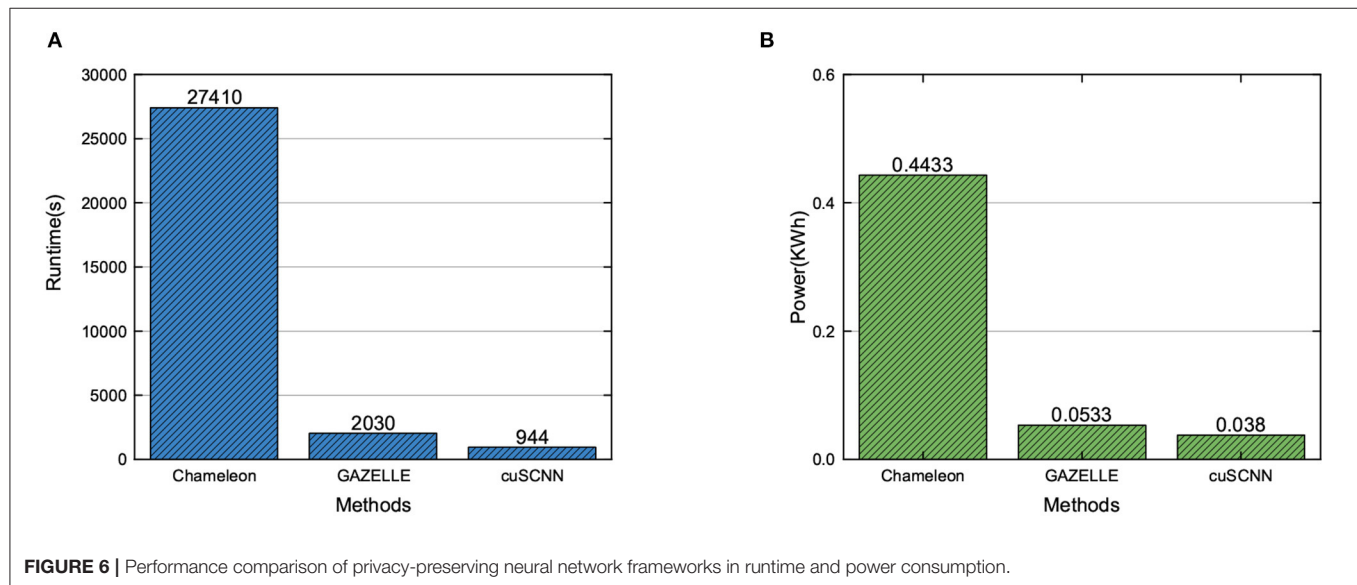
Bold values indicate our methods have a lower running time than the comparison methods.

5.3. Performance of Homomorphic Matrix Encryption Scheme

In this part, we test the performance of our method compared with Jiang's scheme (Jiang et al., 2018) and seIMC (Bai et al., 2020). For Jiang's method, it is a BGV-based secure matrix computation scheme that includes a novel matrix encoding method and an efficient evaluation strategy for basic matrix operations (e.g., matrix addition and multiplication). For seIMC, it is a GSW-based secure matrix computation scheme. We set the security level of seIMC and Jiang to 80 in this experiment. To achieve this security level, the cyclotomic ring dimension of our homomorphic encryption is chosen as $n = 450$, based on the estimator of Albrecht et al. (2015). The parameter settings of Jiang's and SeIMC schemes are the same as in Jiang et al. (2018) and Bai et al. (2020). **Table 3** is the comparison results of the three mentioned secure matrix computation schemes. From the result, we find that: (1) Compared to the BGV-based scheme (i.e., Jiang's scheme) and SeIMC, the running time of our

TABLE 4 | Benchmarks of cuSCNN in Conv and FC layers.

Layer	Input	Filter/output	Setup	Time (ms)		Time per image (ms)
				Online	Total	
Conv layer	(28 × 28 × 1, 846)	(5 × 5 × 1, 5)	2696.9636	0.0074	2696.971	3.19
	(846, 846)	(100, 846)	820.523	0.077	820.6	0.97
FC layer	(101, 846)	(10, 846)	760.109	0.091	760.2	0.9



GSW-based scheme is faster in terms of homomorphic matrix multiplication and decryption. (2) GSW-based solutions can deal with a large-scale matrix, while Jiang's scheme fails to cope with it. Hence, the results demonstrate that our secure matrix computation solution is more suitable for real applications with large-scale data.

5.4. Performance Evaluation for cuSCNN

In this part, we evaluated our cuSCNN framework in an individual layer, and compared it with state-of-the-art methods. By using the proposed homomorphic matrix encryption to secure matrix computations, Conv and FC layers are the main advantage in cuSCNN. For the implementation of cuSCNN, we replace implementations of Conv and FC layers in GAZELLE with proposed optimization methods, while we also adopt the GC to perform the ReLU operation.

Runtime of each layer required for cuSCNN are presented in **Table 4**. Furthermore, we set $T = 8$ for all of the matrix multiplication operations on GPU.

In **Table 4**, we present the timing result of Conv and FC layers with different input sizes. We notice that: (1) Due to adopting the GPU to accelerate the online computing part, the running time of the online part is less than 1 ms either in the Conv layer or FC layer. Hence, the dominant cost of evaluating cuSCNN is that of performing the setup part, including the memory switch between CPU and GPU, the assignment, and initialization operations. (2) Compared to the

FC layer, cuSCNN spends almost $3 \times$ more time executing the Conv layer's convolutional operations.

Finally, we evaluate the performance of the cuSCNN framework on the MNIST dataset, compared to the previous approaches. For comparison with previous approaches, we adopt the same CNN network architecture for all mentioned models. The CNN model takes a gray scale image with size 28×28 as input and has one Conv, two FC, and two ReLU layers. As the comparison framework is performed on a CPU perform, to conduct a fair comparison, we present the runtime (including computation time and communication time) and power consumptions of different models when dealing with 10,152 images. The images are able to predict with 99.1% accuracy. For the power consumption of each approach, we adopt a similar method as proposed in Tian et al. (2018). The compared results are shown in **Figure 6**. From the figure, we can find that: (1) Compared to the existing MPC-based framework, the mixed frameworks can enjoy a better runtime and power consumptions, which can trade-off the advantage of different secure computation technologies, as shown in **Figure 6A**. (2) The performance of cuSCNN outperforms GAZELLE in terms of runtime and power consumption, as shown in **Figure 6B**. That is because cuSCNN adopts the matrix-matrix multiplication to perform the Conv and FC layers, while GAZELLE utilizes the matrix-vector multiplication to finish these layers. Thus, cuSCNN can execute a set of images in one iteration. With the advantage of GPU's

powerful computing ability, cuSCNN designed a hybrid parallel approach to implement the homomorphic matrix computations in Conv and FC layers. Therefore, it demonstrates that cuSCNN has a higher efficiency in executing the privacy-preserving neural networks.

6. CONCLUSION

The increasing popularity of cloud-based deep learning poses a natural question about privacy protection: if massive personal data are collected for model training and prediction, will this result in a rise in disclosing sensitive information? This paper focuses on tackling the privacy-preserving deep learning problem of a client that wishes to classify private images utilizing a convolution neural network (CNN) trained by a cloud server. Our target is to build efficient protocols whereby the cloud server executes the prediction task but also allows both client and model data to remain private. We find that matrix-based computations are the core operations in the neural network prediction task. However, the existing solutions have the limitations of computational efficiency and perform in a serial mode. To track it, this study proposes cuSCNN, a secure and efficient framework to perform the privacy prediction task of a convolution neural network, which utilizes the HE and GC jointly in a batch mode. The hybrid optimization approach is proposed to accelerate the execution of secure matrix computations on GPU to deal with the large-scale dataset. Extensive experiments conducted on the real network show that cuSCNN achieves a better performance on running time and power consumption than the state-of-the-art methods, when dealing with the larger-scale dataset. In the next step, we will conduct comprehensive experiments on different

GPUs to evaluate the performance of the proposed method, including at the server level, desktop level, and embedded levels.

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

YB completed the framework's design, implemented the encryption scheme, and wrote and revised this paper. QL performed the optimization method on GPU for matrix multiplication. WW gave the main idea of the experiment flow design. YF made constructive suggestions on the organization, writing, and revision of the paper. All authors contributed to the article and approved the submitted version.

FUNDING

This work was supported in parts by the National Key Research and Development Project (2020YFA0712303), in parts by Chongqing Research Program (cstc2019yszx-jcyjX0003, cstc2020yszx-jcyjX0005, cstc2021yszx-jcyjX0004), in parts by Guizhou Science and Technology Program ([2020]4Y056) and NSFC (11771421), in parts by Youth Innovation Promotion Association of CAS (2018419), in parts by the Key Cooperation Project of Chongqing Municipal Education Commission (HZ2021017, HZ2021008), in parts by CAS "Light of West China" Program.

REFERENCES

- Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., et al. (2016). "Deep learning with differential privacy," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, (Vienna), 308–318.
- Albrecht, M. R., Player, R., and Scott, S. (2015). On the concrete hardness of learning with errors. *J. Math. Cryptol.* 9, 169–203. doi: 10.1515/jmc-2015-0016
- Alperin-Sheriff, J., and Peikert, C. (2014). "Faster bootstrapping with polynomial error," in *Annual Cryptology Conference* (Santa Barbara, CA: Springer), 297–314.
- Assistance, H. C. (2003). *Summary of the Hipaa Privacy Rule*. Office for Civil Rights (Washington, D.C.).
- Bai, Y., Shi, X., Wu, W., Chen, J., and Feng, Y. (2020). seimc: a gsw-based secure and efficient integer matrix computation scheme with implementation. *IEEE Access* 8, 98383–98394. doi: 10.1109/ACCESS.2020.2996000
- Deng, S., Cai, Q., Zhang, Z., and Wu, X. (2021a). User behavior analysis based on stacked autoencoder and clustering in complex power grid environment. *IEEE Trans. Intell. Transport. Syst.* doi: 10.1109/TITS.2021.3076607
- Deng, S., Chen, F., Dong, X., Gao, G., and Wu, X. (2021b). Short-term load forecasting by using improved gep and abnormal load recognition. *ACM Trans. Intern. Technol.* 21, 1–28. doi: 10.1145/3447513
- Dowlin, N., Giladbachrach, R., Laine, K., Lauter, K. E., Naehrig, M., and Wernsing, J. (2016). "Cryptonets: applying neural networks to encrypted data with high throughput and accuracy," in *Proceedings of the 33rd International Conference on International Conference on Machine Learning* (New York, NY), 48, 201–210.
- Duong, D. H., Mishra, P. K., and Yasuda, M. (2016). Efficient secure matrix multiplication over lwe-based homomorphic encryption. *Tatra Mountains Math. Publ.* 67, 69–83. doi: 10.1515/tmmp-2016-0031
- Gentry and Craig (2009). Fully homomorphic encryption using ideal lattices. *Stoc* 9, 169–178. doi: 10.1145/1536414.1536440
- He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas), 770–778.
- Hiromasa, R., Abe, M., and Okamoto, T. (2016). Packing messages and optimizing bootstrapping in gsw-fhe. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* 99, 73–82. doi: 10.1587/transfun.E99.A.73
- Jiang, X., Kim, M., Lauter, K., and Song, Y. (2018). "Secure outsourced matrix computation and application to neural networks," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security* (Toronto), 1209–1222.
- Juvekar, C., Vaikuntanathan, V., and Chandrakasan, A. (2018). "{GAZELLE}: A low latency framework for secure neural network inference," in *27th {USENIX} Security Symposium ({USENIX} Security 18)* (Baltimore, MD), 1651–1669.
- Li, S., Xue, K., Zhu, B., Ding, C., Gao, X., Wei, D., et al. (2020). "Falcon: a fourier transform based approach for fast and secure convolutional neural network predictions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Seattle, WA), 8705–8714.
- Liu, A., Shen, X., Xie, H., Li, Z., Liu, G., Xu, J., et al. (2020). Privacy-preserving shared collaborative web services qos prediction. *J. Intell. Inform. Syst.* 54, 205–224. doi: 10.1007/s10844-018-0525-4
- Liu, J., Juuti, M., Lu, Y., and Asokan, N. (2017). Oblivious neural network predictions via minioonn transformations, 619–631.

- Lu, W.-J., Kawasaki, S., and Sakuma, J. (2016). Using fully homomorphic encryption for statistical analysis of categorical, ordinal and numerical data. *Cryptol. Arch.* doi: 10.14722/ndss.2017.23119
- Micciancio, D., and Peikert, C. (2012). "Trapdoors for lattices: Simpler, tighter, faster, smaller," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques* (Zagreb: Springer), 700–718.
- Mishra, P. K., Duong, D. H., and Yasuda, M. (2017). "Enhancement for secure multiple matrix multiplications over ring-lwe homomorphic encryption," in *International Conference on Information Security Practice and Experience* (Melbourne: Springer), 320–330.
- Paverd, A., Martin, A., and Brown, I. (2014). *Modelling and Automatically Analysing Privacy Properties for Honest-But-Curious Adversaries*. Univ. Oxford Tech. Rep.
- Riazi, M. S., Rouani, B. D., and Koushanfar, F. (2019). Deep learning on private data. *IEEE Secur. Privacy* 17, 54–63. doi: 10.1109/MSEC.2019.2935666
- Rouhani, B. D., Riazi, M. S., and Koushanfar, F. (2018). "Deepsecure: scalable provably-secure deep learning," in *Proceedings of the 55th Annual Design Automation Conference* (San Francisco), 2, 1–6.
- Shamir, A. (1979). How to share a secret. *Commun. ACM* 22, 612–613. doi: 10.1145/359168.359176
- Shen, D., Wu, G., and Suk, H.-I. (2017). Deep learning in medical image analysis. *Annu. Rev. Biomed. Eng.* 19, 221–248. doi: 10.1146/annurev-bioeng-071516-044442
- Shi, X., He, Q., Luo, X., Bai, Y., and Shang, M. (2020). Large-scale and scalable latent factor analysis via distributed alternative stochastic gradient descent for recommender systems. *IEEE Trans. Big Data.* doi: 10.1109/TBDATA.2020.2973141
- Tian, W., He, M., Guo, W., Huang, W., Shi, X., Shang, M., et al. (2018). On minimizing total energy consumption in the scheduling of virtual machine reservations. *J. Netw. Comput. Appl.* 113, 64–74. doi: 10.1016/j.jnca.2018.03.033
- Wang, L., Aono, Y., and Phong, L. T. (2017). "A new secure matrix multiplication from ring-lwe," in *International Conference on Cryptology and Network Security* (Hong Kong: Springer), 93–111.
- Wu, D., and Haven, J. (2012). Using homomorphic encryption for large scale statistical analysis, FHE-SI-Report. *Univ. Stanford Tech. Rep. TR-dwu4*.
- Wu, D., He, Y., Luo, X., and Zhou, M. (2021a). A latent factor analysis-based approach to online sparse streaming feature selection. *IEEE Trans. Syst. Man Cybern. Syst.* doi: 10.1109/TSMC.2021.3096065
- Wu, D., Luo, X., Shang, M., He, Y., Wang, G., and Wu, X. (2020). A data-characteristic-aware latent factor model for web services qos prediction. *IEEE Trans. Knowl. Data Eng.* doi: 10.1109/TKDE.2020.3014302
- Wu, D., Shang, M., Luo, X., and Wang, Z. (2021b). An l1-and-l2-norm-oriented latent factor model for recommender systems. *IEEE Trans. Neural Netw. Learn. Syst.* doi: 10.1109/TNNLS.2021.3071392
- Yao, A. C. (1986). "How to generate and exchange secrets," in *27th Annual Symposium on Foundations of Computer Science* (Toronto: IEEE), 162–167.

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

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

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



An Adaptive Time-Varying Impedance Controller for Manipulators

Xu Liang^{1,2}, Tingting Su¹, Zhonghai Zhang³, Jie Zhang¹, Shengda Liu², Quanliang Zhao¹, Junjie Yuan¹, Can Huang¹, Lei Zhao¹ and Guangping He^{1*}

¹ Department of Mechanical and Electrical Engineering, North China University of Technology, Beijing, China, ² State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China, ³ Beijing Aerospace Measurement & Control Technology Co., Ltd, Beijing, China

OPEN ACCESS

Edited by:

Di Wu,
Chongqing Institute of Green and
Intelligent Technology (CAS), China

Reviewed by:

Junyong Zhai,
Southeast University, China
Lei Liu,
Liaoning University of Technology,
China
Yuanxin Li,
Liaoning University of Technology,
China

*Correspondence:

Guangping He
hegp55@ncut.edu.cn

Received: 05 October 2021

Accepted: 11 February 2022

Published: 18 March 2022

Citation:

Liang X, Su T, Zhang Z, Zhang J,
Liu S, Zhao Q, Yuan J, Huang C,
Zhao L and He G (2022) An Adaptive
Time-Varying Impedance Controller
for Manipulators.
Front. Neurobot. 16:789842.
doi: 10.3389/fnbot.2022.789842

Aiming at the situation that the structural parameters of the general manipulators are uncertain, a time-varying impedance controller based on model reference adaptive control (MRAC) is proposed in this article. The proposed controller does not need to use acceleration-based feedback or to measure external loads and can tolerate considerable structure parameter errors. The global uniform asymptotic stability of the time-varying closed-loop system is analyzed, and a selection approach for control parameters is presented. It is demonstrated that, by using the proposed control parameter selection approach, the closed-loop system under the adaptive controller is equivalent to an existing result. The feasibility of the presented controller for the general manipulators is demonstrated by some numerical simulations.

Keywords: adaptive, intelligent control, time-varying, human-robot interaction, MRAC

1. INTRODUCTION

The control issues of a multi-degree-of-freedom (multi-DOF) mechanical system with force and motion task constraints are significant for many advanced practical applications, such as minimally invasive surgeries (Burgner-Kahrs et al., 2015), rehabilitation nursing (Jutinico et al., 2017; Ansari et al., 2018), *in-situ* inspection, and machining for the repair of aeroengine parts (Dong et al., 2017; Su et al., 2020), life rescues (McMahan et al., 2006), teleoperation based on haptic interfaces (Sharifi et al., 2016), etc. The operation tasks with force and motion constraint include force-position approximately decoupled operation tasks and more general force-position coupled operation tasks. With regard to the operation task with decoupling force and motion constraint, the closed-loop control system can be stabilized through hybrid force/motion control strategies (Yip and Camarillo, 2016). As to the task with coupling force and motion constraints, in general, an impedance controller has to be utilized to track the time-varying trajectories of the constraint task (Kronander and Billard, 2016). At present, most researches focus on the invariant impedance control of the robot system. The adjustment range of the manipulator's dynamic characteristics under the invariant impedance controller is limited, and it can only complete some rough human-machine cooperation/coordination tasks. For mechanical assembly tasks, especially for those relatively precise assembly operations, such as bearing press-mounting and quantitative fastening of screws/nuts, it is necessary to accurately control the position and pose of the end-effector in the direction of force, as well as the force/torque during the operations. Therefore, the application

of invariant impedance control is relatively limited, and the time-varying impedance control has important practical application requirements in engineering tasks. Since the time-varying impedance control has extensive and important application requirements in complex systems or high-level applications such as the universal operation of industrial robots, the interactive motion of rehabilitation robots, human-machine fusion control of exoskeleton robots, telepresence teleoperation robots, etc., in recent years, scholars have conducted related research on time-varying impedance control. The time-varying impedance closed-loop system is a kind of time-varying dynamic system, and it is difficult to analyze its global or large range asymptotic stability. The integration of robotics and artificial intelligence promotes the development of controllers under time-varying operation tasks (Su et al., 2018; Wu et al., 2021a,b). The theoretical and practical research of artificial intelligence control methods based on fuzzy control, neural network and other theories have been carried out internationally for more than 30 years (Deng et al., 2021a,b; Wu et al., 2021c,e). Artificial intelligence control methods often use large-scale inference rule bases or network structures with a large number of nodes and layers in order to ensure their large-scale effectiveness. Due to their good learning ability, artificial intelligence methods are often used in the cognitive science of human-machine interaction systems (Wu et al., 2020b, 2021d).

In practical engineering applications, the control systems often encounter comprehensive characteristics such as strong non-linearity, uncertainty, and time-varying parameters (Liu et al., 2020; Liang et al., 2021b), which will affect the stability of the system. Because the accurate dynamic modeling of the robot system is rather hard, which brings difficulties to the control law design of the system and reduces the dynamic characteristics of the closed-loop system, it is difficult for the robot to achieve high-quality practical applications. Adaptive control and its improvement (Tong et al., 2020; Liu et al., 2021b), sliding mode control and its improvement (Zhai and Xu, 2021), non-linear feedback control and its improvement, observers and its improvement (Liang et al., 2021a; Li et al., 2021; Liu et al., 2021a), and other methods (Yang et al., 2021) can be used to solve this problem. In literature (Li, 2021), a novel command filter adaptive tracking controller is designed to achieve asymptotic tracking for a class of uncertain non-linear systems with time-varying parameters and uncertain disturbances by introducing a smooth function with positive integrable time-varying function to compensate the unknown time-varying parameters and uncertain disturbances. In this article, we study adaptive time-varying impedance controllers. In recent years, adaptive impedance control problems have attracted the attention of many scholars due to the wide and different application requirements (Xu et al., 2011; Jamwal et al., 2017), such as the relevant developments about haptic interfaces (Sharifi et al., 2016), upper/lower limb rehabilitation robots (Li et al., 2017; Liu et al., 2017), robotic exoskeleton systems (Hussain et al., 2013), and so on (Wu et al., 2020a; Deng et al., 2021b). At present, most research studies on adaptive impedance control are actually focused on online “impedance

planning,” which means online searching for a target impedance profile for the purpose of improving the application effects of robots. The stability issues of the time-varying closed-loop systems with regard to the target impedance profile are not analyzed except few works (Ferraguti et al., 2013; Kronander and Billard, 2016). In some application-oriented research studies, experiments are always used to demonstrate the stability of the controlled plants (Hamedani et al., 2019; Pena et al., 2019; Perez-Ibarra et al., 2019). However, demonstrating the stability of an adaptive impedance control system by experiments is commonly task-dependent, and different operating tasks require different experiments to verify the stability of the system. Therefore, an analysis or control method that can ensure the stability of the time-varying impedance control system is required. To this end, the literature (Kronander and Billard, 2016) and (Ferraguti et al., 2013) addressed this issue. Through an in-depth analysis of the method presented in Ferraguti et al. (2013), the literature (Kronander and Billard, 2016) presented the stability conditions for the variable damping and stiffness system, and the proposed stability conditions do not rely on the controlled plant's states. The benefit of the stability conditions is that they can be verified offline before performing a task. However, this approach has two main shortcomings: (1) accurate dynamics model of the controlled plant is needed in the controller; and (2) measurement of external loads or joint accelerations is required in the controller.

In this article, aiming at the above two problems, a globally uniform stability condition is proposed in which the variable damping and stiffness are independent of the state of the robot. As we all know, the closed-loop system under an adaptive impedance controller is actually a time-varying dynamic system, and it is also a complex non-linear system, which makes it difficult to design the controller. To be specific, the main contribution of this article is summarized as follows:

- 1) In this article, we use variable damping and stiffness control to adjust damping and stiffness parameters to improve compliant operation performance and use adaptive control to adjust the parameters to achieve the stability of the system when the system parameters are disturbed.
- 2) Under the frame of model reference adaptive control (MRAC), the rigorous canonical reduction form of the dynamics of the general robot system can be transformed into linear or special non-linear “recursive canonical form.” By using the recursive canonical expression and the design method of the time-varying impedance controller of the linear system, the analytical expression of the parameter adaptive regulation law can be obtained, and the time-varying impedance controller with parameter adaptive characteristics can be designed.
- 3) The time-varying impedance controller is reconstructed under the frame of MRAC, and the stability condition given in Kronander and Billard (2016) remains unchanged. Therefore, the stability condition under the adaptive control frame is still state independent, while the above two shortcomings are eliminated.

The remainder of this article is organized as follows. Section 2 presents the stability condition provided by Kronander and Billard (2016), since it does not depend on any controllers. Section 3 presents our main contributions, the time-varying impedance controller based on MRAC and a control parameter selection approach. As an example, the method is tested through an uncertain planar 2R manipulator in section 4. Section 5 gives the conclusions.

2. EXISTING TIME-VARYING IMPEDANCE CONTROLLER FOR MANIPULATORS

In general, the dynamic equation of the manipulators has the following form

$$M(\Theta)\ddot{\Theta} + C(\Theta, \dot{\Theta})\dot{\Theta} + N(\Theta) - \tau_e = \tau_a \quad (1)$$

where $\Theta \in \mathbb{R}^n$ represents the generalized coordinates of the manipulator in configuration space, $M(\Theta) \in \mathbb{R}^{n \times n}$ represents the inertial matrix of the system, $C(\Theta, \dot{\Theta})\dot{\Theta} \in \mathbb{R}^n$ represents centrifugal and Coriolis torque vector, $N(\Theta) \in \mathbb{R}^n$ represents the gravity and elastic force vector, $\tau_e \in \mathbb{R}^n$ represents an equivalent torque caused by the external forces, while $\tau_a \in \mathbb{R}^n$ represents the actuation torque.

For time-varying impedance control issues, the closed-loop target dynamic equation of a manipulator can be given as follows

$$H\ddot{\bar{\Theta}} + D(t)\dot{\bar{\Theta}} + K(t)\bar{\Theta} = \tau_e \quad (2)$$

where $\bar{\Theta} = \Theta - \Theta^d$ is defined to be an error vector of the generalized coordinates and Θ^d denotes the desired position of the generalized coordinates, H denotes a positive definite and symmetric constant matrix, $D(t)$ denotes a time-varying damping matrix, and $K(t)$ denotes a time-varying stiffness matrix. Both of $D(t)$ and $K(t)$ are also positive definite and symmetric. Usually, $K(t)$ should be determined by the designated operation tasks, and $D(t)$ should be selected to ensure the global asymptotic stability at the origin $(\bar{\Theta}, \dot{\bar{\Theta}}) = (0, 0)$ of the closed-loop system (2) when the equivalent external torque satisfies $\tau_e = 0$. If the equivalent external torque τ_e does not equal to zero, then the origin $(\bar{\Theta}, \dot{\bar{\Theta}}) = (0, 0)$ of the closed-loop system (2) should be globally stable in Lyapunov's sense. An elegant result of designing a time-varying impedance controller of the manipulator can be stated as Lemma 1, which is an adapted result that was first presented in the literature (Kronander and Billard, 2016).

Lemma 1. *For the dynamic systems (1) and the target system (2), suppose the stiffness matrix $K(t)$ is continuous, then $\dot{K}(t)$ is bounded, which means $\|\dot{K}(t)\| \leq \Omega$, where Ω is a positive constant. Then there exists a positive constant α and a matrix $D(t)$ satisfying the following set of inequalities*

$$\begin{cases} \alpha > 0 \\ K(t) + \alpha D(t) - \alpha^2 H > 0 \\ -D(t) + \alpha H < 0 \\ \dot{K}(t) + \alpha \dot{D}(t) - 2\alpha K(t) < 0 \end{cases} \quad (3)$$

which makes the following closed-loop system

$$\begin{cases} M(\Theta)\ddot{\Theta} + C(\Theta, \dot{\Theta})\dot{\Theta} + N(\Theta) - \tau_e = \tau_a \\ \tau_a = M\ddot{\bar{\Theta}} + C\dot{\bar{\Theta}} + N + (M - H)\ddot{\bar{\Theta}} + [C - D(t)]\dot{\bar{\Theta}} \\ -K(t)\bar{\Theta} \end{cases} \quad (4)$$

globally uniformly asymptotically stable at the origin $(\bar{\Theta}, \dot{\bar{\Theta}}) = (0, 0)$ when $\tau_e = 0$. When $\tau_e \neq 0$, then the origin $(\bar{\Theta}, \dot{\bar{\Theta}}) = (0, 0)$ is globally uniformly stable.

REMARK 1. By applying a Lyapunov candidate function $V(\dot{\bar{\Theta}}, \bar{\Theta}, t) = \frac{1}{2}(\dot{\bar{\Theta}} + \alpha\bar{\Theta})^T H(\dot{\bar{\Theta}} + \alpha\bar{\Theta}) + \frac{1}{2}\bar{\Theta}^T \beta(t)\bar{\Theta}$ with the time-varying function definition $\beta(t) = K(t) + \alpha D(t) - \alpha^2 H$, it is not hard to show that the first two inequalities in (3) are used to ensure the positive definiteness of Lyapunov function $V(\dot{\bar{\Theta}}, \bar{\Theta}, t)$, and the last two inequalities in (3) can ensure the negative definiteness of $\dot{V}(\dot{\bar{\Theta}}, \bar{\Theta}, t)$. Furthermore, by proving the function $V(\dot{\bar{\Theta}}, \bar{\Theta}, t)$ is also a decrescent function, then the global uniform asymptotic stability of the closed-loop system (2) can be concluded. For the purpose of simplifying control parameters selection, in He et al. (2020) the authors presented a simple stability condition

$$D(t) = \alpha H + \varepsilon I \quad (5)$$

where $\varepsilon > 0$ is a small constant and I denotes an identity matrix. Even though the damping matrix given in (5) shows certain conservatism for some applications, it is sufficient to show that the solution of the inequality group (3) exists.

REMARK 2. Note that the torque controller $\tau_a(t)$ in (4) uses acceleration feedbacks, and the dynamics model (1) is supposed to be accurate. In real world applications, these two points may not be easily achieved, since the acceleration sensors are not standard accessories for many manipulators and it is also rather difficult to accurately determine the dynamics parameters of a multi-DOF mechanical system. In the next section, it will be shown that these problems can be resolved by developing an MRAC based time-varying impedance controller.

3. A MRAC BASED TIME-VARYING IMPEDANCE CONTROLLER FOR MANIPULATORS

For a controlled system with an adaptive controller, in general, the uniform asymptotic stability of the closed-loop system cannot be concluded by following the same method as that provided in Remark 1. The main reason is that a parameter estimation law is also included in the closed-loop system besides a control law, such that the Lyapunov candidate function cannot be constructed as that presented in Remark 1. On the contrary, the following lemma (Slotine and Li, 1991) can be utilized to analyze the uniform asymptotic stability of a closed-loop system with an adaptive controller.

LEMMA 2. *If a scalar function $V(t)$ has the following properties, then $\lim_{t \rightarrow \infty} \dot{V}(t) \rightarrow 0$.*

- (1). $V(t)$ is lower bounded;
- (2). $\dot{V}(t)$ is negative semi-definite;
- (3). $\dot{V}(t)$ is uniformly continuous in time.

Now, we derive the adaptive time-varying impedance controller. First, we define a virtual velocity error vector

$$s = \dot{\Theta} + \Lambda \bar{\Theta} = \dot{\Theta} - \dot{\Theta}^d + \Lambda \bar{\Theta} = \dot{\Theta} - \dot{\Theta}_r \quad (6)$$

where $\Lambda \in \mathbb{R}^{n \times n}$ is a symmetric and positive definite matrix, or more generally a matrix so that $-\Lambda$ is Hurwitz, $\bar{\Theta} \in \mathbb{R}^n$, and the virtual reference velocity $\dot{\Theta}_r \in \mathbb{R}^n$ in Equation (6) is defined as

$$\dot{\Theta}_r = \dot{\Theta}^d - \Lambda \bar{\Theta}. \quad (7)$$

It is well known that the dynamics of a mechanical system commonly satisfies the linearly parameterized property, that is, the left-hand side of the dynamic system (1) can be expressed as the following form

$$M(\Theta)\ddot{\Theta} + C(\Theta, \dot{\Theta})\dot{\Theta} + N(\Theta) - \tau_e = \chi(\Theta, \dot{\Theta}, \ddot{\Theta})\rho \quad (8)$$

where $\chi(\Theta, \dot{\Theta}, \ddot{\Theta})$ denotes a matrix, ρ denotes an unknown parameter vector that describes the mass properties of a mechanical system. If we replace the differential variables $\dot{\Theta}$ and $\ddot{\Theta}$ of the system (1) with the virtual reference velocity $\dot{\Theta}_r$ and its differential variable $\ddot{\Theta}_r$, then the linearly parameterized property does not change, and the resulted virtual dynamic system can also be expressed as a similar form

$$M(\Theta)\ddot{\Theta}_r + C(\Theta, \dot{\Theta})\dot{\Theta}_r + N(\Theta) - \tau_e = \chi(\Theta, \dot{\Theta}, \ddot{\Theta}_r)\rho. \quad (9)$$

By applying the linearly parameterized form Equation (9), we can obtain the following result.

THEOREM 1. For the dynamic systems (1), by applying the following controller

$$\tau_a = \chi(\Theta, \dot{\Theta}, \ddot{\Theta}_r)\hat{\rho} - K_D s \quad (10)$$

and the following parameter estimator

$$\dot{\hat{\rho}} = -\Gamma^{-1} \chi^T s \quad (11)$$

where K_D in Equation (10) is a continuous positive definite matrix, i.e., K_D is bounded, $\hat{\rho}$ denotes the estimation of ρ , and the matrix Γ in Equation (11) is also positive definite, then the origin $(\bar{\Theta}, \dot{\bar{\Theta}}) = (0, 0)$ of the closed-loop system

$$\begin{cases} M(\Theta)\ddot{\Theta} + C(\Theta, \dot{\Theta})\dot{\Theta} + N(\Theta) - \tau_e = \tau_a \\ \tau_a = \chi(\Theta, \dot{\Theta}, \ddot{\Theta}_r)\hat{\rho} - K_D s \\ \dot{\hat{\rho}} = -\Gamma^{-1} \chi^T s \end{cases} \quad (12)$$

is globally uniformly asymptotically stable when the external loads $\tau_e = 0$. If the external loads $\tau_e \neq 0$, the origin $(\bar{\Theta}, \dot{\bar{\Theta}}) = (0, 0)$ of the system Equation (12) is globally uniformly stable in the Lyapunov's sense.

PROOF. Let us define $\bar{\rho} = \hat{\rho} - \rho$ to be an error vector of the parameter estimates $\hat{\rho}$ and select a Lyapunov candidate function

$$V(t) = \frac{1}{2} (s^T M s + \bar{\rho}^T \Gamma \bar{\rho}). \quad (13)$$

By using the definition of the virtual velocity error vector given by Equation (6), the time derivative of Equation (13) can be given as

$$\dot{V}(t) = s^T \dot{M} s + \frac{1}{2} s^T \dot{M} s + \bar{\rho}^T \Gamma \dot{\bar{\rho}} = s^T (M\ddot{\Theta} - M\ddot{\Theta}_r) + \frac{1}{2} s^T \dot{M} s + \bar{\rho}^T \Gamma \dot{\bar{\rho}}. \quad (14)$$

Since $\dot{M} - 2C$ is a skew-symmetric matrix (Murray et al., 1994), which means that $(\dot{M} - 2C)^T = -(\dot{M} - 2C)$, we have

$$\dot{M}^T + \dot{M} = 2C^T + 2C \quad (15)$$

and since M is a symmetric and positive definite (Murray et al., 1994), which means that $M = M^T$, then from Equation (15) we can get

$$\dot{M}^T + \dot{M} = 2\dot{M} = 2C^T + 2C \quad (16)$$

So

$$\dot{M} = C + C^T \quad (17)$$

By using the equation above, Equation (14) can be written as

$$\dot{V}(t) = s^T (M\ddot{\Theta} - M\ddot{\Theta}_r) + \frac{1}{2} s^T (C + C^T) s + \bar{\rho}^T \Gamma \dot{\bar{\rho}}. \quad (18)$$

Referring to the dynamics Equation (1), it is easy to obtain

$$M\ddot{\Theta} = \tau_a - C\dot{\Theta} - N + \tau_e \quad (19)$$

and from (6) we can obtain

$$\dot{\Theta} = s + \dot{\Theta}_r. \quad (20)$$

Substituting Equations (20) into (19) and then bringing Equations (19) into (18), it can be shown that

$$\begin{aligned} \dot{V}(t) &= s^T [\tau_a - M\ddot{\Theta}_r - C(s + \dot{\Theta}_r) - N + \tau_e] + \frac{1}{2} s^T (C + C^T) s + \bar{\rho}^T \Gamma \dot{\bar{\rho}} \\ &= s^T [\tau_a - M\ddot{\Theta}_r - C\dot{\Theta}_r - N + \tau_e] + \bar{\rho}^T \Gamma \dot{\bar{\rho}}. \end{aligned} \quad (21)$$

Due to $\bar{\rho} = \hat{\rho} - \rho$ and ρ is a constant for any manipulator system, we have $\dot{\bar{\rho}} = \dot{\hat{\rho}}$. Therefore, Equation (21) follows that

$$\dot{V}(t) = s^T [\tau_a - M\ddot{\Theta}_r - C\dot{\Theta}_r - N + \tau_e] + \bar{\rho}^T \Gamma \dot{\hat{\rho}}. \quad (22)$$

By applying the linearly parameterized form Equation (9), Equation (22) can be expressed as

$$\dot{V}(t) = s^T [\tau_a - \chi(\Theta, \dot{\Theta}, \ddot{\Theta}_r)\rho] + \bar{\rho}^T \Gamma \dot{\hat{\rho}}. \quad (23)$$

If we adopt the controller Equation (10), it is straightforward that the Equation (23) can be rewritten as

$$\dot{V}(t) = s^T [\chi \hat{\rho} - K_D s - \chi \rho] + \bar{\rho}^T \Gamma \dot{\hat{\rho}} = s^T \chi \bar{\rho} - s^T K_D s + \bar{\rho}^T \Gamma \dot{\hat{\rho}}. \quad (24)$$

By using the parameter estimator Equation (11), which is given as $\dot{\hat{\rho}} = -\Gamma^{-1} \chi^T s$, then we can obtain

$$\dot{V}(t) = -s^T K_D s \leq 0 \quad (25)$$

since K_D is positive definite. This implies $V(t) \leq V(0)$, and therefore, both of the vectors s and $\bar{\rho}$ are bounded [see Equation (13)]. To observe the uniform continuity of the function $\dot{V}(t)$, we calculate the second order differential function of $V(t)$, and it can be written as

$$\ddot{V}(t) = -2s^T K_D \dot{s} - s^T \dot{K}_D s. \quad (26)$$

See definition Equation (6), it shows the vector s is smooth. On the other hand, the differential matrix \dot{K}_D is supposed to be bounded. Then we can conclude that $\ddot{V}(t)$ is bounded. According to Lemma 2, we can get $\lim_{t \rightarrow \infty} \dot{V}(t) \rightarrow 0$, which means $s \rightarrow 0$ as $t \rightarrow \infty$. It is obvious that \dot{s} is bounded.

On the surface $s = 0$, referring to the definition $s = \dot{\Theta} + \Lambda \bar{\Theta}$, we can conclude the origin $(\bar{\Theta}, \dot{\bar{\Theta}}) = (0, 0)$ of the closed-loop system Equation (12) is uniformly asymptotically stable since $-\Lambda$ is Hurwitz. In addition, the function $V(t)$ is unbounded, thus the stability of the closed-loop system is globally effective.

REMARK 3. Theorem 1 shows that both the control law Equation (10) and the parameter estimator (11) only use state feedback $s = \dot{\Theta} + \Lambda \bar{\Theta}$. This is helpful for improving the feasibility of the controller in real world applications. In particular, the adaptive controller does not need an accurate dynamic model, thus better robust stability of the closed-loop system Equation (12) could be expected.

REMARK 4. Even though Theorem 1 gives an adaptive controller for the dynamic system Equation (1), so far the adaptive controller is not related to the time-varying impedance control issues of the manipulators. By using the following result, we can get that the time-varying impedance control problems can be resolved under the adaptive control strategy.

THEOREM 2. If the control parameters Λ and K_D of the adaptive controller Equation (10) are chosen as

$$\Lambda = \gamma M^{-1}, K_D = \frac{1}{\gamma} K(t)M - C \quad (27)$$

where $\gamma > 0$ is a constant, then the origin $(\bar{\Theta}, \dot{\bar{\Theta}}) = (0, 0)$ of the closed-loop system Equation (12) is globally uniformly stable in Lyapunov's sense.

PROOF. By subtracting Equation (9) from Equation (1), we have

$$M\dot{s} + Cs = \tau_a - \chi\rho \quad (28)$$

where $s = \dot{\Theta} - \dot{\Theta}_r$ is considered. Then, substituting the adaptive control law Equation (10) into Equation (28), we can obtain that

$$M\dot{s} + (C + K_D)s = \chi\bar{\rho} \quad (29)$$

where $\bar{\rho} = \hat{\rho} - \rho$ is considered. Since the vector also satisfies the relationship $s = \dot{\Theta} + \Lambda \bar{\Theta}$, we can obtain

$$M\ddot{\bar{\Theta}} + (M\Lambda + C + K_D)\dot{\bar{\Theta}} + (C + K_D)\Lambda\bar{\Theta} = \chi\bar{\rho}. \quad (30)$$

According to Equation (27), if we select $\Lambda = \gamma M^{-1}$ and $K_D = \frac{1}{\gamma} K(t)M - C$, then Equation (30) can be written as

$$M\ddot{\bar{\Theta}} + D(t)\dot{\bar{\Theta}} + K(t)\bar{\Theta} = \chi\bar{\rho} \quad (31)$$

where

$$D(t) = M\Lambda + C + K_D = \gamma I + \frac{1}{\gamma} K(t)M. \quad (32)$$

Comparing Equation (32) with Equation (5), it shows the damping matrix given by (32) satisfies the stability condition Equation (5) if we select $\alpha = \frac{1}{\gamma} K(t)$ and $\varepsilon = \gamma$. In addition, on the basis of Theorem 1, under the control law (10) and the parameter estimator Equation (11), the error vector $\bar{\rho}$ is bounded. According to Lemma 1, the origin $(\bar{\Theta}, \dot{\bar{\Theta}}) = (0, 0)$ of the closed-loop system Equation (31) is globally uniformly asymptotically stable when $\bar{\rho} = 0$. If the error vector $\bar{\rho} \neq 0$, the origin $(\bar{\Theta}, \dot{\bar{\Theta}}) = (0, 0)$ of the system Equation (31) is globally uniformly stable in Lyapunov's sense.

REMARK 5. It is worth noting that, in Equation (2), the inertial matrix H is generally different from the inertial matrix M , so that the term $(M - H)\ddot{\Theta}$ is appeared in the controller Equation (4), and then an accelerated feedback or sensing the external loads τ_e is necessary. If we select $H = M$, the closed-loop system Equation (4) is given by

$$M\ddot{\bar{\Theta}} + D(t)\dot{\bar{\Theta}} + K(t)\bar{\Theta} = \tau_e \quad (33)$$

which is very similar to the adaptive control law based closed-loop system Equation (31). However, if the dynamics model Equation (1) is not accurate, then the error terms $\Delta M(\Theta)\ddot{\Theta}$, $\Delta C(\Theta, \dot{\Theta})\dot{\Theta}$, and $\Delta N(\Theta)$ will appear in the closed-loop system Equation (4), as well as in the system Equation (33), so that some more complex robust controllers have to be used to overcome the effects caused by the un-modeled errors for guaranteeing the stability of the closed-loop system Equation (4). On the contrary, the adaptive control law Equation (10) has considered the un-modeled error and updated the virtual reference model $\chi(\Theta, \dot{\Theta}, \ddot{\Theta}_r, \ddot{\Theta}_r)\hat{\rho}$ in the controller Equation (10) online by using the parameter estimator Equation (11). This makes the virtual velocity vectors s and the parameters errors $\bar{\rho}$ be bounded, and finally, the virtual velocity vectors $s \rightarrow 0$ as $t \rightarrow \infty$. On the surface $s = \dot{\Theta} + \Lambda \bar{\Theta} = 0$, the stability of the state $(\bar{\Theta}, \dot{\bar{\Theta}})$ of the closed-loop system is ensured by the Hurwitz matrix $-\Lambda$. Thus, the two problems mentioned in Remark 2 can be resolved or relaxed by using the MRAC based time-varying impedance controller.

REMARK 6. It is also worth noting that, in general, the external loads τ_e cannot be estimated by using the linearly parameterized form Equation (9). Thus, for some accurate force tracking control tasks, the linearly parameterized form Equation (9) should be changed as

$$M(\Theta)\ddot{\Theta}_r + C(\Theta, \dot{\Theta})\dot{\Theta}_r + N(\Theta) = \chi(\Theta, \dot{\Theta}, \ddot{\Theta}_r)\bar{\rho} \quad (34)$$

then under the control law Equation (10) and the parameter estimator Equation (11), the closed-loop system can be given as

$$M\ddot{\Theta} + D(t)\dot{\Theta} + K(t)\Theta = \chi\bar{\rho} + \tau_e \quad (35)$$

where the control parameter selection Equation (27) is considered. Since the right side of Equation (35) is bounded under control law Equation (10) with the parameter estimator Equation (11), the origin $(\bar{\Theta}, \dot{\bar{\Theta}}) = (0, 0)$ of the system Equation (35) is still globally uniformly stable in the Lyapunov's sense. However, the error term $\chi\bar{\rho}$ on the right side of Equation (35) will cause certain force tracking errors. Thus, for accurate force tracking control tasks, measurement of external loads is required, and the MRAC based control law should be changed as

$$\begin{cases} \tau_a = \chi(\Theta, \dot{\Theta}, \ddot{\Theta}_r)\hat{\rho} - K_D s - \tau_e \\ \dot{\hat{\rho}} = -\Gamma^{-1}\chi^T s \end{cases} \quad (36)$$

then the closed-loop system Equation (35) will changed to that same as Equation (31).

4. NUMERICAL SIMULATIONS

To test the feasibility of the proposed adaptive time-varying impedance controller, a model-uncertain planar 2R manipulator is adopted as the plant. Suppose the mass of two links are m_1 and m_2 , respectively, the inertia of two links is I_1 and I_2 , respectively, the length of two links is L_1 and L_2 , respectively, the distance between the mass center of links and joint axes are L_{c1} and L_{c2} , respectively, the dynamic equation of planar 2R manipulator can be given as

$$\begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \quad (37)$$

where θ_1 and θ_2 are the joint angles of the two links, $m_{11} = \rho_1 + 2\rho_3 \cos \theta_2$, $m_{12} = \rho_2 + \rho_3 \cos \theta_2$, $m_{21} = m_{12}$, $m_{22} = \rho_2$, $c_{11} = -\rho_3 \sin \theta_2 \dot{\theta}_2$, $c_{12} = -\rho_3 \sin \theta_2 (\dot{\theta}_1 + \dot{\theta}_2)$, $c_{21} = \rho_3 \sin \theta_2 \dot{\theta}_1$, and $c_{22} = 0$ with $\rho_1 = I_1 + m_1 L_{c1}^2 + I_2 + m_2 (L_1^2 + L_{c2}^2)$, $\rho_2 = I_2 + m_2 L_{c2}^2$, and $\rho_3 = m_2 L_1 L_{c2}$. For the planar 2R manipulator, the linearly parameterized form Equation (34) can be expressed as

$$\chi \bar{\rho} = \begin{bmatrix} \chi_{11} & \chi_{12} & \chi_{13} \\ \chi_{21} & \chi_{22} & \chi_{23} \end{bmatrix} \begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \end{bmatrix} \quad (38)$$

TABLE 1 | Physical parameters of the planar 2R manipulator.

Parameter Symbols	Initial value used in $\hat{\rho}$	Actual value of the plant	Physical Units
m_1	0	2.0	kg
m_2	0	2.0	kg
L_1	0	0.5	m
L_2	0	0.6	m
L_{c1}	0	0.3	m
L_{c2}	0	0.4	m
$I_1 = m_1 L_{c1}^2$	0	0.18	Kg · m ²
$I_2 = m_2 L_{c2}^2$	0	0.32	Kg · m ²

TABLE 2 | Control parameters of the adaptive controller.

Parameters	Symbols	Values	Physical Units
Coefficient	γ	0.04	/
Inertial matrix	M	Given by (37)	Kg · m ²
Coefficient matrix	Λ	γM^{-1}	(Kg · m ²) ⁻¹
Coefficient matrix	Γ	80I	/
Desired stiffness matrix	$K(t)$	$\begin{bmatrix} 5 + 4 \sin(\pi t) & 0 \\ 0 & 5 - 4 \cos(\pi t) \end{bmatrix}$	Nm/rad
Coefficient matrix	K_D	$\frac{1}{\gamma} K(t) M - C$	/
Desired damping matrix	$D(t)$	$\gamma I + \frac{1}{\gamma} K(t) M$	Nm/rad/s

where $\chi_{11} = \ddot{\theta}_{1r}$, $\chi_{12} = \ddot{\theta}_{2r}$, $\chi_{13} = (2\ddot{\theta}_{1r} + \ddot{\theta}_{2r}) \cos \theta_2 - (\dot{\theta}_2 \dot{\theta}_{1r} + \dot{\theta}_1 \dot{\theta}_{2r} + \ddot{\theta}_2 \dot{\theta}_{2r}) \sin \theta_2$, $\chi_{21} = 0$, $\chi_{22} = \ddot{\theta}_{1r} + \ddot{\theta}_{2r}$, $\chi_{23} = \ddot{\theta}_{1r} \cos \theta_2 + \dot{\theta}_1 \dot{\theta}_{1r} \sin \theta_2$. In the simulation, the control task is described as

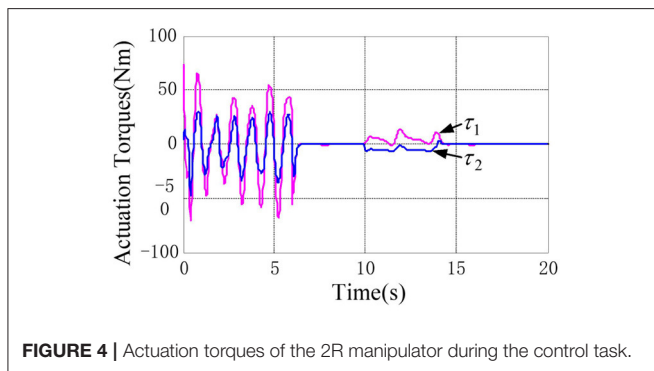
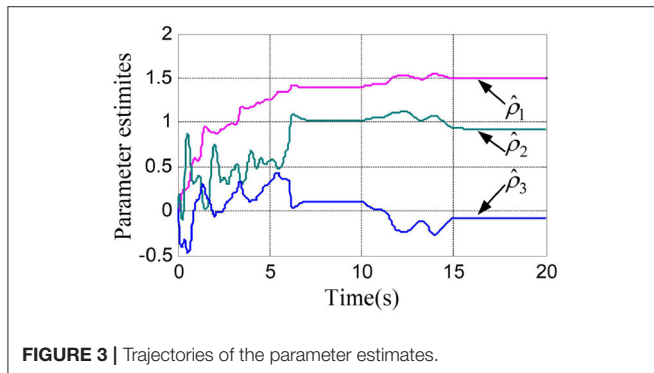
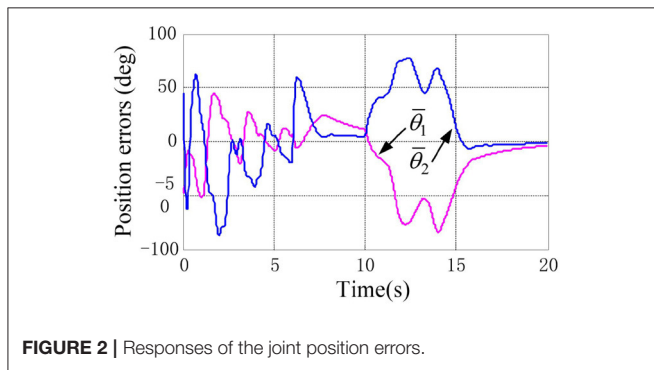
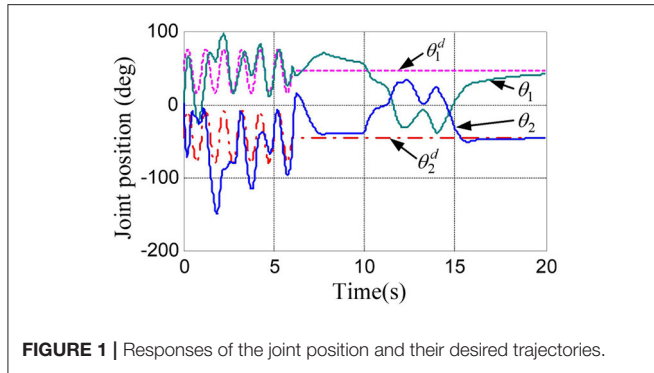
$$\begin{cases} \theta_1^d(t) = \frac{\pi}{4} + \frac{\pi}{6} \sin(2\pi t) & t \leq 6s \\ \theta_2^d(t) = -\frac{\pi}{4} + \frac{\pi}{5} \sin(2\pi t) & t \leq 6s \\ \theta_1^d(t) = \frac{\pi}{4}, \theta_2^d(t) = -\frac{\pi}{4} & t > 6s \end{cases} \quad (39)$$

and

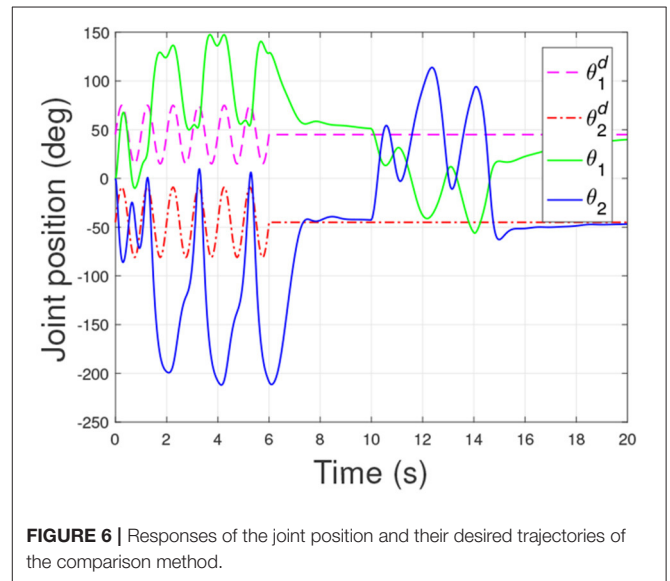
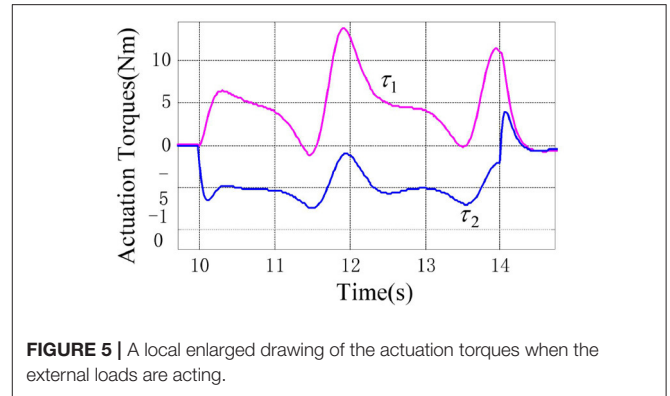
$$\begin{cases} \tau_e = [0 \ 0]^T & t \leq 10s \\ \tau_e = [5 \ -5]^T & 10s < t \leq 14s \\ \tau_e = [0 \ 0]^T & t > 14s \end{cases} \quad (40)$$

The physical parameters of the manipulator are shown in **Table 1**, and the control parameters are shown in **Table 2**, then the response results of the closed-loop system Equation (35) are plotted in the **Figures 1–5**.

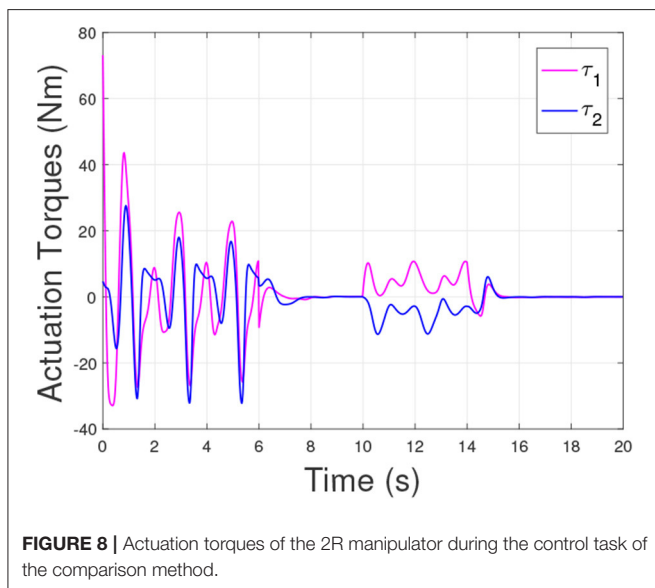
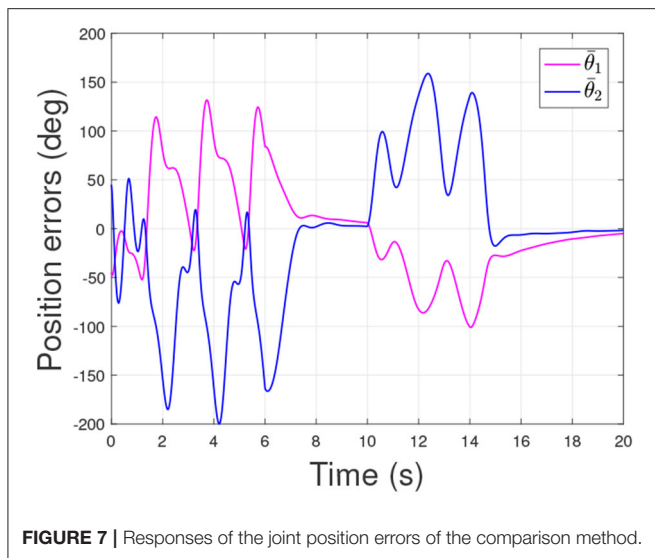
According to the numerical simulation results, even though the physical parameters of the plant are supposed to be zero at the initial moment (see **Figure 3** and **Table 1**), it shows that system Equation (35) is uniformly stable for the controlled planar 2R manipulator. From **Figures 1, 2**, it can be seen that the joint trajectory tracking errors are bounded and converge to zero



when the trajectory tracking task is switched to a stabilization task after the time is larger than 6s. Meanwhile, from **Figure 3**, it is observed that the parameter estimates $\hat{\rho}$ are changed to constant values, and from **Figure 4**, one sees that the actuation



torques converge to zero after the desired joint trajectories $\theta_i^d(t)$ are constants. When the time is falling in the interval $t \in (10, 14](s)$, there are non-zero external loads $\tau_e = [5 \ -5]^T$ acting on the joints, and then the joint angles demonstrate large deviations (as shown in **Figure 2**) due to the small given closed-loop stiffness $K(t)$ (as shown in **Table 2**). Since the desired joint stiffness $K(t)$ is time-varying, the joint position deviations are varying even though the external loads τ_e are constant. **Figure 5** shows a local enlarged drawing of the actuation torques during $\tau_e \neq 0$. It is observed that the average values of the actuation torques happen to be $\tau_a \approx [5 \ -5]^T$, since the planar 2R manipulator moves in the horizontal plane [see Equation (37) where the gravity of the manipulator is not considered here], then the actuation torques τ_a should balance the external loads τ_e . However, due to the desired time-varying stiffness $K(t)$, the parameter estimates $\hat{\rho}$ show certain fluctuations (such that $\hat{\rho} \neq 0$), then the error term $\chi \hat{\rho}$ shown in Equation (35) causes the actuation torques τ_a to show certain fluctuations. The selection of controller parameters γ and Γ affect the performance of the system. We make a performance analysis of the closed-loop control system with different parameters γ and Γ . We found



that with other conditions unchanged, when Γ increases within a certain range, the root-mean-square error (RMSE) of the joint position will increase, and the peak value of the error Θ will also increase, while the RMSE of control torque will decrease. When γ is too large or too small, the performance of the control system will deteriorate. Therefore, the state-independent property allows us to tune the controller parameters offline in advance through simulation, which lays a good foundation for ensuring the performance of the robot.

In order to verify the effectiveness of the controller proposed in this article, we also compared it with the controller in He et al. (2020). Under the same initial conditions and parameters as the proposed controller, the simulation of the comparison controller is carried out, and the response results of the closed-loop system under the comparison controller are shown in **Figures 6–8**. Comparing **Figures 2, 7**, we can get that the RMSE

of the joint position under the proposed controller in **Figure 2** is 0.416 and 0.494, while the RMSE of the joint position under the comparison controller in **Figure 7** is 0.865 and 1.337. Then, it can be concluded that the controller proposed in this article can better realize the trajectory tracking control with higher accuracy. From the simulation results, we can also get that the proposed controller has a smaller peak error. All these simulation results verify the effectiveness of the controller proposed in this article.

5. CONCLUSION

Under the design frame of an MRAC based control system, a time-varying impedance controller is proposed for manipulators with uncertain structure parameters. We show that the proposed controller does not need to use acceleration-based feedback or measurement of the external loads, and the adaptive controller can tolerate considerable structure parameter errors. By employing a Lyapunov-like stability analysis approach, the globally uniform stability of the time-varying closed-loop system is analyzed, and a simple controller parameters selection approach is presented. Through a planar 2R manipulator, the feasibility of the proposed control method is verified by some numerical simulations. Our future work will focus on the anti-interference ability of the proposed controller.

DATA AVAILABILITY STATEMENT

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

AUTHOR CONTRIBUTIONS

XL: conceptualization, visualization, and supervision. GH: methodology. TS: software. SL and QZ: validation. CH: formal analysis. LZ: investigation. JY: resources. ZZ: writing-original draft preparation. TS and GH: writing-review and editing. JZ: project administration. XL and TS: funding acquisition. All authors have read and agreed to the published version of the manuscript. All authors agree to be accountable for the content of the work.

FUNDING

This work is supported by the National Key R&D Program of China under Grants 2019YFB1309603 and 2020AAA0105801, the Natural Science Foundation of Beijing under Grants L202020 and 4204097, the Natural Science Foundation of China under Grant 62003005, 62103007 and 51775002, Beijing Municipal Education Commission under Grant KM202110009009 and KZ202010009015, China Postdoctoral Science Foundation under Grant 2021M693404, Fundamental Research Funds for Beijing Municipal Universities, Yuyou Talent Support Project of North China University of Technology, and open research fund of the State Key Laboratory for Management and Control of Complex Systems under Grant 20210103.

REFERENCES

- Ansari, Y., Manti, M., Falotico, E., Cianchetti, M., and Laschi, C. (2018). Multiobjective optimization for stiffness and position control in a soft robot arm module. *IEEE Robot. Autom. Lett.* 3, 108–115. doi: 10.1109/LRA.2017.2734247
- Burgner-Kahrs, J., Rucker, D. C., and Choset, H. (2015). Continuum robots for medical applications: a survey. *IEEE Trans. Robot.* 31, 1261–1280. doi: 10.1109/TRO.2015.2489500
- Deng, S., Cai, Q., Zhang, Z., and Wu, X. (2021a). User behavior analysis based on stacked autoencoder and clustering in complex power grid environment. *IEEE Trans. Intell. Transp. Syst.* 1–15. doi: 10.1109/TITS.2021.3076607
- Deng, S., Chen, F., Dong, X., and Gao, G. (2021b). Short-term load forecasting by using improved GEP and abnormal load recognition. *ACM Trans. Internet Technol.* 21, 95. doi: 10.1145/3447513
- Dong, X., Axinte, D., Palmer, D., Cobos, S., Raffles, M., Rabani, A., et al., (2017). Development of a slender continuum robotic system for on-wing inspection/repair of gas turbine engines. *Robot. Comput. Integrat. Manuf.* 44, 218–229. doi: 10.1016/j.rcim.2016.09.004
- Ferraguti, F., Secchi, C., and Fantuzzi, C. (2013). A tank-based approach to impedance control with variable stiffness. in *Proceedings of IEEE International Conference on Robotics and Automation* (Karlsruhe), 4948–4953.
- Hamedani, M. H., Zekri, M., and Sheikholeslam, F. (2019). Adaptive impedance control of uncertain robot manipulators with saturation effect based on dynamic surface technique and self-recurrent wavelet neural networks. *Robotica* 37, 161–188. doi: 10.1017/S0263574718000930
- He, G. P., Fan, Y., Su, T. T., Zhao, L., and Zhao, Q. (2020). Variable impedance control of cable actuated continuum manipulators. *Int. J. Control Autom. Syst.* 18, 1839–1852. doi: 10.1007/s12555-019-0449-y
- Hussain, S., Xie, S. Q., and Jamwal, P. K. (2013). Adaptive impedance control of a robotic orthosis for gait rehabilitation. *IEEE Trans. Cybern.* 43, no. 3, 1025–1034. doi: 10.1109/TSMCB.2012.2222374
- Jamwal, P. K., Hussain, S., Ghayesh, M. H., and Rogozina, S. V. (2017). Adaptive impedance control of parallel ankle rehabilitation robot. *J. Dyn. Syst. Meas. Control* 139, 1–7. doi: 10.1115/1.4036560
- Jutinico, A., Jaimes, J., Escalante, F., Perez-Ibarra, J., Terra, M., and Siqueira, A. (2017). Impedance control for robotic rehabilitation: a robust markovian approach. *Front. Neurobot.* 11, 1–16. doi: 10.3389/fnbot.2017.00043
- Kronander, K., and Billard, A. (2016). Stability considerations for variable impedance control. *IEEE Trans. Robot.* 32, 1298–1305. doi: 10.1109/TRO.2016.2593492
- Li, Y. (2021). Command filter adaptive asymptotic tracking of uncertain nonlinear systems with time-varying parameters and disturbances. *IEEE Trans. Autom. Control* 1–8. doi: 10.1109/TAC.2021.3089626
- Li, Y., Liu, Y., and Tong, S. (2021). Observer-based neuro-adaptive optimized control of strict-feedback nonlinear systems with state constraints. *IEEE Trans. Neural Netw. Learn. Syst.* 1–15. doi: 10.1109/TNNLS.2021.3051030
- Li, Z., Huang, Z., He, W., and Su, C. Y. (2017). Adaptive impedance control for an upper limb robotic exoskeleton using biological signals. *IEEE Trans. Ind. Electron.* 64, 1664–1674. doi: 10.1109/TIE.2016.2538741
- Liang, H., Guo, X., Pan, Y., and Huang, T. (2021a). Event-triggered fuzzy bipartite tracking control for network systems based on distributed reduced-order observers. *IEEE Trans. Fuzzy Syst.* 29, 1601–1614. doi: 10.1109/TFUZZ.2020.2982618
- Liang, H., Liu, G., Zhang, H., and Huang, T. (2021b). Neural-network-based event-triggered adaptive control of nonaffine nonlinear multiagent systems with dynamic uncertainties. *IEEE Trans. Neural Netw. Learn. Syst.* 32, 2239–2250. doi: 10.1109/TNNLS.2020.3003950
- Liu, L., Cui, Y., Liu, Y. J., and Tong, S. (2021a). Observer-based adaptive neural output feedback constraint controller design for switched systems under average dwell time. *IEEE Trans. Circuits Syst. I Regul. Papers* 68, 3901–3912. doi: 10.1109/TCSL.2021.3093326
- Liu, L., Liu, Y. J., Chen, A., Tong, S., and Chen, C. L. P. (2020). Integral Barrier Lyapunov function-based adaptive control for switched nonlinear systems. *Sci. China Inf. Sci.* 63, 132203. doi: 10.1007/s11432-019-2714-7
- Liu, Q., Liu, A., Meng, W., Ai, Q., and Xie, S. (2017). Hierarchical compliance control of a soft ankle rehabilitation robot actuated by pneumatic muscles. *Front. Neurobot.* 11, 1–19. doi: 10.3389/fnbot.2017.00064
- Liu, Y. J., Zhao, W., Liu, L., Li, D., Tong, S., and Chen, C. L. P. (2021b). Adaptive neural network control for a class of nonlinear systems with function constraints on states. *IEEE Trans. Neural Netw. Learn. Syst.* 1–10. doi: 10.1109/TNNLS.2021.3107600
- McMahan, W., Chitrakaran, V., Csencsits, M., Dawson, D., Walker, I. D., Jones, B. A., et al. (2006). “Field trials and testing of the OctArm continuum manipulator,” in *Proceedings of IEEE International Conference on Robotics and Automation* (Orlando, FL), 2336–2341.
- Murray, R., Li, Z. X., and Sastry, S. (1994). *A Mathematical Introduction To Robotic Manipulation*. Boca Raton, FL: CRC Press.
- Pena, G. G., Consoni, L. J., dos Santos, W. M., and Siqueira, A. A. G. (2019). Feasibility of an optimal EMG-driven adaptive impedance control applied to an active knee orthosis. *Robot. Auton. Syst.* 112, 98–108. doi: 10.1016/j.robot.2018.11.011
- Perez-Ibarra, J. C., Siqueira, A. A. G., Silva-Couto, M. A., de Russo, T. L., and Krebs, H. I. (2019). Adaptive impedance control applied to robot-aided neuro-rehabilitation of the ankle. *IEEE Robot. Autom. Lett.* 4, no. 2, 185–192. doi: 10.1109/LRA.2018.2885165
- Sharifi, M., Behzadipour, S., and Salarieh, H. (2016). Nonlinear bilateral adaptive impedance control with applications in telesurgery and telerehabilitation. *J. Dyn. Syst. Meas. Control* 138, 111010. doi: 10.1115/1.4033775
- Slotine, J., and Li, W. (1991). *Applied Nonlinear Control*. Upper Saddle River, NJ: Prentice Hall.
- Su, T., Cheng, L., Wang, Y., Liang, X., Zheng, J., and Zhang, H. (2018). Time-optimal trajectory planning for delta robot based on quintic pythagorean-hodograph curves. *IEEE Access* 6, 28530–28539. doi: 10.1109/ACCESS.2018.2831663
- Su, T., Niu, L., He, G., Liang, X., Zhao, L., and Zhao, Q. (2020). Coordinated variable impedance control for multi-segment cable-driven continuum manipulators. *Mech. Mach. Theory* 153, 1–19. doi: 10.1016/j.mechmachtheory.2020.103969
- Tong, S., Min, X., and Li, Y. (2020). Observer-based adaptive fuzzy tracking control for strict-feedback nonlinear systems with unknown control gain functions. *IEEE Trans. Cybern.* 50, 3903–3913. doi: 10.1109/TCYB.2020.2977175
- Wu, D., He, Y., Luo, X., and Zhou, M. (2021a). A latent factor analysis-based approach to online sparse streaming feature selection. *IEEE Trans. Syst. Man Cybern. Syst.* 1–15. doi: 10.1109/TSMC.2021.3096065
- Wu, D., Luo, X., Shang, M., He, Y., Wang, G., and Dong, X. (2020a). A data-characteristic-aware latent factor model for web service QoS prediction. *IEEE Trans. Knowl. Data Eng.* 1–12. doi: 10.1109/TKDE.2020.3014302
- Wu, D., Luo, X., Shang, M., He, Y., Wang, G., and Zhou, M. (2021b). A deep latent factor model for high-dimensional and sparse matrices in recommender systems. *IEEE Trans. Syst. Man Cybern. Syst.* 51, 4285–4296. doi: 10.1109/TSMC.2019.2931393
- Wu, D., Shang, M., Luo, X., and Wang, Z. (2021c). An L_1 -and- L_2 -norm-oriented latent factor model for recommender systems. *IEEE Trans. Neural Netw. Learn. Syst.* 1–14. doi: 10.1109/TNNLS.2021.3071392
- Wu, E. Q., Hu, D., Deng, P. Y., Tang, Z., Cao, Y., Zhang, W. M., et al. (2020b). Nonparametric bayesian prior inducing deep network for automatic detection of cognitive status. *IEEE Trans. Cybern.* 51, 5483–5496. doi: 10.1109/TCYB.2020.2977267
- Wu, E. Q., Lin, C. T., Zhu, L. M., Tang, Z. R., Jie, Y. W., and Zhou, G. R. (2021d). Fatigue detection of pilots’ brain through brain cognitive map and multilayer latent incremental learning model. *IEEE Trans. Cybern.* 1–13. doi: 10.1109/TCYB.2021.3068300
- Wu, E. Q., Zhou, G. R., Zhu, L. M., Wei, C. F., Ren, H., and Sheng, R. S. F. (2021e). Rotated sphere haar wavelet and deep contractive auto-encoder network with fuzzy Gaussian SVM for pilot’s pupil center detection. *IEEE Trans. Cybern.* 51, 332–345. doi: 10.1109/TCYB.2018.2886012
- Xu, G., Song, A., and Li, H. (2011). Adaptive impedance control for upper-limb rehabilitation robot using evolutionary dynamic recurrent fuzzy neural network. *J. Intell. Robot. Syst.* 62, 501–525. doi: 10.1007/s10846-010-9462-3
- Yang, Y., Vamvoudakis, K. G., Modares, H., Yin, Y., and Wunsch, D. C. (2021). Hamiltonian-driven hybrid adaptive dynamic programming.

- IEEE Trans. Syst. Man Cybern. Syst.* 51, 6423–6434. doi: 10.1109/TSMC.2019.2962103
- Yip, M. C., and Camarillo, D. B. (2016). Model-less hybrid position/force control: a minimalist approach for continuum manipulators in unknown, constrained environments. *IEEE Robot. Autom. Lett.* 1, 844–851. doi: 10.1109/LRA.2016.2526062
- Zhai, J., and Xu, G. (2021). A novel non-singular terminal sliding mode trajectory tracking control for robotic manipulators. *IEEE Trans. Circuits Syst. II Exp. Briefs* 68, 391–395. doi: 10.1109/TCSII.2020.2999937

Conflict of Interest: ZZ was employed by Beijing Aerospace Measurement & Control Technology Co., Ltd.

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

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

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



Generative Adversarial Training for Supervised and Semi-supervised Learning

Xianmin Wang¹, Jing Li^{1,2,3*}, Qi Liu¹, Wenpeng Zhao¹, Zuoyong Li² and Wenhao Wang³

¹ Institute of Artificial Intelligence and Blockchain, Guangzhou University, Guangzhou, China, ² Fujian Provincial Key Laboratory of Information Processing and Intelligent Control, Minjiang University, Fuzhou, China, ³ State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China

OPEN ACCESS

Edited by:

Song Deng,
Nanjing University of Posts and
Telecommunications, China

Reviewed by:

Yi He,
Old Dominion University, United States
Lina Yao,
University of New South
Wales, Australia

*Correspondence:

Jing Li
lijing@gzhu.edu.cn

Received: 21 January 2022

Accepted: 25 February 2022

Published: 24 March 2022

Citation:

Wang X, Li J, Liu Q, Zhao W, Li Z and
Wang W (2022) Generative
Adversarial Training for Supervised
and Semi-supervised Learning.
Front. Neurobot. 16:859610.
doi: 10.3389/fnbot.2022.859610

Neural networks have played critical roles in many research fields. The recently proposed adversarial training (AT) can improve the generalization ability of neural networks by adding intentional perturbations in the training process, but sometimes still fail to generate worst-case perturbations, thus resulting in limited improvement. Instead of designing a specific smoothness function and seeking an approximate solution used in existing AT methods, we propose a new training methodology, named Generative AT (GAT) in this article, for supervised and semi-supervised learning. The key idea of GAT is to formulate the learning task as a minimax game, in which the perturbation generator aims to yield the worst-case perturbations that maximize the deviation of output distribution, while the target classifier is to minimize the impact of this perturbation and prediction error. To solve this minimax optimization problem, a new adversarial loss function is constructed based on the cross-entropy measure. As a result, the smoothness and confidence of the model are both greatly improved. Moreover, we develop a trajectory-preserving-based alternating update strategy to enable the stable training of GAT. Numerous experiments conducted on benchmark datasets clearly demonstrate that the proposed GAT significantly outperforms the state-of-the-art AT methods in terms of supervised and semi-supervised learning tasks, especially when the number of labeled examples is rather small in semi-supervised learning.

Keywords: neural networks, adversarial training, generative AT, worst-case perturbations, smoothness function, trajectory-preserving-based alternating update strategy

1. INTRODUCTION

Neural networks have launched a profound reformation in various fields, such as intelligent driving (Feng et al., 2021), neuro-inspired computing (Zhang et al., 2020; Deng et al., 2021b), smart health (Khan et al., 2021), and human computer interaction (Deng et al., 2021a; Pustejovsky and Krishnaswamy, 2021; Fang et al., 2022). However, in practical classification and regression applications (Wu et al., 2021a), since the number of training examples is finite, the error rate calculated by the training examples may be considerably deviated from the one by test examples. This fact causes the overfitting problem (Wu et al., 2021b), which greatly impacts the generalization performance of neural networks. In order to prevent the neural networks from overfitting, one popular approach is to augment the loss function by introducing a regularization term, which encourages the model to be less dependent on the empirical risk for the finite training examples.

Based on Bayesian theory, this regularization term can be interpreted as a prior distribution reflecting the preconceived notion of the model (Bishop and Nasser, 2006; Wu et al., 2020). Accordingly, the prior distribution of a model is usually assumed to be smooth. That is to say, the outputs of a naturally occurring system tend to be smooth with respect to the spatial or temporal inputs (Wahba, 1990). This assumption indicates that the data points close to each other should be highly likely to infer the same predictions. Unfortunately, recent studies show that most of the neural networks suffer from misclassifying some data points that have only small differences from the correctly classified data points (Goodfellow et al., 2014b; Strauss et al., 2017; Yuan et al., 2019). These misclassified data points are called the adversarial examples, which are crafted by the addition of some imperceptible perturbations to the natural examples in the input space.

To overcome the problem that the neural networks are vulnerable to small but malicious perturbations, adversarial training (AT) is proposed (Goodfellow et al., 2014b; Wang et al., 2019; Cui et al., 2021; Zhang et al., 2022). AT aims to smooth the model outputs by penalizing the deviations caused by the adversarial perturbations. The major challenge of AT is how to accurately estimate such perturbations that alter the output distribution around the input data points. To this end, several perturbation-based methods have been proposed by solving an internal optimization problem at the current status of the model. For instance, random AT (RAT) (Zheng et al., 2016) improves the model smoothness by adding the randomly generated perturbations to the input data. These perturbed data points are encouraged to produce the same prediction given by its corresponding unperturbed versions. Since the perturbations around the input appear in random directions, RAT is referred to as an isotropic smoothing approach. However, it is shown that the isotropic smoothing makes the model particularly sensitive to adversarial examples (Szegedy et al., 2013; Goodfellow et al., 2014b). Based on this consideration, Goodfellow et al. (2014b) proposed a standard AT (SAT). SAT is an anisotropic method that smoothes the output distribution by making the model robust against perturbations in a specific direction. This specific direction in the input space is called the adversarial direction, in which the output of the model is the most sensitive. To identify the perturbations in the adversarial direction, SAT first formulates an objective function based on the differences between the prediction and correct labels and then solves this function with an efficient Frank-Wolfe optimizer. SAT requires the use of labels when calculating the adversarial perturbations. Hence, SAT cannot be applied to the regime of semi-supervised learning. Virtual AT (VAT) (Miyato et al., 2018) extends the notion of SAT in the sense that it defines the adversarial direction without label information, and thus can be applied to both supervised and semi-supervised learning tasks. We observe that in order to generate the adversarial perturbations, the existing AT methods explicitly define a smoothness function to regularize the neural networks. This leads to two limitations. First, it is extremely difficult to find a universal smoothness function due to the various output patterns and distance metrics. Second, there is no analytical solution to such a box-constrained function. Consequently, a numerical method is generally used to seek an

approximate solution, which greatly affects the performance of identifying the worst-case adversarial perturbations.

Different from previous methodologies, we propose a novel AT methodology, named generative AT (GAT) in this article, to improve the smoothness of output distribution of neural networks for the supervised and semi-supervised learning tasks. The objective of the proposed GAT is to train the target classifier such that it not only achieve the minimum prediction error but also has the best robustness against the adversarial perturbations. To this aim, we formalize the regularizing process as a minimax game. To be specific, we exploit the cross entropy method to construct a new *adversarial loss* function. Moreover, we develop an effective alternating update strategy to optimize the challenging non-convex problems. The experimental results tested on benchmark datasets show that the proposed GAT obtains the empirical equilibrium point and state-of-the-art performance.

The main contributions of this article are summarized as follows:

- We formulate the regularizing for the learning task as a minimax game according to the outputs of the target classifier from the natural example and its adversarial version derived by a perturbation generator. As the game approaches the empirical equilibrium, the target classifier achieves the best performance.
- A new *adversarial loss* function is constructed based on the *cross entropy* method, which not only accurately reflects the deviation caused by the perturbation but also efficiently assesses the confidence of network output.
- An effective alternating update strategy based on trajectory preserving is proposed to control the minimax optimization training to be stable.
- The proposed GAT regularizes the model without label information, hence it can be applied to the supervised and semi-supervised learning tasks.

It is worth emphasizing that our method differs from any one of the generative-model-based AT methods (Kingma et al., 2014; Maaløe et al., 2016; Salimans et al., 2016; Dai et al., 2017). This family of methods is considered to be an improvement of Generative Adversarial Network (GAN), in the sense that the target classifier in their frameworks is the extension of the GAN's discriminator serving for distinguishing the natural and generated examples. For our method, the discriminator is not the target classifier; instead, it is manually designed according to the outputs of the target classifier over the natural example and its adversarial version.

2. PROBLEM SETTING AND RELATED WORKS

Without loss of generality, we consider the classification tasks in a semi-supervised setting. Let $x \in \mathcal{X} = R^I$ be the input vector with I -dimension and $y \in \mathcal{Y} = Z^K$ be the one-hot vector of labels with K categories. $\mathcal{D}^l = \{x_{(i)}^l, y_{(i)}^l | i = 1, \dots, N^l\}$ and $\mathcal{D}^{ul} =$

$\{x_{(j)}^{ul} | j = 1, \dots, N^{ul}\}$ denote the labeled and unlabeled dataset, where N^l and N^{ul} are the number of labeled and unlabeled examples. AT regularizes the neural network such that both the natural and perturbed examples output the intended predictions. That is, we aim to learn a mapping $\mathbb{F}: X \rightarrow [0, 1]^K$ parameterized with $\theta \in \Theta$ via solving the following optimization problem

$$\min \left\{ \mathcal{L}_S(D^l, \theta) + \lambda \cdot \mathcal{L}_R(D^l, D^{ul}, \theta) \right\}. \quad (1)$$

The symbol \mathcal{L}_S in Equation 1 represents the *supervised loss* over the labeled dataset, which can be expanded as

$$\mathcal{L}_S = \mathbb{E}_{(x^l, y^l) \sim D^l} \Gamma(y^l, F_\theta(x^l)), \quad (2)$$

where $F_\theta(x^l)$ denotes the output distribution vector of the neural network on the input x^l given the model parameter θ , y^l is the one-hot vector of the true label for x^l . The operator $\Gamma(\cdot, \cdot)$ denotes the distance measure used to evaluate the similarity of two distributions. A common choice of Γ for the supervised cost \mathcal{L}_S is the measure of *cross entropy*. \mathcal{L}_R is the *adversarial loss*, which is served as a regularization term for promoting the smoothness of the model. The *adversarial loss* plays an important role in enhancing the generalization performance while the number of labeled examples is small relative to the number of the whole training examples (i.e., $N^l \ll N^{ul} + N^l$). λ is a non-negative value that controls the relative balance between the *supervised loss* and the *adversarial loss*.

Many approaches are presented to construct \mathcal{L}_R based on the smoothness assumption, which can be generally represented in a framework as

$$\mathcal{L}_R = \mathbb{E}_{x \sim \mathcal{D}} \Gamma(F_\theta(x; \xi), \tilde{F}_{\theta'}(x; \xi')), \quad (3)$$

where x is sampled from the dataset \mathcal{D} which consists of both labeled and unlabeled examples. $\Gamma(F_\theta(x; \xi), \tilde{F}_{\theta'}(x; \xi'))$ is termed as the smoothness function, which is comprised of a teacher model $F_\theta(x; \xi)$ and a student model $\tilde{F}_{\theta'}(x; \xi')$. The teacher model is parameterized with parameter θ and perturbation ξ , while the student model is parameterized with parameter θ' and perturbation ξ' . The goal of \mathcal{L}_R is to improve the model's smoothness by forcing the student model to follow the teacher model. That is to say, the output distributions yielded by \tilde{F} is supported to be consistent with the outputs derived by F . To this end, the teacher model, student model, and similarity measure are required to be carefully crafted for formulating an appropriate smoothness function against the perturbation of the input and the variance of the parameters. Based on the implementations of this smoothness function, some typical AT approaches can be explicitly defined.

Random Adversarial Training: In RAT, random noises are introduced in the student model instead of the teacher model, and the parameters of the student model are shared with the teacher model. Moreover, L_2 distance is used to measure the similarity of the output distributions derived by \tilde{F} and F on the whole training examples. That is, $\theta' = \theta$, $\xi' \sim \mathcal{N}(0, 1)$, $\xi = 0$, and $\mathcal{D} = \mathcal{D}^{ul} \cup \mathcal{D}^l$ for Equation 3.

Adversarial Training With Π -Model: In contrast to RAT, Π -model introduces random noises to both the teacher model and student model, i.e., $\xi', \xi \sim \mathcal{N}(0, 1)$. The reason for this is based on the assumption that predictions yielded by natural example may itself be an outlier, hence it is reasonable to make two noisy predictions learn from each other. In this case, optimizing the smoothness function for Π -model is equivalent to minimizing the prediction variance of the classifier (Luo et al., 2018).

Standard Adversarial Training: Instead of adding random noises to the teacher/student model, the perturbation adopted in SAT is some imperceptible noise that is carefully designed to fool the neural network. The *adversarial loss* \mathcal{L}_R^{sat} of SAT can be written as

$$\begin{aligned} \mathcal{L}_R^{sat} &= \mathbb{E}_{(x^l, y^l) \sim D^l} \text{KL}(y^l || \tilde{F}_\theta(x^l; \xi_{adv})) \\ \text{s.t. } \xi_{adv} &= \arg \max_{\xi: \|\xi\| \leq \varepsilon} \text{KL}(y^l || \tilde{F}_\theta(x^l; \xi)), \end{aligned} \quad (4)$$

where the operator $\text{KL}(\cdot || \cdot)$ denotes the similarity measure of *Kullback-Leibler (K-L) divergence*. ξ_{adv} denotes adversarial perturbation which is added into x^l to make the output distribution of the student model most greatly deviate y^l . ε is a prior constant that controls the perturbation strength. Note that the teacher model, in this case, is degenerated into the one-hot vector of the true label. Generally, we cannot obtain the exact adversarial direction of ξ_{adv} in a closed form. Hence, a linear approximation of this objective function is applied to approximate the adversarial perturbation. For ℓ_∞ norm, the adversarial perturbation ξ_{adv} can be efficiently approximated by using the famous fast gradient sign method (FGSM) (Madry et al., 2017). That is,

$$\xi_{adv} \approx \varepsilon \cdot \text{sign}(\nabla_{x^l} \text{KL}(y^l || \tilde{F}_\theta(x^l; \xi))). \quad (5)$$

Some alternative invariants such as the iterative gradient sign method (IGSM) (Tramèr et al., 2017) and the momentum IGSM (M-IGSM) (Dong et al., 2018) are available to solve the objective function. By adding adversarial perturbations to the student model, SAT obtains better generalization performance than RAT and Π -model. Unfortunately, SAT can only be applied in supervised learning tasks since it has to use the labeled examples to compute the *adversarial loss*.

Virtual Adversarial Training: Different from SAT, the key idea of VAT is to define the *adversarial loss* based on the output distribution inferred on the unlabeled examples. In this regard, the *adversarial loss* \mathcal{L}_R^{vat} of VAT can be written as

$$\begin{aligned} \mathcal{L}_R^{vat} &= \mathbb{E}_{x \sim \mathcal{D}^l \cup \mathcal{D}^{ul}} \text{KL}(F_\theta(x) || \tilde{F}_\theta(x; \xi_{adv})) \\ \text{s.t. } \xi_{adv} &= \arg \max_{\xi: \|\xi\| \leq \varepsilon} \text{KL}(F_\theta(x) || \tilde{F}_\theta(x; \xi)). \end{aligned} \quad (6)$$

To obtain the adversarial perturbation ε_{adv} , Miyato et al. (2018) proposed to approximate the objective function with a second-order Taylor's expansion at $\varepsilon = 0$. That is,

$$\xi_{adv} \approx \arg \max_{\xi: \|\xi\| \leq \varepsilon} \frac{1}{2} \xi^T H(x, \theta) \xi, \quad (7)$$

where H is a Hessian matrix which is defined by $H(x, \theta) = \nabla \nabla_{\xi} \text{KL}(\mathbf{F}_{\theta}(x) \parallel \tilde{\mathbf{F}}_{\theta}(x; \xi))$. This binomial optimization is an eigenvalue problem that can be solved using power iteration algorithm. Since VAT acquires the adversarial perturbation in the absence of label information, this method is applicable to both supervised and semi-supervised learning.

3. THE PROPOSED METHOD

Adversarial training methods regularize the neural network *via* forcing the output distribution to be robust against adversarial examples. To obtain intentional perturbations, the existing AT methods require to explicitly define a smoothness function to compute the perturbations. Due to the non-convex characteristic of the smoothness function, the existing AT methods usually fail to generate worst-case perturbation by approximation analysis. To tackle this problem, we propose a novel AT framework termed GAT for improving the smoothness of the neural network, where the worst-case perturbation of the input is generated by a generator. In the following sections, we construct our framework by answering two central questions: (1) how to formulate the loss function with the perturbation generator and target classifier and (2) how to effectively optimize this loss function during the training process.

3.1. GAT Loss Based on Minimax Game

In our framework, two neural networks are considered, i.e., the target classifier $\mathbf{T}_{\theta}(x)$ parameterized with θ and the perturbation generator $\mathbf{G}_{\varphi}(x)$ parameterized with φ . In our framework, the target classifier is the optimization objective that will be required eventually. The perturbation generator is constructed by an auto-encoder-like neural network. Specifically, the perturbation generator can be defined as a mapping $\mathbf{G}_{\varphi}: \mathcal{X} \rightarrow \mathcal{X}$, which takes a natural example in \mathcal{X} and then transforms it into an imperceptible perturbation in the same space \mathcal{X} . For ℓ_{∞} norm, such constraints can be represented as

$$\forall x, \|\mathbf{G}_{\varphi}(x)\|_{\infty} \leq \varepsilon, \quad (8)$$

where ε is the perturbation bounds that controls the adversarial strength. To implement the constraints indicated by Equation 8, the activation function of the last layer in \mathbf{G}_{φ} is particularly defined as $\varepsilon \cdot \tanh(\cdot)$. Then, the generated perturbation is added into the corresponding natural example to composite an adversarial example.

The goal of \mathbf{G}_{φ} is to find a perturbation that most deviates the current inferred output of the target classifier from the status quo, while $\mathbf{T}_{\theta}(x)$ is to minimize the prediction error for the natural example as well as the deviation caused by such perturbation. This problem can be formulated as a minimax game and the loss function of which can be formulated as

$$\min_{\theta} \max_{\varphi} \mathbb{E}_{(x^l, y^l) \sim \mathcal{D}^l} \Gamma_S(y^l, \mathbf{T}_{\theta}(x^l)) + \lambda \cdot \mathbb{E}_{x \sim \mathcal{D}^l \cup \mathcal{D}^{ul}} \Gamma_R(\mathbf{T}_{\theta}(x), \mathbf{T}_{\theta}(\mathbf{G}_{\varphi}(x) + x)). \quad (9)$$

Equation 9 is referred to as the GAT loss, which is comprised of a *supervised loss* \mathcal{L}_S and an *adversarial loss* \mathcal{L}_R . \mathcal{L}_S is

determined by labeled examples, while \mathcal{L}_R is independent of the labels and served as a regularization term smoothing the model. The parameter λ controls the balance of \mathcal{L}_S and \mathcal{L}_R . For the maximization and minimization loop of the minimax game, φ and θ are the parameters required to be optimized. Since \mathcal{L}_R is defined over the whole data set, our method is applicable to semi-supervised learning. Note that for the *adversarial loss*, the target classifier $\mathbf{T}_{\theta}(x)$ is considered as the teacher model, while the compound function of $\mathbf{T}_{\theta}(\mathbf{G}_{\varphi}(x) + x)$ is served as the student model.

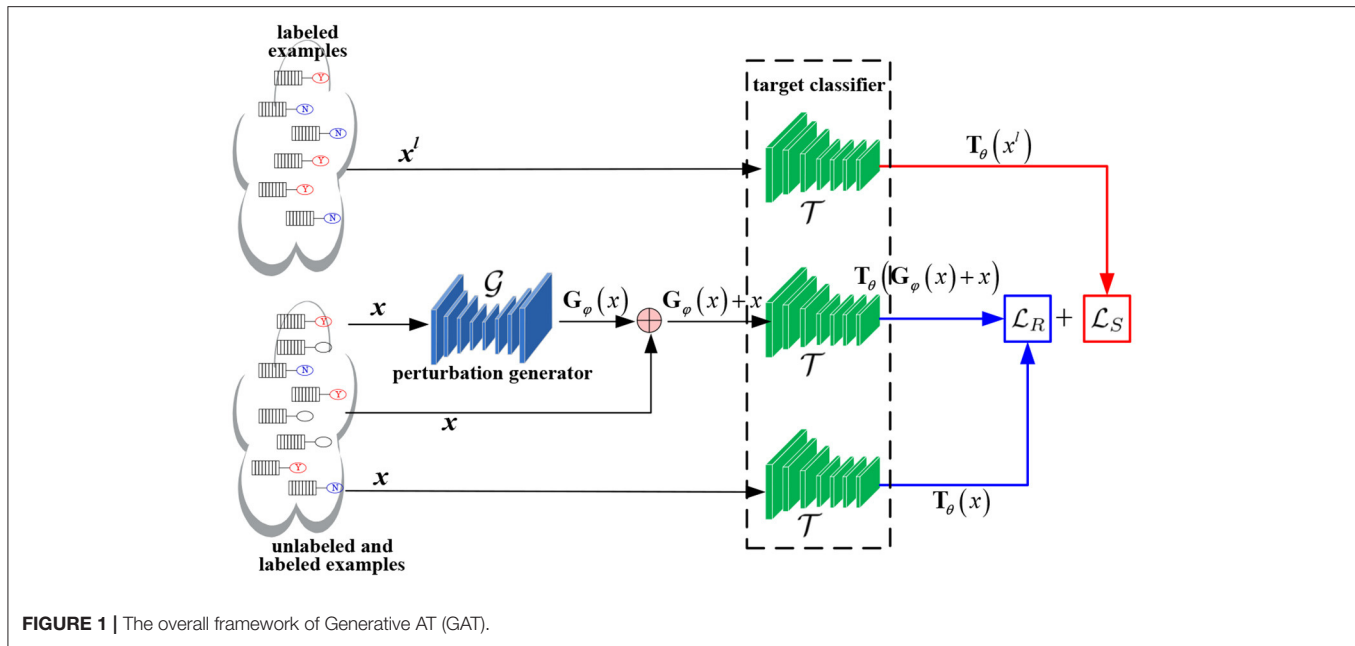
In addition, the operator $\Gamma_S(\cdot, \cdot)$ and $\Gamma_R(\cdot, \cdot)$ are the similarity measures for \mathcal{L}_S and \mathcal{L}_R , respectively. Here, Γ_R is crucial for the construction of *adversarial loss*. Instead of using *K-L divergence* to define the *adversarial loss* as VAT/SAT does, we exploit *cross entropy* measures to formulate the *adversarial loss* function. There are two beneficial effects for this implementation. First, *cross entropy* overcomes the problem of zero avoiding, an inward nature for the *K-L divergence* (Bishop and Nasser, 2006). Second, since *cross entropy* can be represented as the sum of *K-L divergence* and *information entropy*, \mathcal{L}_R not only implies the deviation of the output distributions, but also signifies the confidence of the prediction of the target classifier. In particular, by substituting Γ_R with *cross entropy* in Equation 9, \mathcal{L}_R in GAT loss can be rewritten as

$$\text{CE}(\mathbf{T}_{\theta}(x), \mathbf{T}_{\theta}(\mathbf{G}_{\varphi}(x) + x)) = \text{KL}(\mathbf{T}_{\theta}(x) \parallel \mathbf{T}_{\theta}(\mathbf{G}_{\varphi}(x) + x)) + H(\mathbf{T}_{\theta}(x)), \quad (10)$$

where the operator $\text{CE}(\cdot, \cdot)$ and $H(\cdot)$ denote *cross entropy* and *information entropy*. In Equation 10, $\text{KL}(\mathbf{T}_{\theta}(x) \parallel \mathbf{T}_{\theta}(\mathbf{G}_{\varphi}(x) + x))$ is termed as smoothness term, which reflects the deviation of the output distributions, while $H(\mathbf{T}_{\theta}(x))$ is termed as confidence term, which indicates the confidence of the output distribution. Moreover, we observed that the confidence term is independent with parameter φ . Hence, for the maximization loop of the minimax game, maximizing \mathcal{L}_R requires to maximize the smoothness term only. Whereas, for the minimization loop, minimizing \mathcal{L}_R requires to minimize both the smoothness term and confidence term. Note that minimizing the confidence term facilitates boosting of the prediction confidence of the neural network. Thus, our *adversarial loss* has the effect of entropy minimization proposed in Grandvalet and Bengio (2004) and Sajjadi et al. (2016).

3.2. Alternating Update Process Based on Trajectory Preserving

Figure 1 depicts the framework of GAT, in which two neural networks are required to be optimized, i.e., the target classifier \mathcal{T} and the perturbation generator \mathcal{G} . \mathcal{G} takes natural example x from the full dataset comprising of both the labeled and unlabeled examples and generates a perturbation $\mathbf{G}_{\varphi}(x)$. Then, $\mathbf{G}_{\varphi}(x)$ is appended into x to composite an adversarial example. Both the adversarial example and its corresponding natural example are fed into \mathcal{T} for constructing the *adversarial loss* \mathcal{L}_R . Meanwhile, labeled example x^l sampled from the labeled dataset is input to \mathcal{T} for formulating the *supervised loss* \mathcal{L}_S .



The objective of our framework is to find stable θ and φ such that \mathcal{G} maximizes the GAT loss for the given fixed θ , while \mathcal{T} minimizes the GAT loss for the given fixed φ . Due to the non-linear constraint of the perturbation and non-convex properties of the loss function, this optimization problem is very challenging. Inspired by the training pattern of GAN (Goodfellow et al., 2014a) and some common tricks in reinforcement learning (Mnih et al., 2015), we propose to optimize the GAT loss by an alternative updating procedure and stabilize this procedure based on trajectory preserving.

First, we decompose the minimax optimization problem into the inner loop and outer loop. The inner loop aims to derive an optimal φ for maximizing the loss, while the outer loop aims to obtain an optimal θ for minimizing the loss. Due to the fact that the parameter φ in the inner loop is independent of the supervised loss during the maximizing procedure, then the optimal φ of \mathcal{G} under the fixed θ can be written as Equation 11. Meanwhile, the optimal θ of \mathcal{T} under the given fixed φ can be represented as Equation 12.

$$\varphi = \arg \max_{\varphi} \mathbb{E}_{x \sim \mathcal{D}^l \cup \mathcal{D}^{ul}} \text{CE}(\mathbf{T}_{\theta}(x), \mathbf{T}_{\theta}(x + \mathbf{G}_{\varphi}(x))), \quad (11)$$

$$\theta = \arg \min_{\theta} \mathbb{E}_{(x^l, y^l) \sim \mathcal{D}^l} \text{CE}(y^l, \mathbf{T}_{\theta}(x^l)) + \lambda \cdot \mathbb{E}_{x \sim \mathcal{D}^l \cup \mathcal{D}^{ul}} \text{CE}(\mathbf{T}_{\theta}(x), \mathbf{T}_{\theta}(x + \mathbf{G}_{\varphi}(x))). \quad (12)$$

Second, since the perturbation generator and the target classifier are assumed to be neural networks, the parameters θ and φ in Equations 11 and 12 can be calculated by stochastic-gradient-based methods (Liu et al., 2021; Jin et al., 2022). A traditional solution to this minimax problem is to alternatively update φ by gradient ascent over the full dataset and update θ by gradient descent over the labeled dataset. However, since the number

Algorithm 1: Trajectory preserving training process.

```

1 Initialize randomly  $\theta$ 
2 for epoch = 1 : E do
3   Create empty list L
4   Initialize randomly  $\varphi_0$ 
5   for t = 0 : T do
6     Sample batch  $\{x_i^{(t)}\}$  of size M from  $\mathcal{D}^{ul} \cup \mathcal{D}^l$ 
7     Store  $(\{x_i^{(t)}\}, \varphi^{(t)})$  into the list L
8     Update  $\varphi^{(t+1)}$  by gradient ascent (Equation 13)
9   end
10  for t = 0 : T do
11    Retrieve  $(\{x_i^{(t)}\}, \varphi^{(t)})$  from the list L
12    Pseudo-update  $\varphi'$  by gradient ascent (Equation 14)
13    Sample batch  $\{(x_j^l, y_j^l)\}$  of size N from  $\mathcal{D}^l$ 
14    Update  $\theta$  by gradient descent (Equation 15)
15  end
16 end
17 return  $\theta$ 

```

of labeled training examples is small, both φ and θ are not easy to converge in practice. We develop a trajectory preserving strategy to tackle this problem. In our method, for each epoch of alternating, we update φ using gradient ascent and record the update trajectories of φ . Then, based on these trajectories, we retrieve the intermediate parameter φ' by executing a pseudo-update procedure for φ . Finally, we update θ by gradient descent under the given φ' .

The implementation details of the proposed trajectory preserving training procedure are illustrated in **Algorithm 1**,

where E is the number of training epochs, T is the maximum iterations in each epochs. Equations 13 and 14 represent the updating and pseudo-updating for φ by gradient ascent. Equation 15 describes the updating process for θ by gradient descent. α_g and α_t are the learning rate for the perturbation generator and target classifier, respectively.

$$\varphi^{(t+1)} = \varphi^{(t)} + \alpha_g \nabla_{\varphi^{(t)}} \frac{1}{M} \sum_{i=1}^M \text{CE}(\mathbf{T}_{\theta}(x_i^{(t)}), \mathbf{T}_{\theta}(x_i^{(t)} + \mathbf{G}_{\varphi^{(t)}}(x_i^{(t)}))) \quad (13)$$

$$\varphi' = \varphi^{(t)} + \alpha_g \nabla_{\varphi^{(t)}} \frac{1}{M} \sum_{i=1}^M \text{CE}(\mathbf{T}_{\theta}(x_i^{(t)}), \mathbf{T}_{\theta}(x_i^{(t)} + \mathbf{G}_{\varphi^{(t)}}(x_i^{(t)}))) \quad (14)$$

$$\theta = \theta - \alpha_t \nabla_{\theta} \left\{ \frac{1}{N} \sum_{j=1}^N \text{CE}(y_j^l, \mathbf{T}_{\theta}(x_j^l)) + \frac{\lambda}{M} \sum_{i=1}^M \text{CE}(\mathbf{T}_{\theta}(x_i^{(t)}), \mathbf{T}_{\theta}(x_i^{(t)} + \mathbf{G}_{\varphi'}(x_i^{(t)}))) \right\}. \quad (15)$$

4. EXPERIMENTS

To validate the performance of our method on supervised and semi-supervised task, we carried out experiments on synthetic datasets and practical benchmarks by comparing with various strong competitors.

4.1. Supervised Learning on a Synthetic Dataset

This section tests the supervised learning performance of our method for binary classification problems using two well-known synthetic datasets, i.e., the “Moons” dataset (termed as M -dataset) and the “Circles” dataset (termed as C -dataset). The data points in the two datasets are sampled uniformly from two trajectories over the space of R^2 and embedded linearly into 100-dimension vector space. Each dataset contains 16 training data points and 1,000 testing points. **Figures 4, 5** provide the visualizations for M -dataset and N -dataset, where the red circles and blue triangles separately stand for the training examples with labels 1 and 0. The target classifier used in this experiment is a

neural network with one hidden layer comprised of 100 hidden units, where ReLU and softmax activation function are applied to the hidden units and output units. We compare our method with some popular AT methods, such as SAT (Goodfellow et al., 2014b), RAT (Zheng et al., 2016), and VAT (Miyato et al., 2018). These AT methods and the proposed GAT are conducted under the setting of $\lambda = 1$ and $\epsilon = 0.2$. Particularly, the perturbation generator in our method has three hidden layers with the unit number 128, 64, and 128, respectively.

Since the number of the training examples is extremely small compared to the input dimension, the target classifier for binary classification is very vulnerable to the problem of overfitting. **Figures 2A,B** depict the transitions of the accuracy rates for the target classifier with the GAT regularization and without this regularization (termed as Plain NN). It can be observed that the training accuracy of Plain NN and GAT achieved 100% for the two datasets. Nevertheless, the test accuracy rate of GAT is noticeably higher than that of Plain NN. Although our method suffers from some fluctuations with the accuracy rate at the initial stage of the training process, the test accuracy rate of our method finally achieves a stable value after a few iterations, thanks to the trajectory preserving training strategy. **Figure 3** visualizes the output distributions of the trained target classifier on the M -dataset and C -dataset with our method and Plain NN. We can observe that compared to plain NN, GAT provides more flat regions for the landscape of the output distribution. This phenomenon indicates that our method is conducive to the smoothness of the model in the sense that flat surfaces of the landscape imply small deviations of the output.

Moreover, we plot the contours of the target classifier's predictions for label 1 on the two synthetic datasets by various regularization methods. As shown in **Figures 4, 5**, the black line in each plot stands for the contour of value 0.5, which is usually used as the decision boundary for the binary classification tasks. From these figures, we can see that the L_2 regularization method fails to acquire correct decision boundary on both the M -dataset and C -dataset, hence, many false predictions are produced by this method. RAT obtains convincing decision boundary for M -dataset, but it generates an unreasonable decision boundary for C -dataset. Among these methods, only SAT, VAT, and

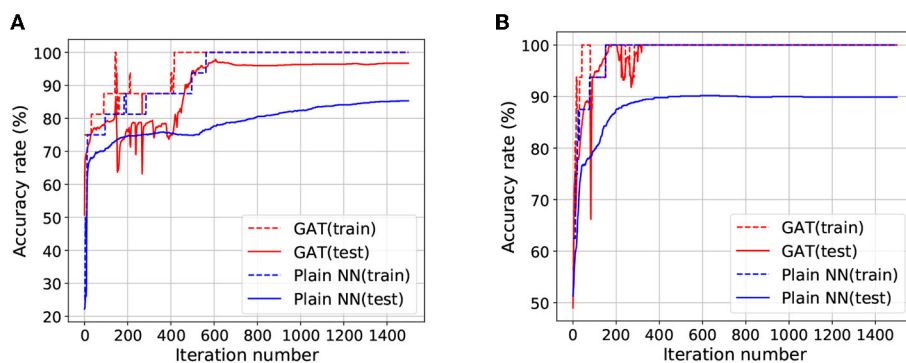
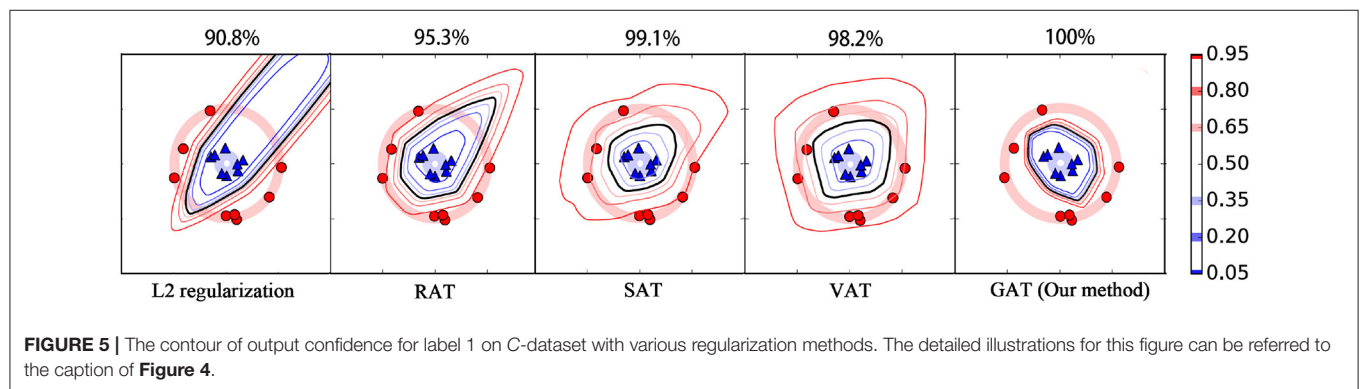
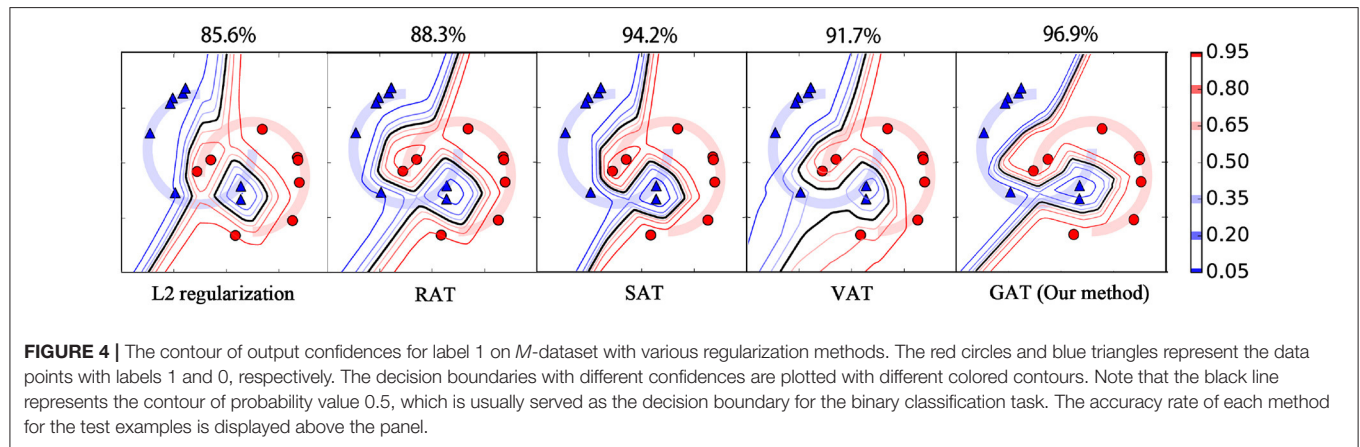
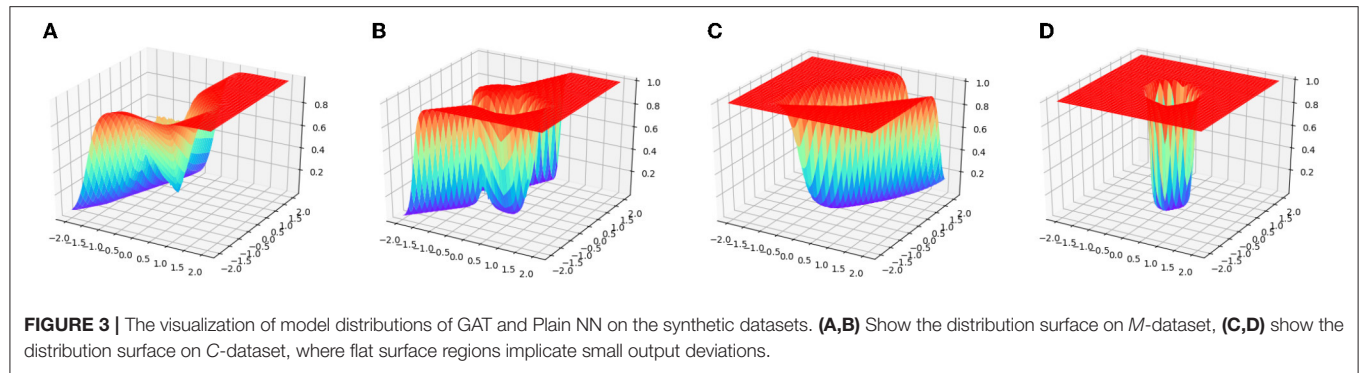


FIGURE 2 | The transition curves of accuracy rates by Plain NN and the proposed GAT on M -dataset and C -dataset. **(A)** Plots the results for M -dataset, **(B)** plots the results for C -dataset.



our method yield applicable decision boundary for both the *M*-dataset and *C*-dataset, because these methods employ an anisotropic way to smooth the classifier. Compared to RAT and VAT, the decision boundaries of our method for different contour values are more compact. This phenomenon illustrates that our method can provide more confidence predictions for the new instances, thanks to the *cross entropy* measure for the adversarial loss. Our method also achieves the highest test accuracy rate against its competitors on both the *M*-dataset and *C*-dataset.

4.2. Supervised Learning on the Benchmark Dataset

In this section, we evaluate the performance of our methods on the MNIST dataset for a supervised learning scenario. The origin 60,000 training examples are split into 50,000 training examples and 10,000 test examples. The target classifier is made up of four hidden dense layers, whose unit numbers are 1200, 600, 300, and 150, respectively. The input dimension of the target classifier is 784 and the output dimension is 10. For each method, we use the

setting of hyper-parameters that exhibits the best performance on the test dataset to train the neural network and record their test errors. The perturbation generator in our method is comprised of hidden layers whose unit numbers are 1200, 600, 300, and 600, respectively. The control parameters of the methods by our implementations are set $\lambda = 1$ and $\epsilon = 0.2$. We compare our method with some typical AT methods on the MNIST dataset for supervised learning task. To verify the capability of the trajectory preserving strategy, we also conducted an ablation experiment for GAT-woTP, a method using the proposed GAT framework but Without Trajectory Preserving strategy during the training. The test error rates of these methods are reported in **Table 1**. The experimental results demonstrate that our method surpasses the previous state-of-the-art AT methods by a large margin. Moreover, our method also outperforms advanced generation-based algorithms such as Ladder network and CatGAN. Besides, note that the error rate obtained by our method is much lower than that acquired by GAT-woTP. This is because the trajectory preserving strategy is benefit to ensure the stability of the training process. Without this strategy, GAT is usually difficult to achieve a favorable convergent point during the training.

4.3. Semi-supervised Learning on Benchmark Dataset

This section validates the effectiveness of our method for semi-supervised learning tasks on three popular benchmarks of MNIST, SVHN, and CIFAR-10. According to the experimental setups in Miyato et al. (2018), we take a test dataset with fixed size 1,000 from the training examples and train the classifier under four sizes of the labeled dataset, i.e., $N_l = \{100, 600, 1000, 3000\}$, where N_l is size of the dataset. The rest instances of the training examples are served as unlabeled examples. Then, we record the test errors under different values of N_l . For our method, we use a mini-batch of size 64 to calculate the *supervised loss*

in Equation 11 and a mini-batch of size 256 to calculate the *adversarial loss* in Equation 12. The control parameters of the methods by our implementations are set at $\lambda = 1$ and $\epsilon = 0.2$. To test the performance of the trajectory preserving strategy for semi-supervised learning, we make several ablation experiments for GAT-woTP which is described in Section 4.2. For the reason that SAT can only be applied to supervised learning task, the results of SAT have not been reported in these experiments.

TABLE 2 | Test error rates of semi-supervised learning methods on MNIST datasets.

Method	Test error rate (%)			
	$N_l = 100$	$N_l = 600$	$N_l = 1,000$	$N_l = 3,000$
SVM	23.44	8.85	7.77	4.21
EmbedNN	16.9	5.97	5.73	3.59
PEA	10.79	2.44	2.23	1.91
Conv-CatGAN [†]	1.93 (± 0.01)	1.86 (± 0.11)	1.73 (± 0.18)	1.67 (± 0.12)
Ladder networks [†]	1.06 (± 0.37)	0.93 (± 0.07)	0.84 (± 0.08)	0.79 (± 0.09)
Auxiliary DGM [†]	0.96 (± 0.02)	0.90 (± 0.05)	0.86 (± 0.13)	0.78 (± 0.05)
RAT	6.62 (± 1.02)	3.75 (± 0.14)	1.61 (± 0.09)	1.51 (± 0.08)
VAT	2.38 (± 0.11)	1.38 (± 0.08)	1.35 (± 0.12)	1.28 (± 0.07)
GAT-woTP	1.97 (± 0.87)	1.66 (± 0.85)	1.58 (± 0.96)	1.32 (± 0.65)
GAT (Our method)	0.90 (± 0.11)	0.85 (± 0.09)	0.83 (± 0.17)	0.75 (± 0.08)

N_l denotes the number of labeled examples for the training dataset.

The results in the upper panel are referred to the reports in prior work, the error rates in the bottom panel are derived by our implementations. [†]Represents the generation-based methods.

TABLE 3 | Test error rates (%) of semi-supervised learning methods on SVHN and CIFAR-10 datasets.

Method	SVHN	CIFAR-10
	$N_l = 1,000$	$N_l = 4,000$
Π -model	5.43 (± 0.25)	16.55 (± 0.29)
Mean teacher	5.21 (± 0.21)	17.74 (± 0.30)
ALI	7.41 (± 0.65)	17.99 (± 1.62)
Ban GAN [†]	4.25 (± 0.03)	14.41 (± 0.30)
Tripple GAN [†]	5.77 (± 0.17)	16.99 (± 0.36)
Improved GAN [†]	4.39 (± 1.20)	16.20 (± 1.60)
TNAR-LGAN (Small) [†]	4.25 (± 0.09)	12.97 (± 0.31)
TNAR-LGAN (Large) [†]	4.03 (± 0.13)	12.76 (± 0.04)
RAT (Small)	8.42 (± 0.22)	18.58 (± 0.26)
RAT (Large)	8.36 (± 0.22)	18.23 (± 0.16)
VAT (Small)	6.83 (± 0.24)	14.87 (± 0.13)
VAT (Large)	5.77 (± 0.32)	14.18 (± 0.38)
GAT-woTP (Small)	6.53 (± 0.95)	14.36 (± 1.03)
GAT-woTP (Large)	5.26 (± 0.92)	14.02 (± 0.88)
GAT (Our method, Small)	4.27 (± 0.14)	12.96 (± 0.15)
GAT (Our method, Large)	4.01 (± 0.11)	12.81 (± 0.13)

N_l represents the number of labeled examples in the training dataset. The results in the upper panel are referred to the reports in prior work, the results in the bottom panel are derived from our implementations. [†]Stands for the generation-based methods.

TABLE 1 | Test error rates of various regularization methods for supervised learning task on MNIST dataset.

Method	Test error rate (%)
SVM (gaussian kernel)	1.40
Dropout	1.05
Maxout networks	0.94
DBM	0.79
Ladder network [†]	0.57
Conv-CatGAN [†]	0.48
Plain NN (Baseline)	1.15
RAT	0.85
SAT (L_∞)	0.78
VAT	0.66
GAT-woTP	0.65
GAT (Our method)	0.45

The upper panel refers to the experimental results reported in prior work, the error rates in the bottom panel are derived by our implementations. [†]Represents the generation-based methods.

For the MNIST dataset, the structures of the target classifier and perturbation generator are identical to the structures employed in Section 4.2. **Table 2** lists the test error rates of the comparing semi-supervised learning methods for different values of N_l on MNIST. The experimental results show that our method achieves the lowest error rates among all the methods for different numbers of labeled examples. Moreover, our method significantly outperforms the state-of-the-art AT methods when the number of labeled examples is small. For the experiments on SVHN and CIFAR-10, two type of convolution neural networks (CNNs), named “Small” (Salimans et al., 2016) and “Large” (Laine and Aila, 2018), are employed as the target classifiers. More details about the settings and structures of the two CNNs can be referred to (Miyato et al., 2018). The structure of the perturbation generator in this experiment is the same as the one applied in the experiment for the MNIST dataset. The performance of various comparing methods for SVHN and CIFAR-10 is reported in **Table 3**. From the table, we can find that GAT obtains the best generalization capability for the SVHN dataset and achieves comparable performance to the state-of-the-art generation-based method such as TNAR-VAE for the CIFAR-10 dataset. In addition, GAT reaches lower error rates compared to GAT-woTP for all the three benchmarks, which verifies the favorable performance of the trajectory preserving strategy for stabilizing the training for our proposal.

5. CONCLUSION

In this article, a novel GAT framework has been proposed to improve the generalization performance of neural networks for both the supervised and semi-supervised learning tasks. In the proposed framework, the target classifier is regularized by letting the perturbation generator watch and move against

the target classifier in a minimax game. We exploit the *cross entropy* to evaluate the output deviation for the regularization term such that the prediction of the target classifier can be reinforced. Furthermore, an effective alternating update method is developed to stably train the target classifier and perturbation generator. Numerous experiments are conducted on synthetic and real datasets and their results demonstrate the effectiveness of our proposal.

DATA AVAILABILITY STATEMENT

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

AUTHOR CONTRIBUTIONS

XW contributed to the conception of the study, performed the data analyses, and wrote the manuscript. JL contributed significantly to analysis and manuscript preparation. QL, WZ, ZL, and WW performed the experiments.

FUNDING

This work was supported by the National Natural Science Foundation of China (Nos. 62072127 and 62002076), Project 6142111180404 supported by CNKLSTISS, Science and Technology Program of Guangzhou, China (Nos. 202002030131 and 201904010493), Guangdong Basic and Applied Basic Research Fund Joint Fund Youth Fund (No. 2019A1515110213), Open Fund Project of Fujian Provincial Key Laboratory of Information Processing and Intelligent Control (Minjiang University) (No. MJUKF-IPIC202101), Natural Science Foundation of Guangdong Province (No. 2020A1515010423).

REFERENCES

- Bishop, C. M., and Nasser, M. N. (2006). *Pattern Recognition and Machine Learning, Vol. 4*. New York, NY: Springer.
- Cui, J., Liu, S., Wang, L., and Jia, J. (2021). “Learnable boundary guided adversarial training,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (Montreal, QC), 15721–15730.
- Dai, Z., Yang, Z., Yang, F., Cohen, W. W., and Salakhutdinov, R. R. (2017). “Good semi-supervised learning that requires a bad gan,” in *Advances in Neural Information Processing Systems*, eds I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Long Beach, CA: Curran Associates), 6510–6520.
- Deng, S., Cai, Q., Zhang, Z., and Wu, X. (2021a). User behavior analysis based on stacked autoencoder and clustering in complex power grid environment. *IEEE Trans. Intell. Transp. Syst.* 1–15. doi: 10.1109/TITS.2021.3076607
- Deng, S., Chen, F., Dong, X., Gao, G., and Wu, X. (2021b). Short-term load forecasting by using improved gep and abnormal load recognition. *ACM Trans. Internet Technol. (TOIT)* 21, 1–28. doi: 10.1145/3447513
- Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., et al. (2018). “Boosting adversarial attacks with momentum,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Salt Lake City, UT), 9185–9193.
- Fang, Y., Chen, P., and Han, T. (2022). Hint: harnessing the wisdom of crowds for handling multi-phase tasks. *Neural Comput. Appl.* 1–23. doi: 10.1007/s00521-021-06825-7
- Feng, S., Yan, X., Sun, H., Feng, Y., and Liu, H. X. (2021). Intelligent driving intelligence test for autonomous vehicles with naturalistic and adversarial environment. *Nat. Commun.* 12, 1–14. doi: 10.1038/s41467-021-21007-8
- Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Bing, X., Warde-Farley, D., Ozair, S., et al. (2014a). “Generative adversarial nets,” in *International Conference on Neural Information Processing Systems* (Montreal, QC).
- Goodfellow, I. J., Shlens, J., and Szegedy, C. (2014b). Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.
- Grandvalet, Y., and Bengio, Y. (2004). “Semi-supervised learning by entropy minimization,” in *International Conference on Neural Information Processing Systems* (Vancouver, BC).
- Jin, L., Wei, L., and Li, S. (2022). Gradient-based differential neural-solution to time-dependent nonlinear optimization. *IEEE Trans. Autom. Control* 1. doi: 10.1109/TAC.2022.3144135
- Khan, M. M., Mehnaz, S., Shaha, A., Nayem, M., and Bourouis, S. (2021). Iot-based smart health monitoring system for covid-19 patients. *Comput. Math. Methods Med.* 2021, 1–11. doi: 10.1155/2021/8591036
- Kingma, D. P., Mohamed, S., Rezende, D. J., and Welling, M. (2014). “Semi-supervised learning with deep generative models,” in *Advances in Neural Information Processing Systems, Vol. 2* (Cambridge, MA: MIT Press), 3581–3589.
- Laine, S. M., and Aila, T. O. (2018). *Temporal Ensembling for Semi-Supervised Learning*. U.S. Patent App. 15/721,433.

- Liu, M., Chen, L., Du, X., Jin, L., and Shang, M. (2021). Activated gradients for deep neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* 1–13. doi: 10.1109/TNNLS.2021.3106044
- Luo, Y., Zhu, J., Li, M., Ren, Y., and Zhang, B. (2018). “Smooth neighbors on teacher graphs for semi-supervised learning,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Salt Lake City, UT), 8896–8905.
- Maaløe, L., Sønderby, C. K., Sønderby, S. K., and Winther, O. (2016). Auxiliary deep generative models. *arXiv preprint arXiv:1602.05473*.
- Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.
- Miyato, T., Maeda, S.-i., Koyama, M., and Ishii, S. (2018). Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE Trans. Pattern Anal. Mach. Intell.* 41, 1979–1993. doi: 10.1109/TPAMI.2018.2858821
- Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., et al. (2015). Human-level control through deep reinforcement learning. *Nature* 518, 529. doi: 10.1038/nature14236
- Pustejovsky, J., and Krishnaswamy, N. (2021). Embodied human computer interaction. *KI-Künstliche Intelligenz* 35, 307–327.
- Sajjadi, M., Javanmardi, M., and Tasdizen, T. (2016). Regularization with stochastic transformations and perturbations for deep semi-supervised learning.
- Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X. (2016). “Improved techniques for training gans,” in *Advances in Neural Information Processing Systems* (Red Hook, NY: Curran Associates), 2234–2242.
- Strauss, T., Hanselmann, M., Junginger, A., and Ulmer, H. (2017). Ensemble methods as a defense to adversarial perturbations against deep neural networks. *arXiv preprint arXiv:1709.03423*.
- Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., et al. (2013). Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.
- Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., and McDaniel, P. (2017). Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*.
- Wahba, G. (1990). Spline models for observational data. *Technometrics* 34, 113–114.
- Wang, X., Li, J., Kuang, X., Tan, Y.-a., and Li, J. (2019). The security of machine learning in an adversarial setting: a survey. *J. Parallel Distrib. Comput.* 130, 12–23. doi: 10.1016/j.jpdc.2019.03.003
- Wu, D., He, Y., Luo, X., and Zhou, M. (2021a). A latent factor analysis-based approach to online sparse streaming feature selection. *IEEE Trans. Syst. Man Cybern. Syst.* 1–15. doi: 10.1109/TSMC.2021.3096065
- Wu, D., Luo, X., Shang, M., He, Y., Wang, G., and Wu, X. (2020). A data-characteristic-aware latent factor model for web services qos prediction. *IEEE Trans. Knowl. Data Eng.* 1. doi: 10.1109/TKDE.2020.3014302
- Wu, D., Shang, M., Luo, X., and Wang, Z. (2021b). An l1-and-l2-norm-oriented latent factor model for recommender systems. *IEEE Trans. Neural Netw. Learn. Syst.* 1–14. doi: 10.1109/TNNLS.2021.3071392
- Yuan, X., He, P., Zhu, Q., and Li, X. (2019). Adversarial examples: Attacks and defenses for deep learning. *IEEE Trans. Neural Netw. Learn. Syst.* 30, 2805–2824. doi: 10.1109/TNNLS.2018.2886017
- Zhang, C., Li, J., Wu, J., Liu, D., Chang, J., and Gao, R. (2022). Deep recommendation with adversarial training. *IEEE Trans. Emerg. Top. Comput.* 1. doi: 10.1109/TETC.2022.3141422
- Zhang, W., Gao, B., Tang, J., Yao, P., Yu, S., Chang, M.-F., Yoo, H.-J., Qian, H., and Wu, H. (2020). Neuro-inspired computing chips. *Nat. Electron.* 3, 371–382. doi: 10.1038/s41928-020-0435-7
- Zheng, S., Song, Y., Leung, T., and Goodfellow, I. (2016). “Improving the robustness of deep neural networks via stability training,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV), 4480–4488.

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

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

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



Research and Application of Fine-Grained Image Classification Based on Small Collar Dataset

Huang Chengcheng^{1,2}, Yuan Jian^{1,3*} and Qin Xiao^{1*}

¹ Guangxi Key Lab of Human-Machine Interaction and Intelligent Decision, Nanning Normal University, Nanning, China,

² Guangxi International Business Vocational College, Nanning, China, ³ Guangxi University for Nationalities, Nanning, China

With the rapid development of apparel e-commerce, the variety of apparel is increasing, and it becomes more and more important to classify the apparel according to its collar design. Traditional image processing methods have been difficult to cope with the increasingly complex image backgrounds. To solve this problem, an EMRes-50 classification algorithm is proposed to solve the problem of garment collar image classification, which is designed based on the ECA-ResNet50 model combined with the MC-Loss loss function method. Applying the improved algorithm to the Coller-6 dataset, and the classification accuracy obtained was 73.6%. To further verify the effectiveness of the algorithm, it was applied to the DeepFashion-6 dataset, and the classification accuracy obtained was 86.09%. The experimental results show that the improved model has higher accuracy than the existing CNN model, and the model has better feature extraction ability, which is helpful to solve the problem of the difficulty of fine-grained collar classification and promote the further development of clothing product image classification.

OPEN ACCESS

Edited by:

Yujie Li,
Fukuoka University, Japan

Reviewed by:

Taige Wang,
University of Cincinnati, United States
Guangwei Gao,
Nanjing University of Posts and
Telecommunications, China

*Correspondence:

Yuan Jian
yuanjian@gxun.edu.cn
Qin Xiao
7670172@qq.com

Received: 28 August 2021

Accepted: 29 November 2021

Published: 11 April 2022

Citation:

Chengcheng H, Jian Y and Xiao Q
(2022) Research and Application of
Fine-Grained Image Classification
Based on Small Collar Dataset.
Front. Comput. Neurosci. 15:766284.
doi: 10.3389/fncom.2021.766284

Keywords: convolutional neural network, collar classification, clothing classification, attention mechanism, loss function

INTRODUCTION

In recent years, due to the emergence of convolutional neural networks, deep learning has been applied more and more widely, including image recognition and natural language processing (Wu et al., 2018; Yuan et al., 2019; Qin et al., 2020; Wu Y. et al., 2020). Wu E. Q. et al. (2019) proposed a Fuzzy Gaussian Support Vector Machine (FGSVM) as a top-level classification tool for deep learning models in order to more accurately classify the pilot's attention state images and analyze the abnormal conditions of the pilot's flight state. Eliminate some Gaussian noise output by the Deep HCAE Network (DHCAEN), which effectively improves the accuracy of image classification. Wu E. Q. et al. (2020) proposed a gamma deep belief network to extract multi-layer depth representation of high-dimensional cognitive data in order to solve the problem of inaccurate identification of pilot fatigue state, and realized automatic reasoning of network structure, with satisfactory results of model accuracy.

In addition, with the advent of the global Internet era, people only need an Internet electronic device to access the Internet and buy products on e-commerce platforms. Therefore, e-commerce is developing rapidly, and recommendation technologies and applications of e-commerce have also attracted the attention of many researchers. For the QoS, Wu D. et al. (2019) propose a

posterior-neighborhood-regularized LF (PLF) model for achieving highly accurate Quality-of-Service (QoS) prediction for web services. Wu D. et al. (2020) proposed a data-characteristic-aware latent factor (DCALF) model to implement highly accurate QoS predictions. For the recommender systems, Wu et al. (2021b) proposed an L1-and-L2-norm-oriented LF (L^3F) model, it has good potential for addressing High-Dimensional and Sparse (HiDS) data from real applications. Wu et al. (2021a) proposed a deep latent factor model (DLFM), it can better describe users' preferences for projects.

Today's popular shopping sites support keyword searches for the style of clothing you want to buy, including keyword searches for clothing collar types. However, product information on websites is often described through a combination of direct image descriptions and key text markups. Text tagging requires a lot of manpower to mark accurately. If images can be directly described, a lot of time and labor costs can be reduced. In the traditional classification and recognition methods of clothing attributes, the amount of feature extraction is huge, and the artificial visual features cannot meet the requirements of real classification, and the efficiency is not high. Therefore, the convolutional neural network in deep learning can be used to efficiently recognize clothing images.

Currently, most researchers focus on apparel category classification or multi-attribute image classification based on apparel. Inoue et al. (2017), in order to solve the multi-label classification problem of fashion images and learn from noisy data unsupervised, provided a new dataset of weakly labeled fashion images of full-body poses Fashion550K with labels containing significant noise and proposed a multi-task label cleaning network to predict the color of clothing and the class of clothing worn by the person in each image. The method generates accurate labels from noisy labels and learns more accurate multi-label classifiers from the generated labels, which effectively solves the multi-label classification problem for fashion images. Liu et al. (2016) collected 800,000 garment images to build a dataset DeepFashion and proposed a deep model of FashionNet based on VGG16, which not only utilizes the attributes and category information of garments but also uses the key point location (landmarks) to assist in extracting features, which can better cope with the deformation of garments. It is an effective way to classify clothing styles and attributes. Nawaz et al. (2018) considered the growing market share of online shopping malls and wide popularity of online sales, collected 1,933 images of five different garments from different online stores and retailers' websites to define the traditional garments of Bangladesh and labeled them accordingly, classified the traditional garments using Google Inception based CNN model and used three different optimizers (SGD, Adam, and RmsProp) to test the constructed models. Among these optimizers, RmsProp performs the best.

Most researchers focus on clothing category classification or clothing multi-attribute image classification. There are very few studies on collar image classification and related datasets are not publicly available. Such image classification is more challenging than ordinary image classification because the differences between classes tend to focus on only a small area.

In this paper, we take advantage of the Efficient Channel Attention (ECA)-ResNet50 network model based on the attention mechanism to continuously focus on the most discriminative regions to achieve image classification and combine Mutual-Channel loss (MC-Loss) to make the original collar image focus on more discriminative regions to improve the model classification effect. The main contributions of this paper are as follows:

- (1) A clothing collar classification image dataset named Collar-6 was established, which contains 6 categories of the round collar, lapel, stand-up collar, hood, V-neck, and fur lapel, with a total of 18,847 images. The dataset has different degrees of noise interference.
- (2) A fine-grained image classification algorithm for a small collar dataset, called ECA MCloss ResNet-50 (EMRes-50), is proposed. experiments are first conducted using the Collar-6 dataset and compared with other popular convolutional neural networks. The experiments do not require any labeled frames and rely only on labels for collar image classification. Second, to verify the effectiveness of the EMRes-50 algorithm, DeepFashion, a public dataset of comparable size to the Collar-6 dataset, is collected for validation. Finally, ablation experiments are performed on EMRes-50. Several experiments have proved that EMRes-50 can effectively solve the problem of collar image classification.

RELATED WORK

ECANet

The visual attention mechanism is unique to visual signal processing in the human brain. After browsing the global image, human vision obtains the visual focus that needs to be focused on, and subsequently devotes more attention resources to this focus region to obtain more detailed information and suppress other useless information, and the Attention Model (AM) (Zhao et al., 2017) of computer vision is generated and has become an important concept in neural networks, which has now been widely used in various types of deep learning tasks such as natural language processing, image recognition and speech recognition (Hu et al., 2018; Woo et al., 2018; Li et al., 2019). The channel attention module assigns different weights to the feature maps, which can be filtered out to help in the classification and attribute prediction of the target. Wang Q. et al. (2020) found by comparing the Squeeze-and-Excitation (SE) module with its three variants Squeeze-and-Excitation Variants 1 (SE-Var1), Squeeze-and-Excitation Variants 2 (SE-Var2), and Squeeze-and-Excitation Variants 3 (SE-Var3) without dimensionality reduction operation that although the design of two fully connected layers in SENet captures the interaction of nonlinear cross-channel information while controlling the complexity of the model, its dimensionality reduction operation is inefficient for capturing the dependencies between all channels. This needs to correspond directly with their weights and avoiding dimensionality reduction is more important than considering the correlation between non-linear channels. Therefore, an efficient channel attention-ECA module for deep convolutional

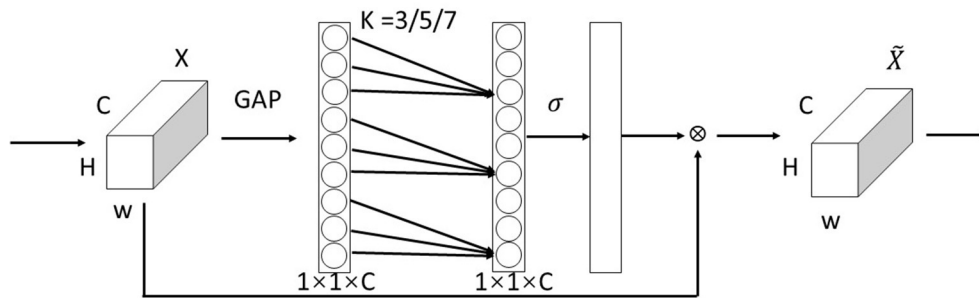


FIGURE 1 | ECA module (Wang Q. et al., 2020).

neural networks is proposed, which avoids dimensionality reduction and captures cross-channel information interactions more effectively, allowing the network to selectively enhance informative features, enabling subsequent processing to make full use of these features, and suppressing useless features to ensure computational performance and model complexity. The ECA module is shown in **Figure 1**.

The ECA module is mainly improved from the SE module (Hu et al., 2018). Using $\omega_{\{k\}}$ to denote the learned channel attention. For the weight $y_{\{i\}}$, ECANet only considers the information exchange between $y_{\{i\}}$ and k neighboring channels, while to further improve the performance, it also allows all channels to share the weight information, as follows:

$$\omega_i = \sigma \left(\sum_{j=1}^k \omega^j y_i^j \right), y_i^j \in \Omega_i^k \quad (1)$$

Where, Ω_i^k represents the set of k adjacent channels of y_i . σ is the activation function. ECANet realizes the information exchange between channels through the one-dimensional convolution with the size of the convolution kernel k :

$$\omega = \sigma(\text{C1D}_k(y)) \quad (2)$$

Where C1D stands for one-dimensional convolution, and the kernel size k represents the coverage of local cross-channel interactions, that is, how many neighbors are involved in the attention prediction of a channel. This method of capturing cross-channel information interactions involves only k parameters, which guarantees performance results and model efficiency. The whole ECA module completes the processing of the attention mechanism in three main steps: First, the global average pooling generates a feature map of $1 \times 1 \times C$ size; Second, the adaptive convolution kernel size k is computed; Third, k is applied in a one-dimensional convolution to obtain the weights of each channel.

The ECA module can be flexibly integrated into existing CNN architectures. ECA-ResNet is an improvement for ResNet networks. **Figure 2** shows the comparison between the original residual block and the residual block with the introduction of the ECA module. The ECA module is placed after the weight layer in the residual block, and the channel attention is paid to the

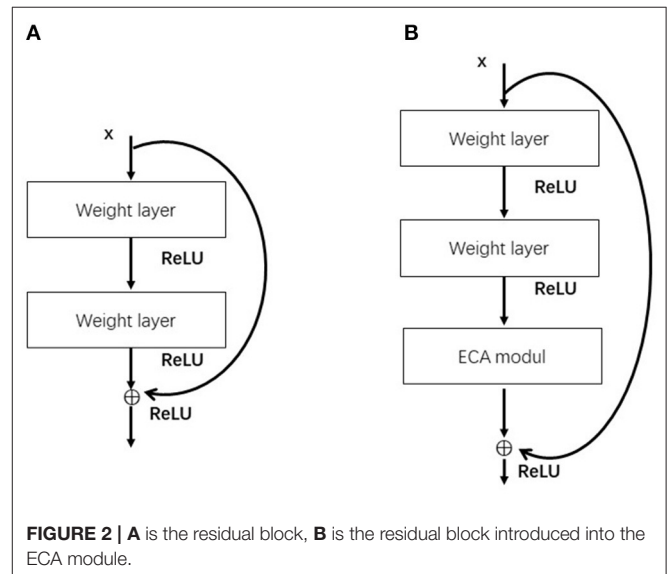


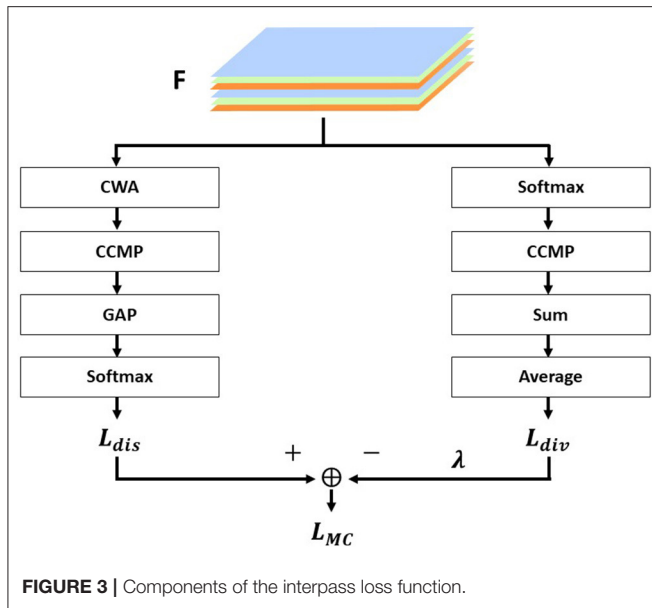
FIGURE 2 | **A** is the residual block, **B** is the residual block introduced into the ECA module.

residual features on the branch before the Addition operation to further increase the feature extraction capability of the network.

Mutual-Channel Loss

Compared with general image classification tasks, the difference and difficulty of fine-grained image classification tasks are that the granularity of the image category is more refined, and the network model is required to find the distinguishable areas between each sub-category to accurately classify the image category. Chang et al. (2020) proposed Mutual-Channel loss (MC-Loss) to group feature channels, each group uses a fixed number of channels to represent a certain class. The Mutual-Channel loss function can be used in combination with any convolutional neural network model. The MC-Loss function takes the output feature channel of the last convolutional layer as input and aggregates it with the cross-entropy loss function through hyperparameters. The loss function of the final model can be expressed as

$$L = L_{CE} + \mu L_{MC} \quad (3)$$



Where, L_{CE} is the traditional cross-entropy loss function, which makes the network extract the global discriminative region of the image. L_{MC} is the multi-channel loss function, which makes the network extract the local discriminative region of the image.

The overall structure of MC-Loss is shown in **Figure 3**, and the two branches are, respectively, represented as L_{dis} and L_{div} . L_{dis} is a discriminability component, and L_{div} is a diversity component. The feature map extracted from the basic network is $F \in \mathbb{R}^{N \times W \times H}$.

The L_{MC} is calculated as follows:

$$L_{MC}(F) = L_{dis}(F) - \lambda \times L_{div}(F) \quad (4)$$

In this framework, the discriminative component L_{dis} is defined as follows:

$$L_{dis}(F) = L_{CE}(y, \underbrace{\frac{[e^{g(F_0)}, e^{g(F_1)}, \dots, e^{g(F_{c-1})}]^T}{\sum_{i=0}^{c-1} e^{g(F_i)}}}_{Softmax}) \quad (5)$$

$$g(F_i) = \underbrace{\frac{1}{WH} \sum_{k=1}^{WH}}_{GAP} \underbrace{j=1, 2, \dots, \xi}_{CCMP} \underbrace{[M_i \cdot F_{ij,k}]}_{CWA} \quad (6)$$

The discriminative component is used to force feature channels to align with class information, and each feature channel corresponding to a particular class should be sufficiently discriminative, which includes four important components. Channel-Wise Attention (CWA), which denotes channel attention, is the process of taking the channel corresponding to each class and discarding it randomly; Cross-Channel Max Pooling (CCMP), which pools all discriminable features of each class into a one-dimensional feature map. Global Average Pooling (GAP), global average pooling, calculates the average

TABLE 1 | EMRes-50 network structure.

EMRes-50 network structure		
	7 × 7 conv 64	
	3 × 3 conv 64	
C:1 × 1	Conv 64] × 3
C:3 × 3	Conv 64	
C:1 × 1	Conv 256	
ECA module	256] × 4
C:1 × 1	Conv 128	
C:3 × 3	Conv 128	
C:1 × 1	Conv 256	
ECA module	256] × 6
C:1 × 1	Conv 256	
C:3 × 3	Conv 256	
C:1 × 1	Conv 1,024	
ECA module	1,024] × 3
C:1 × 1	Conv 512	
C:3 × 3	Conv 512	
C:1 × 1	Conv 2,048	
ECA module	2,048	
C:1 × 1	Conv 2,200	

Average pool, 6d, fc, softmax.

response of each feature channel to obtain a C-dimensional vector where each element corresponds to a separate class. Finally, Softmax, for classification.

Diversity component L_{div} is defined as follows:

$$L_{div}(F) = \frac{1}{c} \sum_{i=0}^{c-1} h(F_i) \quad (7)$$

$$h(F_i) = \sum_{k=1}^{WH} \underbrace{j=1, 2, \dots, \xi}_{CCMP} \underbrace{\left[\frac{e^{F_{ij,k}}}{\sum_{k'=1}^{WH} e^{F_{ij,k'}}} \right]}_{Softmax} \quad (8)$$

Polynomial components are used in order to make the variability between each component of the feature map F greater and to obtain as many diverse features as possible. It consists of four main components: Softmax, which acts as a spatial dimension normalization; CCMP, which is a cross-channel maximum pooling; Sum, which sums all elements on each feature map; and Average, which averages the values of all channels.

EMRES-50

In the clothing images shown on major shopping sites, the collar region accounts for a small proportion of the whole image, and the arbitrary angle of the shot usually makes the collar region appear distorted, missing, and other features. The use of classical convolutional neural networks to process this kind of image data cannot effectively allow deep learning models to focus more on a piece of certain local information. The attention mechanism network can be used to emphasize or select the important information of the target processing

object and suppress some irrelevant detailed information. ResNet (He et al., 2016) effectively solves the degradation problem triggered by increasing depth in deep neural networks due to easy optimization and residual blocks using jump connections, making it easy for the network to learn constant mappings and keep performance without degradation. Therefore, the ResNet network has become a mainstream model in the image field. ECA-ResNet50 is an improvement on ResNet by applying the attention mechanism ECA module to the residual block, which effectively channels the attention learning mechanism makes the network's image feature extraction capability improved. The traditional cross-entropy loss function is the most commonly used loss function in classification, which is used to measure the difference between the distribution learned by the model and the true distribution. Although the cross-entropy function uses an inter-class competition mechanism, which only cares about the accuracy of the prediction probability for the correct label and is good at learning information between classes, it ignores the differences of other non-correct labels, resulting in the learned features being more scattered and only focusing on the global information, which cannot classify and recognize the smaller collar regions in the collar image well. The main idea of fine-grained classification is to identify distinguishable features among subclasses. MC-Loss drills down on the channels to effectively navigate the model, focusing on different distinguishing regions and highlighting diverse features. At the same time, Mutual-Channel Loss does not require any fine-grained qualifying boxes or component annotations and can be combined with cross-entropy loss on commonly used network structures to enhance the network classification ability during the network training phase.

Based on the combination of ECA-ResNet50 and MC-Loss, this paper proposes a fine-grained image classification model—EMRes-50 based on a small collar dataset. EMRes-50 model, due to the addition of MC-Loss, enables the model to continuously focus and distinguish distinguishable regions and discriminable regions from the channel aspect, which can effectively avoid the problem that the features learned by the traditional cross-entropy function are not strongly distinguishable when dealing with fine-grained image classification. So, it can effectively deal

with the fine granularity classification of collar images in complex backgrounds. The architecture of the main feature extraction modules of EMRes-50 is shown in **Table 1**. After each residual block in the original ResNet, the ECA module is added to aggregate multi-scale contextual information from the channels.

The overall network architecture of the training phase of EMRes-50 is shown in **Figure 4**. The interoperability channel loss takes the output feature channels as input and uses hyperparametric support with the cross-entropy loss function assembled together to guide the update of the weights during the training phase, making the model output more diverse features for each class and a more pronounced feature gap between classes.

The weight update process is shown in **Table 2**. The weight update of EMRes-50 is divided into two steps. Through these two steps, the weight update of the entire network is completed:

The first step: In addition to the fully connected layer, other weight layers are combined with the cross-entropy loss and MC-Loss to obtain the weight W_i , and update W_i through the loop network layer N .

Step 2: The last fully connected layer uses the traditional cross-entropy loss to update the weight W_n , and updates W_n through the loop network layer N .

EXPERIMENT

Dataset

This paper constructs a clothing collar type dataset named Collar-6, the images are from Taobao (<https://www.taobao.com/>), Tmall (<https://www.tmall.com/>), clothing brand official websites, and other major e-commerce platforms, through the manual collection, crawler way to collect the images collected by the figure. The images are used for experimental purposes only, not for commercial use.

The Collar-6 dataset contains 6 categories: round collar, lapel collar, stand collar, hooded collar, V collar, and fur lapel, with men's, women's, and children's clothing, with a total of 18,847 images. The collar part in most of the images only occupies a small part of the image, and the rest of the area belongs to the noise which is not related to classification. Therefore, this dataset

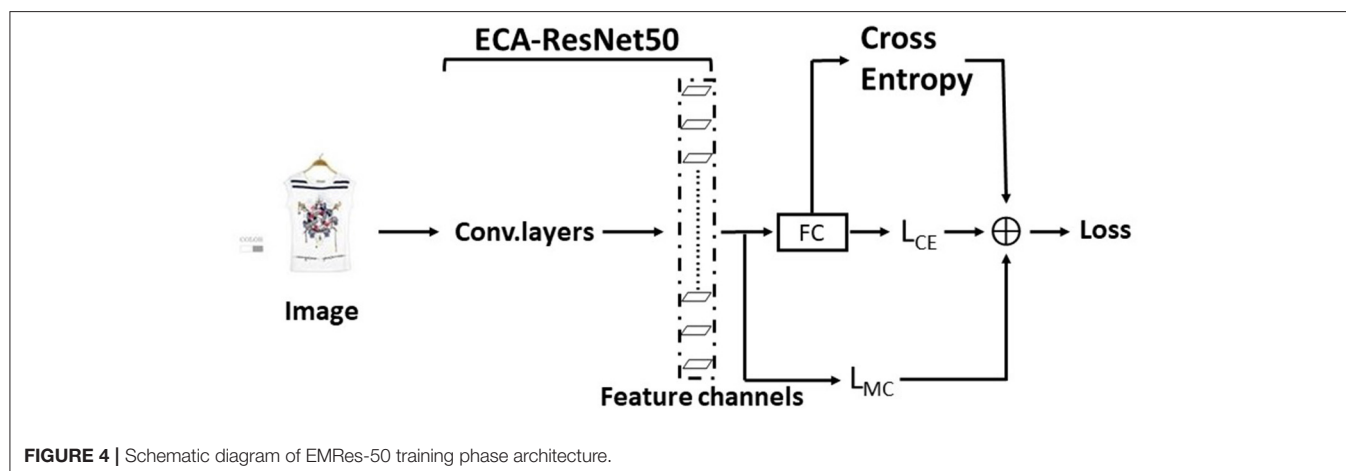


TABLE 2 | EMRes-50 weight update process.**Algorithm: EMRes-50 weighting update process**

Input: Let the network have N layers, W_n is the weight parameter of the last FC layer, W_i is the weight parameter of a layer of the network, $i \in 1 \dots n$,

```

1: While( $i \leq n$ )
2:   if  $i = 1 \dots n - 1$ 
3:      $W_i = W_i - \lambda \frac{\partial (L_{CE} + \mu L_{MC})}{\partial W_i}$  //Except for FC, other weight layers are
      combined by cross-entropy loss and MC-Loss
4:   if  $i = n$ 
5:      $W_n = W_n - \lambda \frac{\partial L_{CE}}{\partial W_n}$  //The last FC layer uses the traditional
      cross-entropy loss to update the weights
6:   Udata by  $W_i$ 
7:   Udata by  $W_n$ 
8: End

```

TABLE 3 | Collar-6 experimental data distribution.

Collar type	Number of training set images	Number of test set images	Total
Round collar	2,480	620	3,100
Lapel collar	2,608	652	3,260
Stand collar	2,464	616	3,080
Hooded collar	2,560	640	3,200
V collar	2,468	617	3,085
Fur lapels	3,122	625	3,747

is difficult to classify images with rich diversity, which helps to learn the features of collars. **Table 3** shows the distribution of the number of images per category in the training and test sets and the total number of that category, each containing about 3,000 RGB of three-channel images. **Figure 5** shows some images of the six categories of collar types.

In order to verify the effectiveness of the classification algorithm proposed in this paper, DeepFashion (Liu et al., 2016), a large publicly available apparel image dataset, is used to verify the effectiveness of the classification algorithm. DeepFashion (<http://mmlab.ie.cuhk.edu.hk/projects/DeepFashion.html>) is a large-scale apparel dataset open to the Chinese University of Hong Kong, with 800,000 images, which contain images from different angles, different scenes, buyer shows, buyer shows, etc. Due to a large number of images in the dataset, and the amount of data in some categories is not large, in order to ensure that the size of the collected images is equivalent to the size of the Collar-6 dataset, the DeepFashion dataset is extracted by the following folder keywords: Dress, Jacket, Jeans, Shorts, Tank, Tee 6 categories, a total of 18,727 images for experimentation. The distribution of experimental data of DeepFashion-6 is shown in **Table 4**.

Experimental Setup

All model experiments in this paper are trained on Intel i7-7700 processor, 1T SSD, 64 RAM, and NVIDIA GTX2080Ti GPU,

using the pytorch framework. Stochastic gradient descent (SGD) is used as the optimization method. The number of iterations is 300. The initial learning rate is set to 1e-2, and the learning rate is adjusted to 1e-3 when the iteration reaches 150. the batch size is set to 32. the size of all images for the experiments is uniformly 224×224 . the comparison networks are compared with EMRes-50 using the cross-entropy loss function in the comparison experiments.

Analysis of Experimental Results

Collar Image Classification Experiments Based on the Collar-6 Dataset

In order to solve the problems of unsatisfactory classification of collar images and imprecise collar feature extraction by traditional convolutional neural networks, the EMRes-50 method is proposed, which adds MC-Loss to the existing channel attention module of the ECA-ResNet network to further ensure that the network focuses as much as possible on the discriminative part and the discriminative part, thus helping the network to perform fine-grained feature learning.

A comparison of the classification accuracy of EMRes-50 with a variant of ResNet or a ResNet-based improved model on Collar-6 is shown in **Table 5**.

As can be seen from **Table 5**, the classification accuracy of EMRes-50 on the Collar-6 dataset obtained the highest in comparison with a variant of ResNet or a model based on ResNet improvements. ResNeXt50 (Xie et al., 2017), SCNet50 (Liu et al., 2020), and Res2Net50 (Gao et al., 2019) are all variants of the ResNet network. ResNeXt50 utilizes group convolution, constructing a parallel stack of blocks with the same topology, and is a simple, highly modular network structure for image classification, but its composition is limited by stacking blocks with the same specifications, and is able to extract The classification accuracy is 73.05%, which is 0.55% lower than that of EMRes-50. Res2Net is designed with finer-grained layer blocks in order to extract more multi-scale features, increasing the range of perceptual fields in each layer, and has a strong multi-scale representation capability, which is suitable for extracting collar images of different scales. However, it ignores the relationship between the global and local position of the collar part in the whole image, so the classification accuracy of Res2Net50 is 73.44%. EMRes-50 introduces MC-Loss based on ECA-ResNet50, which, together with cross-entropy, can make the network focus on both global discriminative regions and local discriminative regions, and the classification accuracy is 0.16% higher than that of Res2Net50 is 0.16% higher than Res2Net50. SCNet50 can effectively improve the range of sensory field through self-correction operation and help the network generate more discriminative feature expressions, but it only takes into account the inter-channel information and local information without considering the global location information, and the effect of classification for collar images is rather poor, with an accuracy of only 66.07%. CBAM (Woo et al., 2018) and SE (Hu et al., 2018) are two classical attention mechanism models, but introducing the attention mechanism only on ResNet50, although it can focus on the collar part, it cannot distinguish the low-level distinguishable features such as different collar edges

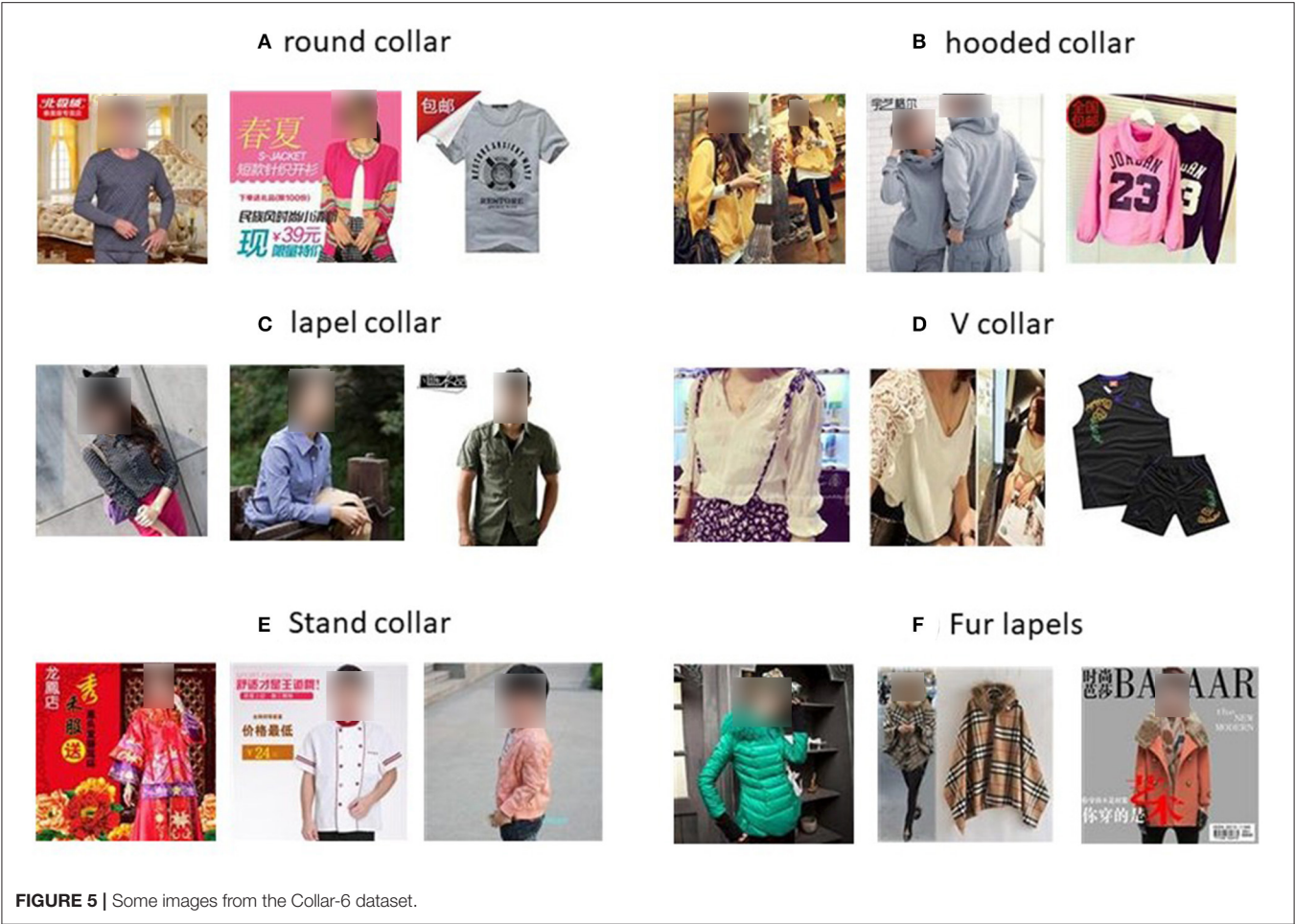


FIGURE 5 | Some images from the Collar-6 dataset.

TABLE 4 | Distribution of experimental data of DeepFashion-6.

Type	Number of training set images	Number of test set images	Total
Dress	2,555	639	3,194
Jacket	2,505	627	3,132
Jeans	2,412	603	3,015
Shorts	2,541	636	3,177
Tank	2,528	632	3,160
Tee	2,439	610	3,049

well, and the classification effect is not satisfactory, only 63.68 and 63.44% accuracy respectively, which are 9.92 and 10.16%.

A comparison of the classification accuracy of EMRes-50 and the lightweight convolutional neural network model on Collar-6 is shown in Table 6.

As can be seen from Table 6, on the Collar-6 dataset, EMRes-50 obtains better classification accuracy on collar images in comparison with the lightweight model. GhostNet (Han et al., 2020) only considers from the perspective of generating feature maps, reducing the total number of parameters it needs and computational complexity, without enhancing the network extraction effect in terms of feature extraction capability, so the

TABLE 5 | Comparison with variants of ResNet or improved models based on ResNet.

Models	Accuracy %
ResNeXt50	73.05
CBAEMRes-50	63.68
SE-ResNet50	63.44
SCNet50	66.07
Res2Net50	73.44
EMRes-50	73.60

classification effect on collar images is The accuracy is only 60.98%, which is 12.62% lower than the accuracy of EMRes-50. The design of the Fire module in SqueezeNet (Iandola et al., 2016) performs model compression by reducing the parameters, and the mixture of 3×3 and 1×1 convolution increases the feature extraction capability of the whole model. However, the SqueezeNet network is not deep, and there are limitations in feature extraction for different types of collars, and the accuracy is 9.84% lower than EMRes-50, which yields 63.76% accuracy. MobileNetV3 (Howard et al., 2019), which uses a large number of 5×5 size convolutional kernels, is not good

TABLE 6 | Comparison with lightweight model.

Models	Accuracy %
MobileNetV3_large	68.48
MobileNetV3_small	63.23
GhostNet	60.98
SqueezeNet1_0	63.76
EMRes-50	73.60

TABLE 7 | Comparison with other models.

Models	Accuracy %
AlexNet	67.53
Xception	70.53
VGG16	63.68
VGG19	65.62
EMRes-50	73.60

for fine-grained collar images because there are no multi-scale convolutional kernels used alternatively classification, MobileNetV3_large and MobileNetV3_small have 68.48 and 63.23% classification accuracy respectively, which are 5.12% and 10.37% less accurate than EMRes-50, respectively.

A comparison of the classification accuracy of EMRes-50 with other classical convolutional neural network models on Collar-6 is shown in **Table 7**.

As can be seen from **Table 7**, EMRes-50 obtains better collar image classification accuracy in comparison with other classical convolutional neural network models on the Collar-6 dataset. AlexNet (Krizhevsky et al., 2012) and VGGNet (Simonyan and Zisserman, 2014) filters are both linear topologies, which means that these networks can only have relatively inflexible perceptual fields and obtain lower classification accuracies for both low, with 63.68% and 65.62% accuracy obtained by VGG16 and VGG19, respectively, and 67.53% accuracy obtained by AlexNet, both of which are lower than the accuracy obtained by EMRes-50. Conventional convolution is a direct extraction of spatial and channel information through a convolutional kernel. Xception (Chollet, 2017), on the other hand, is a convolutional neural network architecture based entirely on depth-separable convolutional layers. As an improved version of InceptionV3, it retains the network's multiscale feature extraction capability, and its model performance on collar image classification is better than AlexNet and VGGNet, obtaining an accuracy of 70.53%, but still 3.07% lower than the accuracy obtained by EMRes-50.

Validation Experiments

To verify the effectiveness of the EMRes-50 method class, the validity of the classification algorithm was verified using the publicly available large apparel image dataset DeepFashion. The experimental results are shown in **Table 8**.

TABLE 8 | Comparison of model accuracy in the DeepFashion-6 dataset.

Models	Accuracy %
AlexNet	83.50
ResNet50_CBAM	82.78
GhostNet	82.84
InceptionV3	73.36
MobileNet_large	83.77
MobileNet_small	83.40
Res2Net	85.13
SCNet	79.57
SqueezeNet1_0	82.03
Xception	85.01
EMRes-50	86.09

It can be seen from **Table 8** that the performance of EMRes-50 on DeepFashion-6 has a certain improvement compared with other convolutional neural networks because after the introduction of MC-Loss in the basic network, the network can capture the discriminative and identifiable performance. There are more distinguishing features, which improve the classification performance of the network to a certain extent. EMRes-50 is 0.08% higher than Xception, which has better accuracy and is higher than other convolutional neural networks. The results show that although EMRes-50 is designed primarily for the collar-6 Collar dataset, it can effectively categorize garment areas without collars while accurately identifying Collar areas. It shows that EMRes-50 can continuously find the distinguishable features of classified objects with different region proportions in an image through algorithm iteration, so as to improve the classification effect. At the same time, the DeepFashion-6 dataset is the same as the Collar-6 dataset. When the classified objects have different angles, different scenes, and other noises, the classification performance of EMRes-50 can still be compared to these two datasets. The improvement indicates that EMRes-50 has a better ability to distinguish the distinguished features. Such results fully verify that EMRes-50 not only has good classification performance on collar images but also shows good classification effects in the field of clothing image classification.

Ablation Experiments

Structural Ablation

The ablation experiments were conducted on two datasets, Collar-6 and DeepFashion-6, with Resnet50 as the base network, and the effects of introducing the attention mechanism ECA block and MC-Loss loss analysis on the experimental effects, respectively. Ablation experiments of EMRes-50 on the Collar-6 dataset are shown in **Table 9**. As can be seen from the table, EMRes-50 improves 14.24% compared to ResNet50, 6.76% compared to ResNet50 by introducing only MC-Loss, and 16.1% compared to ResNet50 by introducing only ECA block. Since the collar part is not solely present in the collar image, the non-collar part also occupies most of the space in the image, which can interfere with the training of the convolutional neural network. The accuracy of ResNet50 is 1.86% lower than that of

TABLE 9 | Models for ablation experiments on the Collar-6 dataset.

Models	Accuracy %
ResNet50	59.36
ResNet50+MC-Loss	66.84
ResNet50+ECA	57.50
EMRes-50	73.60

TABLE 10 | Ablation experiments of the model on the DeepFashion-6 dataset.

Models	Accuracy %
ResNet50	81.47
ResNet50+MC-Loss	84.53
ResNet50+ECA	80.94
EMRes-50	86.09

ResNet50 when only the ECA module is introduced, indicating that the attention module makes the model focus too much on the overall part of the collar and cannot effectively guide the network to focus on the distinguishable areas of the collar, ignoring the differences between different collar types, while the introduction of the MCLoss loss function can consider more distinguishable areas of the collar and guide the network to perform the correct weight optimization for fine-grained classification. The introduction of the MCLoss loss function can guide the network to optimize the correct weights and greatly promote the network to learn the distinguishable local features for fine-grained classification, effectively avoiding the negative impact of the ECA attention module.

EMRes-50 introduces both the ECA module and MCLoss, which makes the network not only focus on the collar region, but also highlight the local regions of different types of collars, and the negative effect brought by the ECA module to the network is transformed into a facilitating effect. Thus, the combined effect of both the ECA module and MC-loss further enhances the feature extraction capability of the network, resulting in a large improvement of the network performance.

The ablation experiments of EMRes-50 on the DeepFashion-6 dataset are shown in **Table 10**. As can be seen from the table, EMRes-50, compared to ResNet50 only introduced MC-Loss improved by 1.56%, and compared to ResNet50 only introduced ECA block improved by 5.15%. The non-category related part of the category image of clothing occupies a larger space of the image, which is not as disturbing to the training of the convolutional neural network as the collar image classification, but EMRes-50 can still bring an improvement in accuracy in the field of clothing image classification, verifying that EMRes-50 can effectively improve the network performance.

Hyperparametric Ablation

Verify the effect of one-dimensional convolutional size k on EMRes-50 and the validity of k size selection. After channel-level global averaging pooling without dimensionality reduction, the ECA module captures local cross-channel interaction information by considering each channel and its k neighbors, where the convolutional kernel size of k represents the coverage

TABLE 11 | Different effects of different k on the ECA module on the Collar-6 dataset.

k	Accuracy %
5	70.82
7	67.37
3	73.60

of local cross-channel interactions, i.e., how many neighbors near that channel are involved in the attention prediction of this channel. In EMRes-50, k is set to 3, 5, and 7. The results are shown in **Table 11**. $k = 3$ gives the best results for EMRes-50, and the accuracy rate decreases with larger k , indicating that for the collar region, which accounts for a smaller percentage of the collar image, the smaller the value of 1D convolution, the easier it is to capture the features of the region and improve the accuracy rate. On the contrary, the larger the value of 1D convolution is, the more noise is introduced, which affects the recognition of collar regions by the network and makes the accuracy rate decrease.

CONCLUSION

In the context of the development of apparel e-commerce, efficient and accurate collar classification is beneficial to merchants for apparel information description, convenient for a wide range of consumers to shop using keyword queries, and promotes the development of the apparel sales industry. In the absence of related research, this paper constructs a collar dataset named Collar-6 and the images contain a lot of noise. Based on ECA-ResNet50 and the introduction of MC-Loss, this paper proposes a fine-grained image classification model based on a small collar dataset, called EMRes-50. Comparative experiments and ablation experiments are conducted on the Collar-6 dataset, and the results show that EMRes-50 can effectively improve the classification performance of the underlying network ECA-ResNet50, and outperforms most classical and novel classification models in recent years, indicating that EMRes-50 can effectively solve the fine-grained collar image classification problem, and the extracted features are more differentiable and enhance the model classification effect. On the other hand, to verify the effectiveness of EMRes-50, comparison experiments are conducted on the public dataset DeepFashion, and EMRes-50 is still able to improve the classification effect of garment image classification, indicating that EMRes-50 can be applied not only to the field of collar image classification but also extended to the field of garment image classification.

AUTHOR'S NOTE

The types of clothing are increasing day by day, and it is becoming more and more important to classify clothing according to its collar design. Nowadays, popular shopping websites all support keyword search for the clothing styles you want to buy, including clothing collar keyword search. However, the product information of the website is often described in the form of a combination of direct image description and key text annotations. If it can be directly described through

images, a lot of time and labor costs can be reduced. In addition, the collar part occupies a small proportion in the entire image. Such image classification is more challenging than ordinary image classification. At present, there are few researches on collar classification and related data sets. Therefore, this paper constructs a six-category small collected data set, and builds a model named EMRes-50 for this data set, and proves the improvement of the model through experiments. It can effectively solve the problem of collar image classification.

DATA AVAILABILITY STATEMENT

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

AUTHOR CONTRIBUTIONS

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

REFERENCES

- Chang, D., Ding, Y., Xie, J., Bhunia, A. K., Li, X., Ma, Z., et al. (2020). The devil is in the channels: Mutual-channel loss for fine-grained image classification. *IEEE Trans. Image Process.* 29, 4683–4695. doi: 10.1109/TIP.2020.2973812
- Chollet, F. (2017). “Xception: deep learning with depthwise separable convolutions,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (California), 1251–1258. doi: 10.1109/CVPR.2017.195
- Gao, S. H., Cheng, M. M., Zhao, K., Zhang, X. Y., Yang, M. H., and Torr, P. (2019). “Res2Net: a new multi-scale backbone architecture,” in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Tianjin.
- Han, K., Wang, Y., Tian, Q., Guo, J., Xu, C., and Xu, C. (2020). “Ghostnet: more features from cheap operations,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Shenzhen), 1580–1589. doi: 10.1109/CVPR42600.2020.00165
- He, K., Zhang, X., Ren, S., and Sun, J. (2016). “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV), 770–778. doi: 10.1109/CVPR.2016.90
- Howard, A., Sandler, M., Chu, G., Chen, L. C., Chen, B., Tan, M., et al. (2019). “Searching for mobilenetv3,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (California), 1314–1324. doi: 10.1109/ICCV.2019.00140
- Hu, J., Shen, L., and Sun, G. (2018). “Squeeze-and-excitation networks,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (Beijing). doi: 10.1109/CVPR.2018.00745
- Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., and Keutzer, K. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size. *arXiv Preprint arXiv:1602.07360*.
- Inoue, N., Simo-Serra, E., Yamasaki, T., and Ishikawa, H. (2017). “Multi-label fashion image classification with minimal human supervision,” in *Proceedings of the IEEE International Conference on Computer Vision Workshops* (Tokyo), 2261–2267. doi: 10.1109/ICCVW.2017.265
- Krizhevsky, A., Sutskever, I., and Hinton, G. (2012). “Imagenet classification with deep convolutional neural networks,” in *Proceedings of Advances in Neural Information Processing Systems* (Toronto), 1097–1105
- Li, X., Wang, W., Hu, X., and Yang, J. (2019). “Selective kernel networks,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Nanjing), 510–519. doi: 10.1109/CVPR.2019.00060
- Liu, J. J., Hou, Q., Cheng, M. M., Wang, C., and Feng, J. (2020). “Improving convolutional networks with self-calibrated convolutions,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Tianjin), 10096–10105. doi: 10.1109/CVPR42600.2020.01011

FUNDING

This work was partially supported by the National Natural Science Foundation of China under Grant Nos. 61962006, 61802035, and 61772091; the Project of Science Research and Technology Development in Guangxi under Grant Nos. AA18118047, AD18126015, and AB16380272; thanks to the support by the BAGUI Scholar Program of Guangxi Zhuang Autonomous Region of China [2016(21), 2019(79)]; the National Natural Science Foundation of Guangxi under Grant Nos. 2018GXNSFAA138005; the Sichuan Science and Technology Program under Grant Nos. 2018JY0448, 2019YFG0106, and 2019YFS0067; Guangxi University Young and Middle-aged Teachers Scientific Research Basic Ability Improvement Project (Grant Nos. 2020KY04031. Project Name: Research on Key Technologies of Intelligent Data Processing in Active Distribution Network environment).

- Liu, Z., Luo, P., Qiu, S., Wang, X., and Tang, X. (2016). “Deepfashion: powering robust clothes recognition and retrieval with rich annotations,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Hong Kong), 1096–1104. doi: 10.1109/CVPR.2016.124
- Nawaz, M. T., Hasan, R., Hasan, M. A., Hassan, M., and Rahman, R. M. (2018). “Automatic categorization of traditional clothing using convolutional neural network,” in *2018 IEEE/ACIS 17th International Conference on Computer and Information Science* (Dhaka: IEEE), 98–103.
- Qin, X., Jiang, J., Fan, W., and Yuan, C. (2020). Chinese cursive character detection method. *J. Eng.* 2020, 626–629. doi: 10.1049/joe.2019.1208
- Simonyan, K., and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv Preprint arXiv:1409.1556*.
- Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., and Hu, Q. (2020). “ECA-net: efficient channel attention for deep convolutional neural networks,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Tianjin), 11534–11542. doi: 10.1109/CVPR42600.2020.01155
- Woo, S., Park, J., Lee, J. Y., and Kweon, I. S. (2018). “Cbam: convolutional block attention module,” in *Proceedings of the European Conference on Computer Vision (ECCV)* (Daejeon), 3–19. doi: 10.1007/978-3-030-01234-2_1
- Wu, D., He, Q., Luo, X., Shang, M., He, Y., and Wang, G. (2019). “A posterior-neighborhood-regularized latent factor model for highly accurate web service QoS prediction,” in *IEEE Transactions on Services Computing* (Chongqing: IEEE).
- Wu, D., Luo, X., Shang, M., He, Y., Wang, G., and Wu, X. (2020). “A data-characteristic-aware latent factor model for web service QoS prediction,” in *IEEE Transactions on Knowledge and Data Engineering* (Chongqing). doi: 10.1109/TKDE.2020.3014302
- Wu, D., Luo, X., Shang, M., He, Y., Wang, G., and Zhou, M. (2021a). A deep latent factor model for high-dimensional and sparse matrices in recommender systems. *IEEE Trans. Syst. Man Cyber. Syst.* 51, 4285–4296. doi: 10.1109/TSMC.2019.2931393
- Wu, D., Shang, M., Luo, X., and Wang, Z. (2021b). “An L1-and-L2-norm-oriented latent factor model for recommender systems,” in *IEEE Transactions on Neural Networks and Learning System* (Chongqing). doi: 10.1109/TNNLS.2021.3071392
- Wu, E. Q., Hu, D. W., Deng, P. Y., Tang, Z., Cao, Y., Zhang, W. M., et al. (2020). Non-Parametric Bayesian prior inducing deep network for automatic detection of cognitive status. *IEEE Trans. Cyber.* 51, 5483–5496. doi: 10.1109/TCYB.2020.2977267
- Wu, E. Q., Zhou, G. R., Zhu, L. M., Wei, C. F., Ren, H., and Sheng, S. F. (2019). Rotated sphere Haar Wavelet and deep contractive auto-encoder network with fuzzy Gaussian SVM for pilot’s pupil center detection.

- IEEE Trans. Cyber. Early Access* 51, 332–345. doi: 10.1109/TCYB.2018.2886012
- Wu, Y., Qin, X., Pan, Y., and Yuan, C. (2018). “Convolution neural network based transfer learning for classification of flowers,” in *2018 IEEE 3rd International Conference on Signal and Image Processing (ICSIP)* (Nanning: IEEE), 562–566. doi: 10.1109/SIPROCESS.2018.8600536
- Wu, Y., Zhang, K., Wu, D., Wang, C., Yuan, C. A., Qin, X., et al. (2020). Person re-identification by multi-scale feature representation learning with random batch feature mask. *IEEE Trans. Cogn. Dev. Syst.* 13, 865–874. doi: 10.1109/TCDS.2020.3003674
- Xie, S., Girshick, R., Dollár, P., Tu, Z., and He, K. (2017). “Aggregated residual transformations for deep neural networks,” in *Proceedings of the Computer Vision and Pattern Recognition* (San Diego), 5987–5995. doi: 10.1109/CVPR.2017.634
- Yuan, C., Wu, Y., Qin, X., Qiao, S., and Pan, Y. (2019). An effective image classification method for shallow densely connected convolution networks through squeezing and splitting techniques. *Appl. Intell.* 49, 3570–3586. doi: 10.1007/s10489-019-01468-7
- Zhao, B., Wu, X., Feng, J., Peng, Q., and Yan, S. (2017). Diversified visual attention networks for fine-grained object classification. *IEEE Trans. Multimedia* 19, 1245–1256. doi: 10.1109/TMM.2017.2648498
- Conflict of Interest:** The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.
- Publisher’s Note:** All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

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



An Adaptive Information Security System for 5G-Enabled Smart Grid Based on Artificial Neural Network and Case-Based Learning Algorithms

Chengzhi Jiang^{1,2*}, Hao Xu^{1,2*}, Chuanfeng Huang¹ and Qiwei Huang¹

¹ School of Economics and Management, Nanjing Institute of Technology, Nanjing, China, ² School of Information Management, Nanjing University, Nanjing, China

OPEN ACCESS

Edited by:

Di Wu,
Chongqing Institute of Green and
Intelligent Technology (CAS), China

Reviewed by:

Avishek Nag,
University College Dublin, Ireland
Ming Su,
Beijing University of Posts and
Telecommunications (BUPT), China

*Correspondence:

Chengzhi Jiang
jcz@njit.edu.cn
Hao Xu
xhnjit@njit.edu.cn

Received: 10 February 2022

Accepted: 11 March 2022

Published: 14 April 2022

Citation:

Jiang C, Xu H, Huang C and Huang Q
(2022) An Adaptive Information
Security System for 5G-Enabled
Smart Grid Based on Artificial Neural
Network and Case-Based Learning
Algorithms.
Front. Comput. Neurosci. 16:872978.
doi: 10.3389/fncom.2022.872978

With the deployment of 5G Internet of Things (IoT) in the power system, the efficiency of smart grid is improved by increasing two-way interactions in different layers in smart grid. However, it introduces more attack interfaces that the traditional information security system in smart grid cannot response in time. The neuroscience-inspired models have shown their effectiveness in solving security and optimization problems in smart grid. How to improve the security mechanism in smart grid while taking into account the optimization of data transmission efficiency using neuroscience-inspired algorithms is the problem to be solved in this study. Therefore, an information security system based on artificial neural network (ANN) and improved multiple protection model is proposed. Based on the ANN algorithm, the link state sample space is used to train the model to obtain the optimal transmission path in 5G power communication network. Integrating the intelligent link state module, the zero-trust security protection platform using case-based learning algorithm is designed and taken as the first protection, the network security logical isolation facility is taken as the second protection, and the forward and backward isolation facilities are set as the third protection to achieve the strengthened security of 5G IoT in smart grid. The experimental results show the efficiency and effectiveness of the proposed algorithms. In addition, the experimental results also show that the proposed system can resist malicious terminal access, terminal hijacking, data tampering and eavesdropping, protocol fuzzy, and denial-of-service attacks, so as to reduce the security risks of 5G IoT in smart grid. Since the proposed system can be easily integrated into the existing smart grid structure in China, the proposed system can provide a reference for the design and implementation of 5G IoT in smart grid.

Keywords: information security, artificial neural network, case-based learning, smart grid, zero trust

INTRODUCTION

The development of smart grid depends on the intelligent infrastructure to enable a control-feedback loop. With the expansion to distribution side and user load side in the smart grid, the deep integration of 5G technology into the smart grid becomes an inevitable trend (Ma et al., 2021). The 5G technology including 5G network slicing technology can be advantageous in

supporting the services of the smart grid such as grid monitoring, precise load control, intelligent distribution automation, and advanced metering infrastructure (AMI) (Matinkhah and Shafik, 2019; Forcan et al., 2020; Liu R. et al., 2021). A 5G communication has the characteristics of high bandwidth, low delay, high reliability, and low power consumption (Zhang et al., 2019). The 5G communication technology has great application potential in scenarios such as enhanced mobile bandwidth, large-scale terminal access, and ultra-low delay communication (Zhang, 2021). Using the advantages of 5G communication technology can not only facilitate the collection and analysis of power consumption data, but also improve the accuracy of power load control. In the power Internet of Things (IoT), building 5G cognitive radio network model and applying it to traditional collection and inspection services can improve the perception and transmission performance of a large number of user nodes (She et al., 2021). The advantages of 5G technology in future smart grid may include that it provides the data acquisition and visualization ability for multiple layers of smart grid (Ahmadzadeh et al., 2021).

At present, the power optical fiber private network communication is mainly used in the power system in China, which has high security and reliability. Due to the limited cost, fiber core resources and mobile operations, it is unable to cover a large number of power business terminals, so that 5G and other wireless communication methods need to be used as a supplement to the optical fiber private network (Wu

et al., 2020; Li et al., 2021). However, the 5G networks do not provides end-to-end security for applications in smart grid where new types of threats may be introduced including security misconfiguration at mobile edge computing host (MECH) and IoT device security problems (Borgaonkar and Jaatun, 2019). The critical applications in smart grid requires additional measures against unauthorized access to the network while wireless technology such as 5G is applied (Ghanem et al., 2021). In addition, denial-of-service (DoS) or false data injection attacks may be launched against different parts of AMI using 5G in smart grid, leading to financial losses or even physical damages (Saghezchi et al., 2017). Therefore, the security of power terminal side is very important for the normal operation of power system communication network. Whether the service terminal of power system in China can be safely connected has become an important research direction of researchers in the field of power safety. Meanwhile, to facilitate the deployment of 5G applications, the security measures need to be easily integrated into the existing power industry security protection strategies (Li et al., 2020).

The current research on 5G IoT in smart grid mainly focuses on meeting different business needs, improving business processing efficiency and network scalability. In terms of security protection, it is mainly based on the existing security protection strategies and equipment that can no longer meet the security requirements in the IoT and 5G era. Therefore, to strengthen its security protection mechanism while improving the efficiency of 5G IoT, this study proposes an improved information system based on ANN and improved multiple protection mechanism, which can be easily integrated into the existing smart grid security architecture. The proposed method evaluates, learns, and predicts the link states in the process of 5G power communication (Hu et al., 2019), and adopts the multiple security protection method in combination with the idea of double isolation power security access area (Cao et al., 2019a) and the encryption, authentication method (Zhao, 2020) to improve the transmission efficiency of power 5G communication while meeting the security requirements in the process of power 5G communications.

RELATED WORK

Scholars in related fields have studied the power communication access scheme and achieved some research results. Li et al. (2018) designed an intelligent power distribution terminal access architecture based on the integration of multiple technologies such as wireless sensor network (WSN), wireless local area network (WLAN) and wired private network, and adopted data hierarchical encryption, access network security classification and isolation to ensure network security. The architecture can effectively meet a variety of business needs of power distribution terminals. Chen et al. designed a joint deployment architecture based on multi-access edge computing (MEC), and designed a task scheduling mechanism by deploying MEC network elements on the access side and the core network side (Chen et al.,

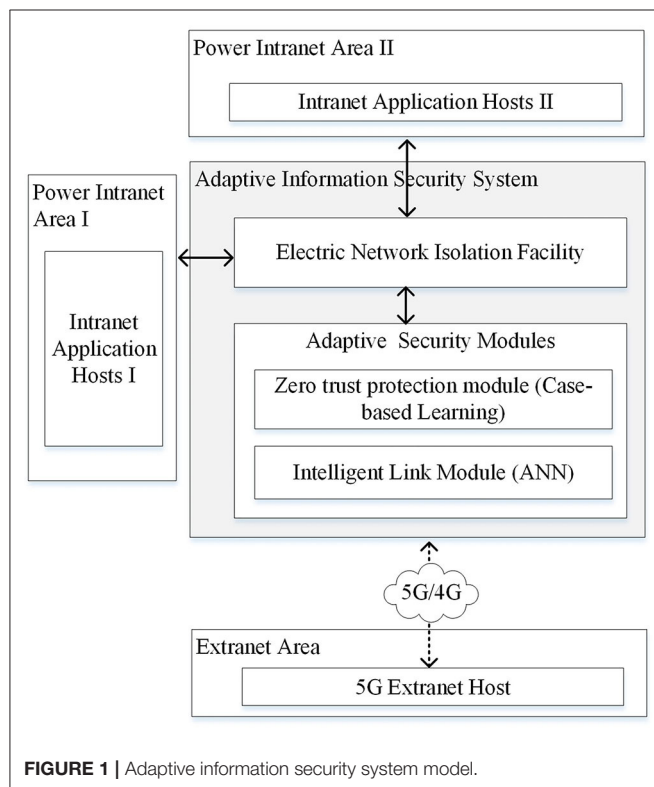


FIGURE 1 | Adaptive information security system model.

2018). The deployment architecture can effectively allocate MEC processing nodes and effectively improve scalability. Saghezchi et al. proposed a security architecture incorporating intrusion detection system (IDS) into AMI to protect the integrity of the information exchanged (Saghezchi et al., 2017).

The neuroscience-inspired methods [including artificial neural network (ANN)] have shown the effectiveness in solving security and optimization problems in smart grid. To mitigate the false data injection attacks in smart grid, the graph neural network (GNN) based detector incorporating physical connections and exploiting spatial correlations (Boyaci et al., 2022) or the detector combining predictions of Kalman filter and recurrent neural network (RNN) (Wang et al., 2022) can be effective methods. The RNN can also be applied to classify multiclass attacks for power systems with high accuracy (Hong et al., 2020). In addition, neuroscience-inspired methods can be applied to optimization problems in smart grid such as link quality estimation in smart grid WSN (Sun et al., 2017), load monitoring (Zhou et al., 2022), short-term load forecasting (Deng et al., 2021), and power user behavior feature classification (Deng et al., 2022).

As the organizational boundaries have become blurred, the zero-trust architecture has been attracting information security researches and is expected to be further explored and implemented in future digital systems (Wylde, 2021). The power grid security architecture can be established based on zero-trust architecture to provide dynamic security policies according to the trust of the access entities (Liu T. et al., 2021). The specific implementation of zero-trust architecture is considered as the improvement on continuous risk management. The intelligent decision support system using case-based reasoning (CBR) and rule-based machine learning may be

used to significantly reduce the risks in software development (Asif and Ahmed, 2020).

Inspired by the adaptive ability and effectiveness of neuroscience-inspired methods and zero-trust models in the above researches, we attempt to design algorithms using ANN and case-based learning to improve the security and communication efficiency in 5G IoT environment of smart grid.

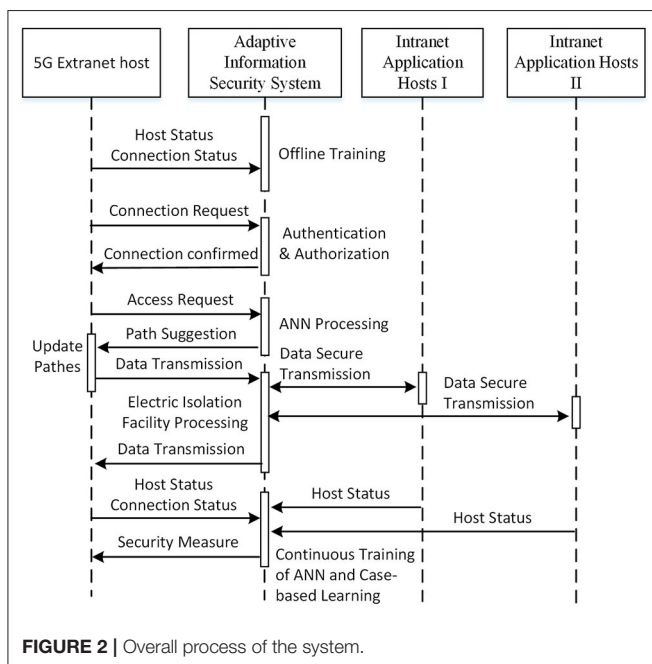
SYSTEM MODEL

According to the power security regulations and current implementation of smart grid information infrastructure in China, an information security system model is proposed as shown in **Figure 1**.

As shown in **Figure 1**, the adaptive information security system is implemented in the secure access area in the power information network, which consists of a zero-trust protection module based on case-based learning and an intelligent link module based on ANN. The details of those two main components will be described in Sections Related Work and System Model. The power intranet area I represents the network area where power production and control related software and hardware are implemented such as supervisory control and data acquisition (SCADA) and energy management system (EMS). Servers and equipment running these applications are represented by intranet application hosts I. The power intranet area II represents the network area where power management related data is processed such as office automation (OA) and enterprise resource planning (ERP). Servers and equipment running these applications are represented by intranet application hosts II. The power intelligent terminal and other equipment that implement in-field monitoring or control functions *via* public network such as 5G/4G, narrow band internet of things (NB-IoT), and long range radio (LoRA) can be represented by 5G extranet host.

Since the security level and requirements of power intranet area I and II are different, customized security policies and measures should be made. The overall secure communication process is shown in **Figure 2**.

It can be seen from **Figure 2** that the initialization of the proposed system is completed by offline training while acquiring status of connections and hosts for a period. First, 5G extranet host initiates a connection request to proposed system that verifies the identity of 5G extranet host. If the identification process succeeds, the appropriate authority is configured to extranet host. Then, 5G extranet host sends a request for data transmission path with its status and requested time slot. The proposed system produces a suggested path for extranet host and the data transmission is processed. The states of links between the proposed system and 5G extranet host are updated periodically so that an up-to-date suggested path can be produced by the proposed system. The security risks in intranet and extranet are continuously monitored by the proposed system and the credibility of each active user in the network is evaluated accordingly. The authority for each active user may be adjusted



according to its real-time credibility so that the multiple security protection is strengthened.

The following sections will describe the intelligent link state module and improved multiple protection model in details.

INTELLIGENT LINK MODULE BASED ON ANN

In the intelligent link module, an ANN algorithm is applied to design an adaptive routing algorithm to obtain the link states in 5G communication network. The 5G and 4G communication modes are both supported in the communication network. Through the forward conduction and the backward conduction, the deep neural network operation is completed (Liu et al., 2020).

Suppose there is a 5G power communication network with N nodes, $\{D_1, D_2, \dots, D_N\}$ represents the node set. The loads of the network nodes are collected during the collection time period Δt of the cognitive plane, while the packet loss rates of the transmission paths from the source node to the target node are calculated. According to the transmission performance requirements of power services, the packet loss rates of the transmission paths are divided into four categories from low to high as $\{0: \text{Ultra low}; 1: \text{Low}; 2: \text{Average}; 3: \text{High}\}$. We can use $l = (\vec{x}_{ij}, ts_{ij}, y_{ij})$ as a data sample where $\vec{x}_{ij} = \{D_i, D_{i+1}, \dots, D_j\}$, ts_{ij} represents the collection time span, y_{ij} represents the categorized packet loss rate from node i to node j , and $y_{ij} \in \{0, 1, 2, 3\}$. Hence, the sample space including the data label y_{ij} is represented as follows:

$$Y = \left\{ (\vec{x}_{ij,1}, ts_{ij,1}, y_{ij,1}), (\vec{x}_{ij,2}, ts_{ij,2}, y_{ij,2}), \dots, (\vec{x}_{ij,n}, ts_{ij,n}, y_{ij,n}), \dots \right\} \quad (1)$$

The forward conduction that outputs link state prediction value is completed based on the non-linear function formed by each layer node in the deep neural network. The forward conduction expression is as follows:

$$Y_{(l,k)}(x) = F \left(\sum_{i=1}^n (w_{i,k}^l \times x_i + b_i^l) \right) \quad (2)$$

where $k = 1, 2, \dots, n$, $w_{i,k}^l$ represents the weight from neuron k of layer $(l+1)$ to neuron i of layer l , F and w denote the non-linear function and the weight matrix, respectively, and b_i^l represents the bias of neuron i of layer l .

The loss function is used to express the error between the sample space and the output value of neural network. The loss function is shown as follows:

$$J(w, b; x, y) = \frac{1}{2n} \sum_{i=1}^n \|Y(w, b, x^i) - y^i\|^2 \quad (3)$$

where b represents the square loss, x^i represents the absolute value loss, and w represents the mean square error loss.

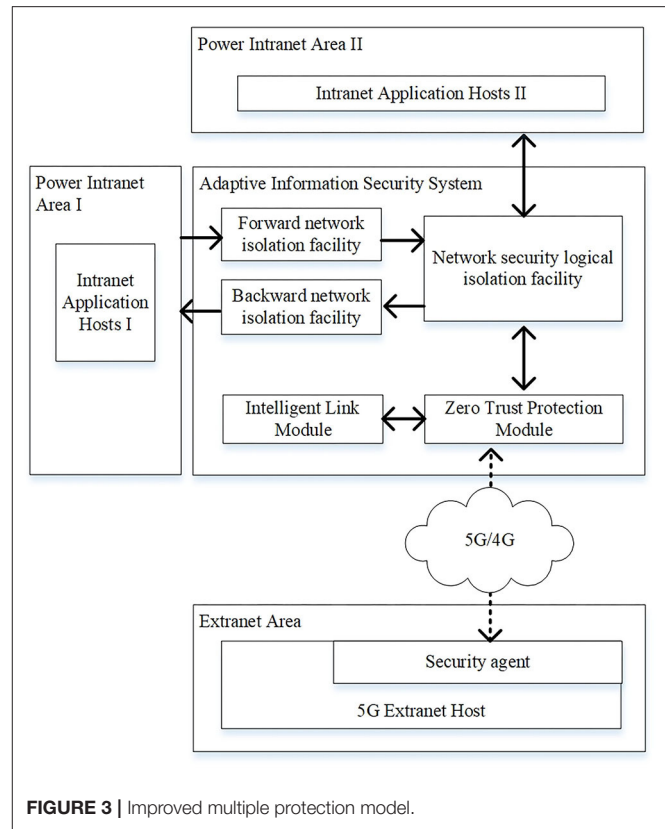


FIGURE 3 | Improved multiple protection model.

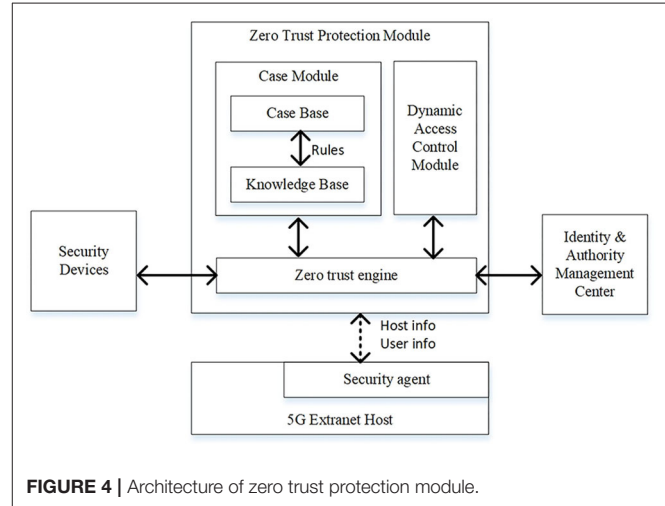


FIGURE 4 | Architecture of zero trust protection module.

The gradient descent method is selected to reduce the error between the calculated sample value and the predicted value. Using the gradient descent method and step-by-step iterative solution, the predicted value of link state, the minimum value of sample space loss function and model parameters can be obtained after completing the backward conduction. The backward function is shown as follows where $i = l - 1$.

$$\delta^{i,l} = (w^{l+1})^T \times \delta^{i,l+1} \times Y'_{(i,l)} \quad (4)$$

The updating formulas of w and b are shown as follows:

$$w^l = w^l - \alpha \sum_{i=1}^n \delta^{i,l} \times (Y_{i,l-1})^T \quad (5)$$

$$b^l = b^l - \alpha \sum_{i=1}^n \delta^{i,l} \quad (6)$$

where α represents the iteration step. We can set the threshold value as ε . When the updated value of w and b are less than the threshold value, the calculation will be terminated. Input the test set samples into the model, and count the error between the model output results and the sample values. Repeating the above process until the error is lower than the predefined threshold, the accuracy test is completed.

The ANN algorithm can be applied to 5G power communication with complex changes, and output the results most consistent with the current environment according to the real-time change of link state in the network (Ge et al., 2020). The link state sample space is input into the model. After the model passes the hidden layer operation, select the softmax function to apply to the output layer, output the probability value of each path (Zhu et al., 2020).

The application plane includes network applications such as routing and network virtualization. The cognitive plane is composed of switches and other devices, and the control plane refers to the controller in the logic set. After receiving the service request sent by the control plane, the application plane forwards it to the cognitive plane. When the output path of the cognitive plane is still the original path, the decision information is set according to the initial routing information table. When the output path of the cognitive plane changes, the new transmission path is sent back to the control plane, and the routing information table is updated by the control plane in real time.

After a fixed interval, the control plane needs to reset the network route. It updates the routing table information in real time (Xu et al., 2018) and transmits the updated routing table to the control layer that controls the cognitive plane to retrain the model, and updates the model in real time after training. Through the above steps, an adaptive routing algorithm is designed using neural network model. Through the forward conduction and the backward conduction, the deep neural network operation is completed to obtain the optimal transmission path in 5G communication network.

IMPROVED MULTIPLE PROTECTION MODEL BASE ON CASE-BASED LEARNING

As shown in **Figure 3**, the improved multiple protection model is composed of the zero-trust protection module, network security logical isolation facility, forward and backward network isolation facility. At present, the power terminals mainly focus on the realization of business functions, and their security functions generally are not fully considered. They need to be

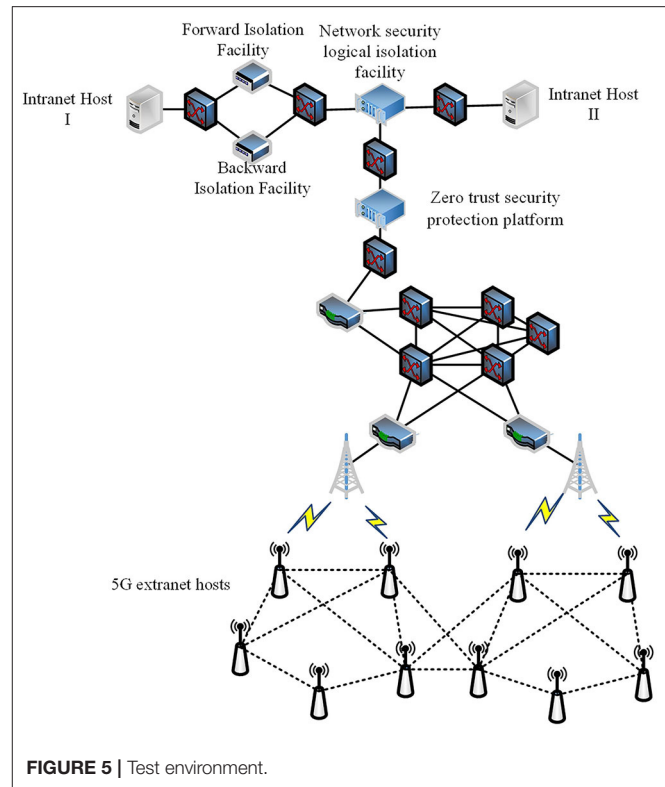


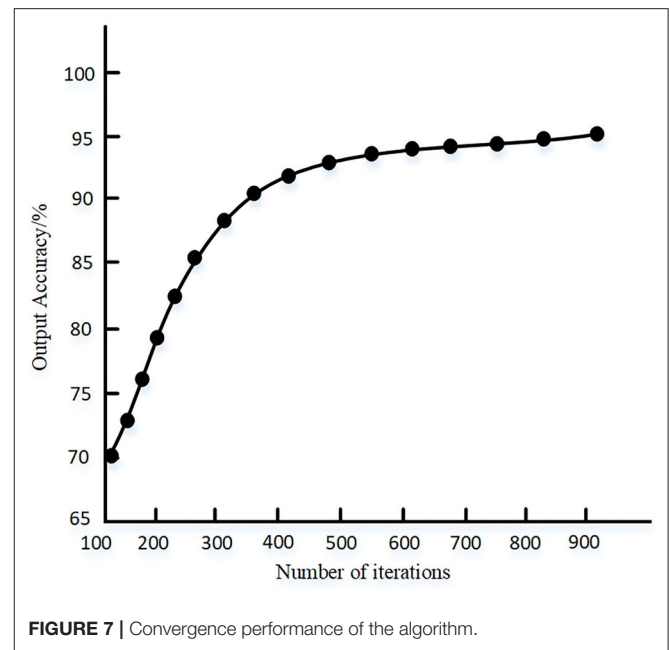
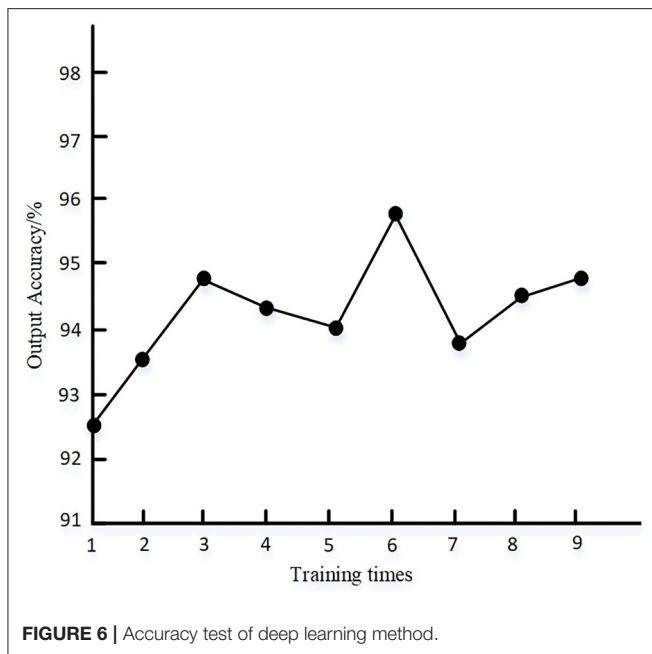
FIGURE 5 | Test environment.

improved in terms of access authorization, audit, and network attack protection (Cheng et al., 2020). Therefore, the zero-trust protection module in the secure access area not only serves as a boundary isolation facility, but also carries out continuous trust and risk assessment for 5G external network hosts. The zero-trust protection module integrates the lightweight encryption and authentication center that uses the identity-based cryptosystem (IBC) or combined public key (CPK) system to generate and distribute the keys to the 5G external network hosts. The network security logical isolation facility in the security access area mainly implements the gate isolation function and the power protocol data security filtering function (Han et al., 2019). The forward and backward network isolation facilities in the secure access area use the existing devices or use the enhanced forward and backward isolation devices (Cao et al., 2019b).

The zero-trust architecture can provide active defense ability and end-to-end security enforcement in a 5G smart application environment where a four-dimensional framework may be designed including subject, object, environment, and behavior (Chen et al., 2021). In power industry of China, the credit management and risk assessment are also paid attentions, considering the risks in the power market transactions (Cai et al., 2020). Thus, a CBR algorithm is proposed in the zero-trust protection module to implement the continuous credit and risk management.

The operational process of improved multiple protection model includes the following steps:

Step 1. The 5G extranet host establishes a network connection with the zero-trust protection module that verifies its identity



information. If it is a legal terminal, an encrypted transmission channel is established and access rights are configured. The 5G extranet host requests the optimal path from the 5G communication link optimization service that then returns the result after calculating the predicted value of the optimal path. The zero-trust protection module evaluates the trust and risk value of the extranet host by monitoring the status and the behavior of the extranet host in real time, and adjusts the access authority of the extranet host according to the CBR algorithm that will be described later in this section.

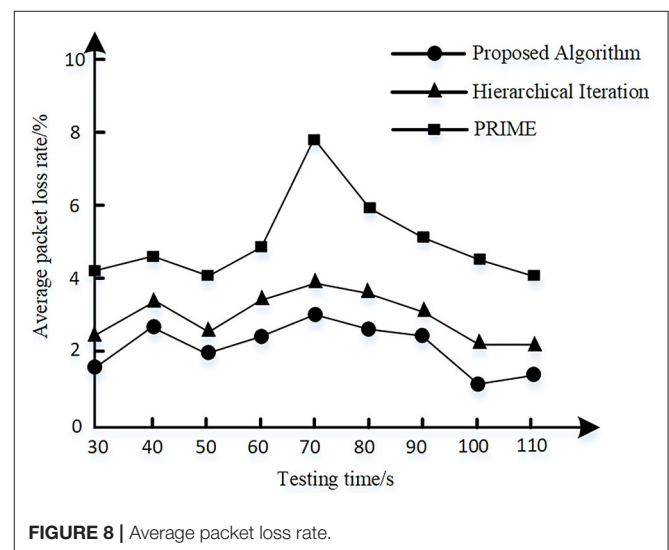
Step 2. After receiving the cipher text sent by 5G power communication network, the zero-trust protection module decrypts the cipher text and transmits the plaintext to the network security isolation facility.

Step 3. After receiving plaintext data, the network security isolation facility implements network protocol stripping (Wang, 2018), and performs security filtering on the obtained data based on pattern matching and feature filtering methods. The plaintext is then signed after security filtering. If the extranet host needs to access the service application of Intranet area II, go to Step 4. If the extranet host needs to access the service application of Intranet area I, go to Step 5.

Step 4. Send the signed data to the access gateway of Intranet area II, encapsulate the network protocol and send the message to the intranet application host II, and the communication of Intranet area II host ends.

Step 5. Convert the signed data into private protocol message of backward isolation facility and output it to backward isolation facility.

Step 6. After receiving the private protocol message, the backward isolation facility performs signature verification, data content filtering and validity check, and sends the data to the access gateway of Intranet area I.



Step 7. The access gateway of Intranet area I encapsulates the data with network protocol and sends it to the intranet application host I, and the communication of Intranet area I host ends.

Step 8. The communication process between the intranet host and the extranet host is opposite to the above process.

The architecture of zero-trust protection module is shown in **Figure 4**.

As shown in **Figure 4**, the zero-trust security platform is comprised of a case module, a dynamic access control module and a zero-trust engine. The operational process of the platform will be described following the design and implementation of CBR method. The CBR method is a recycling process including

five phases: modeling, search, reuse, review, and retain (Chourib et al., 2020). Each case ($case_i$) in the case base of case module is modeled in Formula (7).

$$\begin{aligned} case_i &= \{caseID_i, state_i, event_i, credit_i, group_i\} \\ state_i &= \{ID_{E_i}, ID_{U_i}, type_i, biz_i, ip_i\} \\ event_i &\in \{time_i, freq_i, tgt_i, vol_i, cfg_i, warn_i, usre_i\} \\ credit_i &\in \{cred_{i,1}, cred_{i,2}, \dots, cred_{i,n}\} \\ effe_i &\in \{e_{i,1,2}, e_{i,2,3}, \dots, e_{i,n-1,n}\} \\ group_i &\in \{CaseID_1, \dots, CaseID_{i-1}, CaseID_{i+1}, \dots, CaseID_n\} \end{aligned} \quad (7)$$

where $state_i$ represents the status of 5G extranet host in $case_i$ where ID_{E_i} is the unique identity name of the 5G extranet host if IBC system is adopted or the equipment certificate otherwise. Here, ID_{U_i} is the user identity certificate in the host, $type_i$ is the type of the host, biz_i is the power business running in the host, and ip_i is the IP address of the corresponding host. Also, $event_i$ represents the events encountered that may be the abnormal behaviors in terms of data transmission time ($time_i$), data transmission frequency ($freq_i$), data transmission target (tgt_i), data transmission volume (vol_i), configuration change (cfg_i), and user defined event ($usre_i$). Now, $credit_i$ is the credits record of the last n credits of ID_{E_i} and ID_{U_i} . records the effectiveness evaluated for each change in $credit_i$. Lastly, $group_i$ represents the set of case IDs related to $case_i$.

The knowledge base is comprised of power business templates and rule sets as defined in Formula (8).

$$\begin{aligned} Template_i &= \{bizID_i, bizText_i, time_i, freq_i, tgt_i, vol_i\} \\ Rule_{default} &= \{bizID, event, mea_{default}\} \\ Rule_i &= \{event_i, mea_i\} \end{aligned} \quad (8)$$

where $Template_i$ represents a template for a specific power business. Here, $bizID_i$ and $bizText_i$ denote the identity number and description of a power business, respectively. Also, $time_i, freq_i, tgt_i, vol_i$ are the data transmission time, frequency, target, and volume, respectively, defined by business personnel based on the regular operations of business applications. Now, $Rule_{default}$ defines the default measures when an event occurs in a business application environment. Also, $Rule_i$ represents a rule defined in the rule sets that decides which measures in mea_i can be adopted when an event in $event_i$ occurs. Therefore, the recycling five phases in CBR may include the following steps:

Step 1. Modeling. The case to be solved can be modeled as $\{sta_i, evt_i\}$. evt_i can be collected by zero-trust engine from security facilities such as IDS, firewall, and UTM. Then, the sta_i can be collected by zero-trust engine from security agents installed in related extranet host and the identity and authority management center.

Step 2. Search. First, sta_i and evt_i are searched in the case base. If $sta_i \in case_i$ AND $evt_i \in case_i$, go to Step 3.1. If $sta_i \notin case_i$ AND $evt_i \in case_i$, go to Step 3.2. Otherwise, go to Step 3.3.

Step 3. Reuse. The information and knowledge from similar case are used to form the solution for encountered case.

Step 3.1. If the $e_{i,n-1,n}$ in $effe_i$ is positive, reuse the last credit change measure ($cred_{i,n} - cred_{i,n-1}$) in $credit_i$ of $case_i$. Go to Step 4. Otherwise, go to Step 3.3.

Step 3.2. If $type_i, biz_i$ in sta_i equals to $type_i, biz_i$ in $state_i$ of $case_i$, go to Step 3.1. Otherwise, go to Step 3.3.

Step 3.3. Execute $Rule_{default}$.

Step 4. Review. The zero-trust engine collects the information from security devices to evaluate the effectiveness of the reused solution.

Step 5. Retain. Update $case_i$ or add a new case to the case base.

In summary, the improved multiple protection model implements the triple security protection from the following aspects:

- (1) The zero-trust protection module in the secure access area implements the first boundary security isolation. The zero-trust security protection platform monitors the data access, configuration update, and other behaviors of 5G extranet hosts in real time, dynamically evaluates the security risks of 5G extranet hosts and controls the dynamic access rights. The advantage of using zero-trust protection module that integrates lightweight encryption and authentication module is that it reduces the computing capability requirements of 5G external network host, and can continuously monitor and control the security of 5G terminals, which can effectively reduce the access risk of extranet host. The zero-trust protection module avoids the security risk caused by the traditional one-time authorization and permanent effectiveness so as to improve the traditional security model.
- (2) As the second protection of the model, the network security logical isolation facility implements data security filtering and network logic isolation, and ensures the encryption and authentication of data interaction between the zero-trust security protection platform and the network security isolation facility.
- (3) As the third protection of the model, the forward and backward isolation facilities are used to block the TCP connection, control the information flow access process, and implement the content filtering in the communication process.

EXPERIMENTS AND RESULTS ANALYSIS

To verify the improvement of communication efficiency and the network security of the proposed system based on ANN and multiple protection model, a test and verification environment combining virtual and reality based on OPNET and security equipment is built as shown in **Figure 5**. The environment is implemented in an experimental 5G power IoT scenario of State Grid Corporation of China (SGCC).

First, through experiment 1, the effectiveness and efficiency of the link state algorithm are verified. The dataset is acquired in experimental 5G power IoT scenario for 2 days from Friday to Saturday and then it is marked manually. The ratio of training set to test set of deep learning network is 7:3, and the

number of nodes in input layer and output layer are set to 19 and 4, respectively. The full connection mode is adopted, and the softmax function is set as the activation function of the model. The number of iterations and learning rate are 1,000 and 0.1, respectively. The statistical results of deep learning output accuracy under different training times are shown in **Figure 6**.

It can be seen from **Figure 6** that the output accuracy of the proposed method is higher than 92% after multiple tests, indicating that the parameters of the deep learning method set by the proposed method can meet the requirements of output accuracy. The convergence of the algorithm when the proposed method is randomly selected for single training is shown in **Figure 7**.

It can be seen from **Figure 7** that when the data space samples are set, the proposed method has fast convergence speed, and the training accuracy can reach about 95% when the algorithm tends to be stable.

A 5G extranet host is set as the data sending node and the zero-trust platform is set as the data receiving node. The communication quality of each link is randomly set. The hierarchical iteration algorithm as provided in Hu et al. (2019) and powerline intelligent metering evolution (PRIME) algorithm as provided in Aruzuaga et al. (2010) are selected as the comparison method. The average packet loss rate of 5G power communication network is calculated where noise interference is randomly added to a link node at 60s. The results are shown in **Figure 8**.

It can be seen from **Figure 8** that the average packet loss rate of the proposed algorithm is lower than the other two algorithms and is $\sim 0.6\%$ lower than the hierarchical iteration algorithm. When noise interference is added, the proposed and hierarchical iteration algorithm both can adjust adaptively and reduce the packet loss rate.

Second, through experiment 2, the performance of 5G power communication security system is tested and verified. The test results are shown in **Table 1**.

As shown in **Table 1**, the authentication, encryption, and decryption delay between 5G extranet host and zero-trust platform is <6 ms each time, accounting for a small proportion in the overall communication delay. In the proposed system, the time delay mainly lies in the time delay of isolation facilities in the power system. Due to its data security filtering functions and technical architecture, the time delay of network security logical isolation device is greater than that of the forward and backward isolation devices. The overall bandwidth limitation in the proposed system mainly lies in the backward isolation device (Boyaci et al., 2022). The bandwidth between 5G extranet host and zero-trust platform can meet the large bandwidth requirements of video monitoring and other applications.

Finally, according to the security risks identified in 5G power communication scenario, the IXIA PerfectStorm ONE testbed is used to verify the security of the proposed system through experiment 3. The test results are shown in **Table 2**.

According to **Table 2**, the proposed system can resist malicious terminal access, terminal hijacking, data tampering and eavesdropping, protocol fuzzy and DoS attacks, so as to reduce the security risk of 5G power communication.

TABLE 1 | Security performance test results of proposed system.

Test items	Test results
Latency from 5G extranet host to intranet host I	<100 ms
Latency from 5G extranet host to intranet host II	<90 ms
Authentication delay between 5G extranet host and zero-trust platform	<5 ms/time
Data encryption and decryption delay between 5G extranet host and zero-trust platform	<1 ms/time
Communication bandwidth between 5G extranet host and zero-trust platform (downlink)	>200 Mbps
Communication bandwidth between 5G extranet host and zero-trust platform (uplink)	>70 Mbps

TABLE 2 | Security test of the proposed system.

Test items	Test results
Malicious 5G terminal attempts to access	Access denied
Legitimate 5G terminal hijacked	Authority of terminal is degraded and the terminal is then disconnected
5G network data tampering	Failed
5G network data eavesdropping	Failed
Protocol fuzzy test	The system operates normally
DOS attack/200 Mbps	The system operates normally

In summary, experiment 1 verified the efficiency and performance of the intelligent link state algorithm, experiment 2 verified the secure communication performance of the proposed system, and experiment 3 verified the security of the proposed system. From these three experiments, it can be seen that the intelligent link state algorithm and improved multiple protection model proposed in this system demonstrated satisfied transmission efficiency and security performance, which may meet the demands of power 5G applications.

CONCLUSIONS

In this study, a 5G power security system is proposed where an intelligent link state algorithm and an improved multiple protection model are designed. The intelligent link state algorithm is based on the deep learning method so as to suggest the optimal data transmission path between the 5G extranet host and the zero-trust security platform. The multiple protection model is improved *via* adopting the zero-trust architecture and CBR methodology. The details and operational process of the proposed system including link state algorithm and CBR algorithm are described. Three experiments are established to validate the efficiency and effectiveness of the proposed system. The future research directions may reside in the further improvement of the efficiency of the multiple protection model

in the era of big data and IoT where millions of terminals will be connected.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article will be made available upon request to corresponding authors.

AUTHOR CONTRIBUTIONS

CJ and CH contributed to conception and design of the study. HX organized the database. QH performed the statistical analysis. CJ

wrote the first draft of the manuscript. CJ, HX, CH, and QH wrote sections of the manuscript. All authors contributed to manuscript revision, read, and approved the submitted version.

FUNDING

This research was supported by the Science and Technology Project of State Grid Jiangsu Electric Power Co., Ltd. (Grant No. J2021125), Jiangsu Provincial Social Science Foundation Youth Project (Grant No. 21TQC003), Major Project of Philosophy, and Social Science Research in Universities of Jiangsu Province Education Department (Grant No. 2020SJZDA069).

REFERENCES

- Ahmadzadeh, S., Parr, G., and Zhao, W. (2021). A review on communication aspects of demand response management for future 5G IoT-based smart grids. *IEEE Access*. 9, 77555–77571. doi: 10.1109/ACCESS.2021.3082430
- Aruzuaga, A., Berganza, I., Sendin, A., Sharma, M., and Varadarajan, B. (2010). "PRIME interoperability tests and results from field," in *2010 First IEEE International Conference on Smart Grid Communications* (Gaithersburg, MD), 126–130.
- Asif, M., and Ahmed, J. (2020). A novel case base reasoning and frequent pattern based decision support system for mitigating software risk factors. *IEEE Access*. 8, 102278–102291. doi: 10.1109/ACCESS.2020.2999036
- Borgaonkar, R., and Jaatun, M. G. (2019). "5G as an enabler for secure IoT in the smart grid: invited paper," in *2019 First International Conference on Societal Automation (SA)* (Krakow), 1–7.
- Boyaci, O., Umunnakwe, A., Sahu, A., Narimani, M. R., Ismail, M., Davis, K. R., et al. (2022). Graph neural networks based detection of stealth false data injection attacks in smart grids. *IEEE Systems J* (in press). doi: 10.1109/JSYST.2021.3109082
- Cai, Y., Chen, Q., and Zhang, W. (2020). "Credit and risk management of electricity transaction: a real case based on Guangdong electricity market rules," in *2020 5th Asia Conference on Power and Electrical Engineering (ACPEE)* (Chengdu), 994–999.
- Cao, X., Hu, S., Zhang, Y., Lin, Q., Tang, Z., and Zhang, C. (2019a). Design and implementation of power universal security access zone based on dual isolation. *Electr. Power Eng. Technol.* 38, 152–158. doi: 10.19464/j.cnki.cn32-1541/tm.2019.02.024
- Cao, X., Zhang, Y., Song, L., Hu, S., Tang, Z., and Zhang, C. (2019b). Design and implementation of forward isolation device based on deep packet inspection and security enhancement. *Autom. Electr. Power Syst.* 43, 162–167.
- Chen, B., Qiao, S., Zhao, J., Liu, D., Shi, X., Lyu, M., et al. (2021). A security awareness and protection system for 5G smart healthcare based on zero-trust architecture. *IEEE Inter. Things J.* 8, 10248–10263. doi: 10.1109/JIOT.2020.3041042
- Chen, X., Wen, X., Wang, L., and Lu, Z. (2018). The architecture design of cooperated deployment for multi-access edge computing in 5G. *J. Beijing Univ. Posts Telecommun.* 41, 86–91. doi: 10.13190/j.jbupt.2018-169
- Cheng, L., Xu, D., Zeng, K., Liu, Z., and Zhu, H. (2020). Design and application of power quality terminal information security. *Electric Power Eng. Technol.* 39, 26–33. doi: 10.12158/j.2096-3203.2020.06.005
- Chourib, I., Guillard, G., Mestiri, M., Solaiman, B., and Farah, I. R. (2020). "Case-based reasoning: problems and importance of similarity measure," in *2020 5th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)* (Sousse), 1–6.
- Deng, S., Cai, Q., Zhang, Z., and Wu, X. (2022). User behavior analysis based on stacked autoencoder and clustering in complex power grid environment. *IEEE Trans. Intellig. Transport. Syst.* (in press). doi: 10.1109/TITS.2021.3076607
- Deng, S., Chen, F., Dong, X., Gao, G., and Wu, X. (2021). Short-term load forecasting by using improved GEP and abnormal load recognition. *ACM Trans. Intern. Technol.* 21, 1–28. doi: 10.1145/3447513
- Forcan, M., Maksimovic, M., Forcan, J., and Jokic, S. (2020). "5G and cloudification to enhance real-time electricity consumption measuring in smart grid," in *2020 28th Telecommunications Forum (TELFOR)* (Belgrade), 1–4.
- Ge, L., Ma, T., Chen, W., Bai, X., and Zhang, S. (2020). A top-level design for time-delay uncertainty analysis of situational awareness in smart distribution network. *Electric Power Engineering Technology* 39: 51–57. doi: 10.12158/j.2096-3203.2020.03.008
- Ghanem, K., Ugwuanyi, S., Asif, R., and Irvine, J. (2021). "Challenges and promises of 5G for smart grid teleprotection applications," in *2021 International Symposium on Networks, Computers and Communications (ISNCC)* (Dubai), 1–7.
- Han, R., Du, Q., Guo, C., Du, X., Wang, Q., and Su, Y. (2019). Study on optimization of special invoice service for value-added tax in service hall of power enterprise. *Power Systems and Big Data* 22(01):35–40.
- Hong, W.-C., Huang, D.-R., Chen, C.-L., and Lee, J.-S. (2020). Towards accurate and efficient classification of power system contingencies and cyber-attacks using recurrent neural networks. *IEEE Access*. 8, 123297–123309. doi: 10.1109/ACCESS.2020.3007609
- Hu, Z., Song, X., Huang, T., Li, X., Zhou, R., Xu, X., et al. (2019). Research on network virtualization scheme and networking algorithm of advanced metering infrastructure for water, electricity, gas, and heat meters. *J. Electr. Inform. Technol.* 41, 588–593. doi: 10.11999/JEIT180396
- Li, H., Xu, Y., Meng, F., Ren, S., Li, H., and Pang, X. (2021). Modeling and analysis of 5G terminal communication channel in substation. *Study Opt. Commun.* 2021, 63–66. doi: 10.13756/j.gtxxy.2021.01.013
- Li, K., Jin, X., Kuai, W., Liu, C., and Yang, Y. (2020). "The customized 5G secondary authentication scheme combined with security protection strategy for electrical automation system," in *2020 IEEE 9th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)* (Chongqing), 757–761.
- Li, Y., Lu, J., Xu, Z., Gong, G., and Liao, B. (2018). Design of terminal communication access architecture for smart power distribution and utilization based on integration of multiple technologies. *Autom. Electr. Power Syst.* 42, 169–175. doi: 10.7500/AEPS20170506002
- Liu, R., Hai, X., Du, S., Zeng, L., Bai, J., and Liu, J. (2021). "Application of 5G network slicing technology in smart grid," in *2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)* (Nanchang), 740–743.
- Liu, T., Ma, Y., Jiang, H., Wu, S., Zuo, J., and Peng, T. (2021). "Research on power grid security protection architecture based on zero trust," *Electric Power Inform. Commun. Technol.* 19, 25–32. doi: 10.16543/j.2095-641x.electric.power.ict.2021.07.004
- Liu, Z., Li, Q., Yan, B., and Shang, K. (2020). Application of depth neural network algorithm with stacked sparse auto-encoder in rolling bearing fault diagnosis. *Machine Tool Hydraul.* 48, 208–213. doi: 10.3969/j.issn.1001-3881.2020.23.039
- Ma, L., Zhang, N., Kong, X., Zhu, Y., Wang, Y., and Wang, Y. (2021). "5G network slicing technology helps smart grid development," in *2021 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS)* (Shenyang), 64–68.

- Matinkhah, S. M., and Shafik, W. (2019). "Smart grid empowered by 5g technology," in *2019 Smart Grid Conference (SGC)* (Tehran), 1–6.
- Saghezchi, F. B., Mantas, G., Ribeiro, J., Al-Rawi, M., Mumtaz, S., and Rodriguez, J. (2017). "Towards a secure network architecture for smart grids in 5G era," in *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)* (Valencia), 121–126.
- She, R., Zhang, N., Wang, Y., Guo, D., Ma, W., Liu, R., et al. (2021). Research on cognitive radio non-orthogonal multiple access system in 5g communications oriented to ubiquitous power internet of things. *Electr. Power.* 54, 35–45. doi: 10.11930/j.issn.1004-9649.202010099
- Sun, W., Lu, W., Li, Q., Chen, L., Mu, D., and Yuan, X. (2017). WNN-LQE: wavelet-neural-network-based link quality estimation for smart grid WSNs. *IEEE Access.* 5, 12788–12797. doi: 10.1109/ACCESS.2017.2723360
- Wang, W. (2018). *Design and Implementation of Industrial Network Security Isolation and Information Exchange System* (master's thesis). Beijing University of Posts and Telecommunications, Beijing, China.
- Wang, Y., Zhang, Z., Ma, J., and Jin, Q. (2022). KFRNN: an effective false data injection attack detection in smart grid based on kalman filter and recurrent neural network. *IEEE Intern. Things J.* (in press). doi: 10.1109/JIOT.2021.3113900
- Wu, J., Bian, Y., Zhang, Q., and Feng, B. (2020). Wireless quantum power distribution system based on wireless. *Tele-commun. Sci.* 36, 72–79. doi: 10.11959/j.issn.1000-0801.2020031
- Wylde, A. (2021). "Zero trust: never trust, always verify," in *2021 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA)* (Dublin), 1–4.
- Xu, G., Wang, Z., Zang, D., and An, X. (2018). Anomaly detection algorithm of data center network based on LSDB. *J. Comput. Res. Dev.* 55, 815–830. doi: 10.7544/issn1000-1239.2018.20160970
- Zhang, N., Yang, J., Wang, Y., Chen, Q., and Kang, C. (2019). 5G communication for the ubiquitous internet of things in electricity: technical principles and typical applications. *Proc. CSEE.* 39, 4015–4024. doi: 10.13334/j.0258-8013.pcsee.190892
- Zhang, Z. (2021). *Research on 5G Communication Antenna Carrying Scheme of Shared Power Towers* (master's thesis). North China University of Technology, Beijing, China.
- Zhao, F. (2020). Authenticated encryption implementation scheme based on tweakable grouping. *Comput. Eng.* 46, 144–148. doi: 10.19678/j.issn.1000-3428.0054421
- Zhou, Z., Xiang, Y., Xu, H., Wang, Y., and Shi, D. (2022). Unsupervised learning for non-intrusive load monitoring in smart grid based on spiking deep neural network. *J. Modern Power Syst. Clean Energy* (in press). doi: 10.35833/MPCE.2020.000569
- Zhu, Z., Jia, J., Li, Z., Qian, H., and Kang, K. (2020). A low latency random access mechanism for 5G new radio in unlicensed spectrum. *J. Electr. Inform. Technol.* 42, 111–119. doi: 10.11999/JEIT190515

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

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

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



OPEN ACCESS

EDITED BY

Di Wu,
Chongqing Institute of Green and
Intelligent Technology (CAS), China

REVIEWED BY

Bai Sun,
Southwest Jiaotong University, China
Junwei Sun,
Zhengzhou University of Light
Industry, China
Xiaobing Yan,
Hebei University, China

*CORRESPONDENCE

Houpeng Chen
chp6468@mail.sim.ac.cn

RECEIVED 19 May 2022

ACCEPTED 28 June 2022

PUBLISHED 27 July 2022

CITATION

Lv Y, Chen H, Wang Q, Li X, Xie C and
Song Z (2022) Post-silicon
nano-electronic device and its
application in brain-inspired chips.
Front. Neurobot. 16:948386.
doi: 10.3389/fnbot.2022.948386

COPYRIGHT

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

Post-silicon nano-electronic device and its application in brain-inspired chips

Yi Lv^{1,2}, Houpeng Chen^{1,2,3*}, Qian Wang¹, Xi Li¹,
Chenchen Xie^{1,4} and Zhitang Song^{1,2}

¹State Key Laboratory of Functional Materials for Informatics, Laboratory of Nanotechnology, Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai, China, ²University of Chinese Academy of Sciences, Beijing, China, ³Shanghai Technology Development and Entrepreneurship Platform for Neuromorphic and AI SoC, Shanghai, China, ⁴Shanghai Nanotechnology Promotion Center, Shanghai, China

As information technology is moving toward the era of big data, the traditional Von-Neumann architecture shows limitations in performance. The field of computing has already struggled with the latency and bandwidth required to access memory ("the memory wall") and energy dissipation ("the power wall"). These challenging issues, such as "the memory bottleneck," call for significant research investments to develop a new architecture for the next generation of computing systems. Brain-inspired computing is a new computing architecture providing a method of high energy efficiency and high real-time performance for artificial intelligence computing. Brain-inspired neural network system is based on neuron and synapse. The memristive device has been proposed as an artificial synapse for creating neuromorphic computer applications. In this study, post-silicon nano-electronic device and its application in brain-inspired chips are surveyed. First, we introduce the development of neural networks and review the current typical brain-inspired chips, including brain-inspired chips dominated by analog circuit and brain-inspired chips of the full-digital circuit, leading to the design of brain-inspired chips based on post-silicon nano-electronic device. Then, through the analysis of N kinds of post-silicon nano-electronic devices, the research progress of constructing brain-inspired chips using post-silicon nano-electronic device is expounded. Lastly, the future of building brain-inspired chips based on post-silicon nano-electronic device has been prospected.

KEYWORDS

brain-inspired chips, post-silicon nano-electronic device, phase change memory, resistive memory, synapse, neuron

Introduction

With the rapid development of big data, the Internet of Things, 5G communication technology, and deep learning algorithms, the amount of data has increased exponentially. The huge amount of data poses a lot of challenges to the storage, processing, and transfer of data. Despite the continuous improvement of computer performance, due to the sharp increase in the amount of computation, there is still

a difference of nearly 5 orders of magnitude in the Von-Neumann architecture based on the separation of traditional storage and computation compared with the human brain (Schuller et al., 2015). The traditional Von-Neumann system adopts the separate structure of data storage and data processing. For the data communication process between the computing unit and storage unit, the data processing will produce a lot of loss and latency, which forms a “Von-Neumann bottleneck.” This problem is increasingly highlighted by the fact that CPU speed and memory capacity are growing much faster than the data traffic on both parties (Sun K. X. et al., 2021). This performance mismatch between the storage unit and the computing unit leads to a large delay in the reading of data and in the storage process of the data, that is, the “storage wall” problem. In the case of massive data, it is increasingly overwhelmed. Therefore, it is necessary to explore a new memory architecture based on the human brain structure that achieves low-power consumption, low latency, and space-time information processing capabilities to complete the direct communication of information. Figure 1 shows the traditional Von-Neumann architecture and the new brain-inspired chip architecture (Burr et al., 2015; Silver et al., 2016).

Brain-inspired chips, as the name suggests, are chips that simulate the way the brain works, which is based on the human brain neuron structure and the way of human brain perception and cognition. The chip is designed with the human brain neuron structure to improve the computing power and achieve complete anthropomorphism. Brain-inspired chips adopt a new architecture that simulates the synaptic transmission structure of the human brain. Many processors are similar to neurons and the communication system is similar to nerve fibers. The computing of each neuron is carried out locally. On the whole, the neurons work in a distributed manner, that is, the overall tasks are divided and each neuron is only responsible for one part of the computing.

Brain-inspired chips are based on the combination of microelectronics technology and new neuromorphic devices. Compared with traditional chips, it has greater advantages in power consumption and learning ability. Traditionally, computer chips are designed according to the Von-Neumann architecture. Storage and computing are separated in space. Every time the computer operates, it needs to reciprocate in the two areas of CPU and memory, which leads to frequent data exchanging in inefficient processing of massive amounts of information. In addition, when the chip is working, most of the electrical energy will be converted into heat energy, resulting in increased power consumption.

Brain-inspired chips will achieve two breakthroughs compared with traditional computing chips: one is to break through the limitations of the traditional “executor” computing paradigm and it is expected to form a new paradigm of “self-service cognition”; the other is to break through the limitations of traditional computer architecture to

realize parallel data transmission and distributed processing, which will process massive data in real-time with extremely low-power consumption.

The exploration of brain-inspired chips needs to solve the following three main problems: (1) how to deal with the production capacity of flash memory from all over the world far lower than the growth of big data; (2) how to detect useful data in the face of vast big data; (3) how to rely on artificial intelligence to process big data in two directions— digital accelerators and analog neural networks.

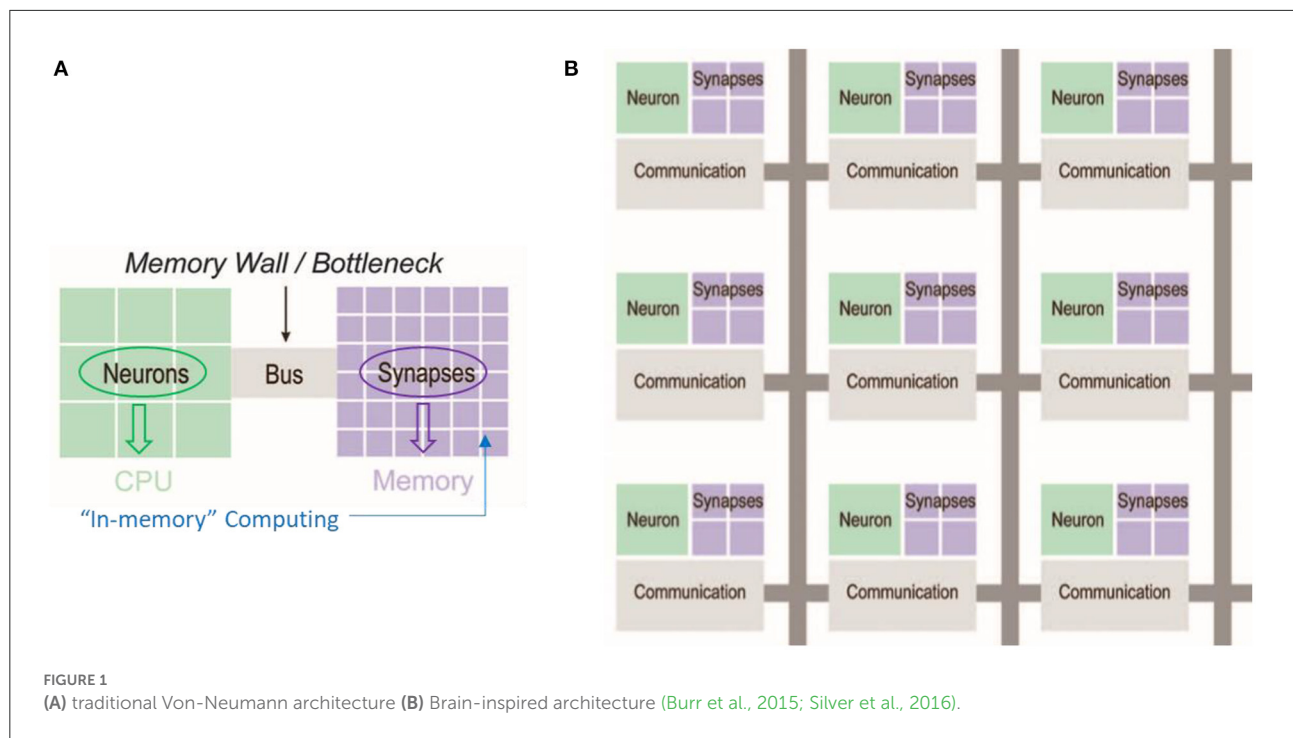
This study first introduces the theory of neural networks and the development of brain-inspired chips. Second, the study focuses on the research progress and application of post-silicon nano-electronic devices. Among them, the application of brain-inspired chips is emphasized. Finally, the research and application prospects of post-silicon nano-electronic device brain-inspired chips have been prospected.

Neural network theory

The basic unit structure of the biological neural network is neuron and synapse. As the connection structure between neurons, the synapse is also the medium of data transmission, as shown in Figure 2A. The three basic functions of neurons are to receive data, integrate data, and transmit data. The typical structure of biological neurons consists of the cell body, dendrite, and axon. In a neuronal system, neurons that send signals are called pre-synaptic neurons. Neurons that receive signals are called post-synaptic neurons. The synaptic structure connects pre-synaptic neurons with post-synaptic neurons which transmit data. The weight of synapses reflects the connection strength between units. One of the cores of the biological neural network is the change of synapses for information transmission efficiency, that is, the plasticity of synaptic connections (Thomas, 2013).

Figure 2B shows the processing of input signals by neurons in a neural network. Neurons not only accept input signals but also need to perform data analysis on the input signals. After being stimulated by other neurons, biological neurons do not simply accumulate all the stimuli and output them to the next neuron. Instead, there is a threshold, and only when the neuron receives a stimulus greater than the threshold will it output a distinct stimulus. Neurons in artificial neural networks also have this function. The artificial neuron accumulates all the input signals processed by the artificial synapse. Artificial neurons only output signals when the cumulative signal exceeds a set threshold.

The neural network mainly includes three layers: the input layer, the output layer, and the hidden layer, in which the hidden layers can be expanded. According to the neuron model, neural networks can be divided into two categories: Artificial Neural



Networks (ANN) (Hopfield, 1982) and Spiking Neural Networks (SNN) (Maass, 1997).

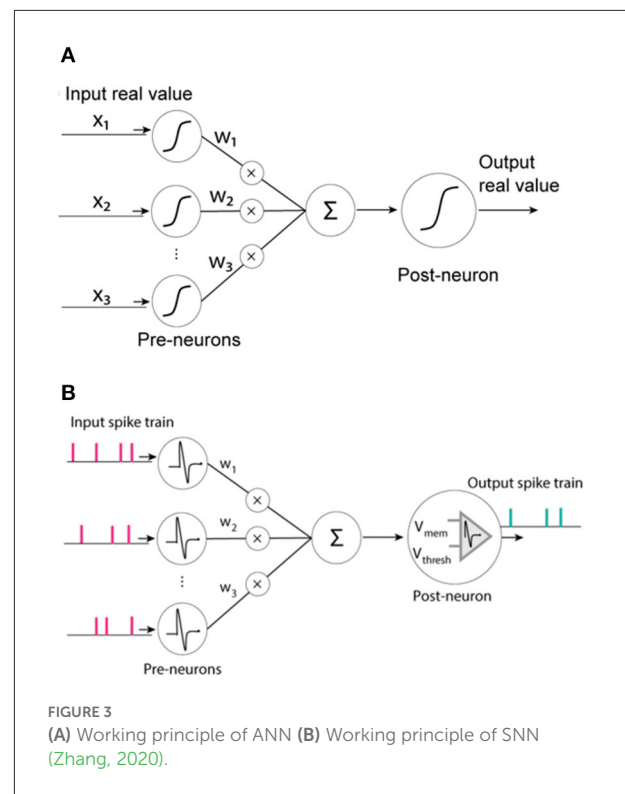
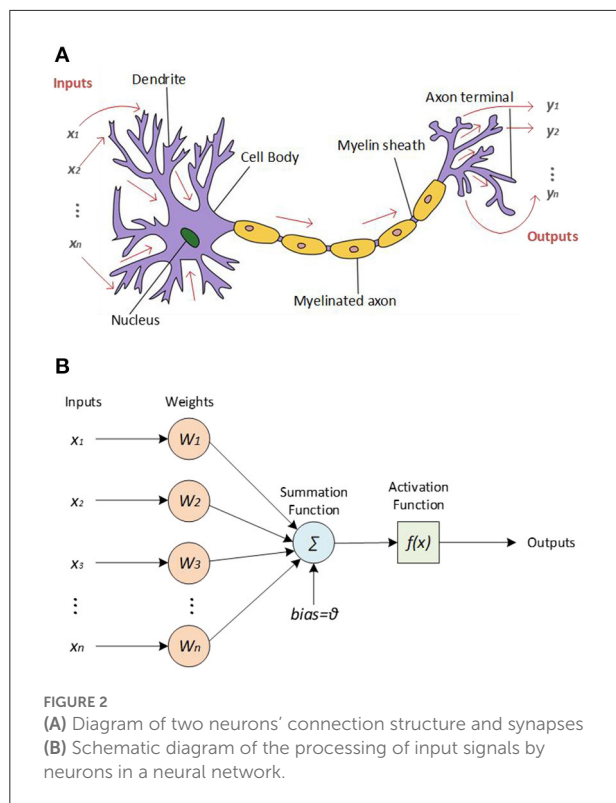
ANN is an information processing system similar to the human brain nervous system which is established inspired by the structure of the biological neural network. The working principle of the ANN is shown in Figure 3A. When the input signal is received, its intensity is first determined, which is commonly referred to as the weighting process. Then, the combined effect of all input signals needs to be determined, that is, the net input, completing the summation process. Finally, the input is transformed through non-linear function calculation to obtain the corresponding output signal. Among them, the functions of non-linear transformation mainly include the sigmoid function, tanh function, and relu function. The unit structure of ANN is similar to that of the biological neural network, which can complete the learning and cognitive training functions of a biological neural network to a certain extent, usually with the Backpropagation (BP) algorithm (Rumelhart et al., 1986). ANN can learn without supervision, that is, it has the ability of self-learning. The advanced function of realizing the associative storage of the human brain can be accomplished by using its feedback network.

SNN is a neural network computing system based on the spiking neuron model. It is a computing model that is closer to the biological neural network. The working principle of SNN is shown in Figure 3B. The pulse signal is discrete, replacing the continuity of the analog signal in ANN. It is similar to ANN. Because the network also

takes the parameters of time information into account, SNN is closer to the biological neuron model. At the same time, the neuron model is also more complicated due to the structure of the pulse signal. From the perspective of the neuron structure in SNN, the input signal will cause the state of the neuron to change, that is, the membrane potential. Only when the membrane potential reaches the threshold potential will the output pulse signal be generated. Among them, Spike timing-dependent plasticity (STDP) algorithm is one of the main learning algorithms of SNN (Fukushima, 1980; Froemke and Dan, 2002).

Brain-inspired chips

At present, brain-inspired chips are mainly divided into brain-inspired chips dominated by analog circuits, brain-inspired chips based on digital circuits, and brain-inspired chips based on post-silicon nano-electronic device. The traditional CMOS technology has been developed to a relatively high degree, and many successful results have been achieved so far. The brain-inspired chip based on a post-silicon nano-electronic device is in the initial stage of exploration and development. At present, the research on brain-inspired chips based on post-silicon nano-electronic device is widely concerned to complete the parallel one-time mapping between input and output. Figure 4 shows the international research status of brain-inspired chips.



Brain-inspired chips dominated by the analog circuit

As early as the end of the twentieth century and the beginning of the twenty-first century, a series of research works on silicon cochlea and silicon neurons laid the foundation for the design of brain-inspired chips dominated by analog circuits. Among them, the most representative is the Neurogrid chip designed by Stanford University in the United States, which has been established to realize the real-time simulation of the biological brain (Benjamin et al., 2014). It uses the SNN neuron model to realize the kinetic calculation of ion channels and fit complex ion channel models. Its system structure is shown in Figure 4A. Each neuron with a size of 256×256 is combined into a neural nucleus, and then 16 neural nuclei are formed into a hierarchical network through a tree topology. Finally, the simulation of a million-level neural network meta-networks is completed.

The BrianScales chip of Heisenberg University in Germany also uses the SNN neuron model to realize the kinetic calculation of ion channels. Its system structure is shown in Figure 4B. A single wafer simulates nearly 200,000 neurons and 49 million synapses. With the cooperation of routing communication circuits, the speed of the entire system is 10,000 times the speed of a biological neural network. However, the power consumption is as high as 1 kW (Davison et al., 2020).

The second generation of BrainScaleS adds online learning capabilities and provides an important reference for completing the real-time learning process.

Brain-inspired chips with full-digital circuit

Because the analog circuit is greatly interfered with by factors such as manufacturing process and environment, the chip does not have advantages in reliability, configurability, scalability, etc. and it is difficult to reproduce the results strictly through simulation, which is not conducive to the research of upper-level algorithms. Therefore, brain-inspired chips based on analog circuits are mainly studied in academia. For the industry, more stable and reliable full-digital circuit brain-inspired chips are preferred (Rast et al., 2010; Benjamin et al., 2014; Merolla et al., 2014; Davies et al., 2018; Davison et al., 2020).

In 2006, the University of Manchester started to develop the SpiNNaker chip, as shown in Figure 4C. The current version is to build an electronic model of the biological brain through 1 million microprocessors from ARM, which can reach 1% of the human brain, achieving the world's first low-power, large-scale digital model of the human brain (Rast et al., 2010), providing a high-performance platform for real-time simulation of large-scale neural networks. The TrueNorth chip released by IBM in

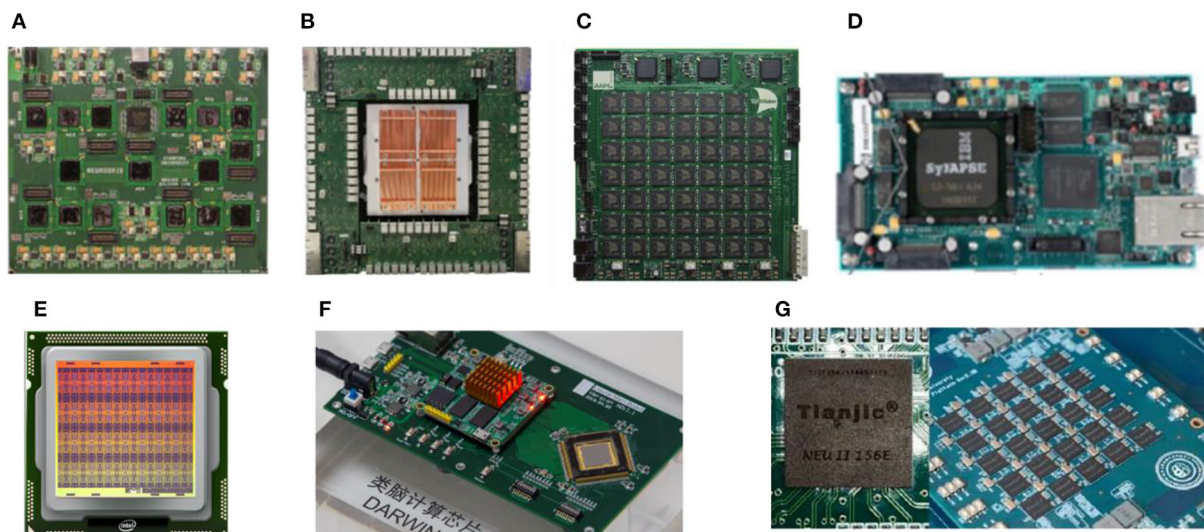


FIGURE 4

(A) The architecture of Neurogrid (B) The architecture of BranScaleS (C) The architecture of SpiNNaker (D) The architecture of TrueNorth (E) The architecture of Loihi (F) The architecture of Darwin (G) The architecture of Tianjic.

TABLE 1 Prevalent brain-inspired chips.

Name	Type	Learning	Simulation time	Capacity	Connection
Neurogrid	Analog-dominated	No	Real-time	256*256 CMOS	USB via FX2
BrainScales	Analog-dominated	No	Slower than real-time	180 K neurons	Ethernet
SpiNNaker	Full-digital	No	Real-time	1% of brain capacity	Ethernet
TrueNorth	Full-digital	No	Faster than real-time	4,096 core per chip	AXI bus to SoC
Loihi	Full-digital	Yes	Faster than real-time	4,096 core per chip	Ethernet, USB
Darwin	Full-digital	No	70 MHz Clock	2,048 neurons per chip	UART to USB
Tianjic	Full-digital	Ni	Real-time	40 k neurons per chip	Not specified

2014 adopts a full-digital circuit, simulating the connection of 1 million neurons and 256 million synapses to complete the neural network function, as shown in Figure 4D, with a very low-power consumption of 73 mW (Merolla et al., 2014). The function of the chip is to perform inference on pre-trained networks, which can be applied to object detection in images. The Loihi chip released by Intel in 2017 contains 128,000 neurons and 128 million synaptic structures, which realizes the complexity of neural network topology and enables on-chip learning with different learning modes (Davies et al., 2018) as shown in Figure 4E. Loihi 2 was released in 2021, which is an upgraded version of Loihi using a new process. It integrates 1 million neurons, but compared with the first generation, the area is reduced by half, and the processing speed is 10 times that of the first generation.

In 2019, Zhejiang University released a new brain-inspired chip, Darwin II, as shown in Figure 4F (Shen et al., 2015). This chip uses a 55 nm process, and the number of neurons

in the entire chip reaches 150,000. Through the cascade of chip systems, a brain-inspired computing system with tens of millions of neurons can be constructed. Tsinghua University released a new artificial intelligence chip Tianjic III (Tianjic) in 2019, as shown in Figure 4G (Pei et al., 2019). The chip adopts multi-core architecture, reconfigurable building blocks, simplified data flow, and hybrid coding. It can not only adapt to machine learning algorithms based on computer science but also easily realize brain-inspired circuits and multiple encodings. Table 1 introduces prevalent brain-inspired chips.

Brain-inspired chips based on post-silicon nano-electronic device

With the continuous development of Moore's Law, the feature size of transistors is getting closer and closer to

their theoretical physical limit. It is difficult to improve the development of the current CMOS process integration technology further. When a brain-inspired chip is integrated on a large scale, the larger the area of the circuit is, the higher the power consumption generated. At the same time, transistors have defects in simulating the dynamic characteristics of neurons and synapses, and their ability to simulate brain-inspired computing needs to be further improved. Therefore, researchers turned their attention to post-silicon nano-electronic devices to realize the design of brain-inspired chips.

The key of brain-inspired chips - post-silicon nano-electronic device

It is urgent to find a memory, whose working behavior characteristics are similar to those of the brain. Brain-inspired chips consist of a large amount of memory. For a long time, researchers have been looking for and constructing suitable post-silicon nano-electronic devices with memory functions. For example, memristive devices can change the working state of the device through different working mechanisms, which is similar to the role of ion channels contained in the membranes of neurons and synapses in the brain. Some memristive devices can keep working like this all the time. Even if the power is turned off, they will not be lost, just like human memory.

Semiconductor memory can be divided into two categories according to the characteristics of stored information: volatile memory (VM) and non-volatile memory (NVM). Generally speaking, volatile memory means that when the system is powered off—all data stored in the device will be automatically lost. It mainly includes two types: Dynamic Random-Access Memory (DRAM) and Static Random-Access Memory (SRAM).

Non-volatile memory means that when the system is powered off, the data stored in the device will always be retained and will not be lost. It mainly includes new memory and flash memory (Nor Flash memory and Nand Flash memory). [Figure 5](#) shows the main distribution of semiconductor memories on the market today.

In terms of data reading and writing speed, the speed of volatile memory is usually very fast. However, in general, the writing latency of non-volatile memory is high. When the number of writes reaches a certain number, the storage of data will fail because the memory will reach its storage limit. Of course, for an ideal memory, it should have both non-volatile characteristics of data and access speed comparable to SRAM, and no read and write restrictions within a certain range.

Post-silicon nano-electronic device designs and mainstream silicon CMOS processes have different new materials and storage mechanisms. These materials mainly

include chalcogenides compounds, transition metal oxides, carbon materials, ferroelectrics, and ferromagnetic metals. Different from the traditional electronic process switching mechanism, they are realized using phase transition, molecular restructuring, quantum mechanical phenomena, and ion reaction. Most non-volatile memories are based on two-terminal switching devices, which are commonly used in high-density memory architectures such as crossbars. In recent years, new storage technologies represented by phase-change random-access memory (PCRAM), resistance random-access memory (RRAM), magnetic random-access memory (MRAM), and ferroelectric random-access memory (FeRAM) have emerged in the field of vision of researchers.

Compared to CMOS technology, which is widely used in chips, post-silicon nano-electronic device-based brain-inspired chips have greater potential in terms of computational density, power efficiency, computational accuracy, and learning ability. In addition, the size of the post-silicon nano-electronic device can be reduced to <2 nm with ultra-high-density integration ([Pi et al., 2019](#)). Therefore, post-silicon nano-electronic device technology will be applied to the large-scale manufacturing of brain-inspired chips in the future.

The performance requirements of post-silicon nano-electronic device-based brain-inspired chips largely depend on their specific applications. [Figure 6A](#) shows the performance requirements for various application scenarios including storage, inference, learning, and typical non-volatile memory. The number of simulated states ([Figure 6B](#)) determines the accuracy of weight matching between synapses, and the formation of larger neural networks requires at least 8 resistance states that can be accurately distinguished ([Jacob et al., 2017](#)). By optimizing device material selection and circuit design, the current post-silicon nano-electronic device chips can achieve up to 256 resistance states. The dynamic range of switching state transitions is defined as the on/off ratio ([Figure 6C](#)) ([Wang et al., 2016](#)), which determines the ability to assign the weights in the algorithm to the device conductivity, which in most cases differs from the conductivity of the device in relation to the threshold switch with two resistors. Compared to the high switching ratio, the switching ratio of the multi-resistor post-silicon nano-electronic device is <10 . The linearity ([Figure 6D](#)) refers to the linearity of the relationship between the conductivity of the device and the number of exciting electric pulses. During the formation of the post-silicon nano-electronic device, the device weights show increasing and decreasing asymmetry ([Figure 6E](#)). In the training process, the conductivity update of post-silicon nano-electronic device is usually in the partial scope of the conductivity window, instead of the full range ([Figure 6F](#)). After tuning the post-silicon nano-electronic device to different conductance levels, the conductance of the device may change over time, and the two levels may overlap after a period of time

(Figure 6G). Failed devices refer to post-silicon nano-electronic device that cannot be tuned to the target conductance Level. (Figure 6H). Based on this, it can be seen that post-silicon nano-electronic device can store weights. According to different application requirements, a suitable new type of post-silicon nano-electronic device can be selected as a memristive device for neural network design (Wang et al., 2022). The memristive device can simulate the function of biological synapses because the sandwich structure of the device unit is similar to nerve synapses (Sun B. et al., 2021c).

Phase-change memory (PCRAM)

PCRAM is a post-silicon nano-electronic device based on GST materials such as $\text{Ge}_2\text{Sb}_2\text{Te}_5$. According to different device characteristics, the composition of GST material can be further adjusted, as shown in Figure 7C. The resistance change characteristic of PCRAM is shown in Figure 7D. For example, Ge-rich GST (N-type doping) can be used in high-temperature automotive applications for better data retention (Cheng et al., 2012). The switching resistance ratio of phase-change memory is much larger than that of STT-MRAM (in the range of 100 to 1,000 times). Therefore, in principle, Multilevel Cell (MLC) operation is feasible (4 bit/cell has been proposed; Nirschl et al., 2007). A major challenge in PCRAM cell design is the need for a relatively large write current when melting the phase-change material. At present, the structure design trend of phase-change memory is from mushroom type to confined type. The limited type reduces the write current by limiting heat dissipation. Extremely scaled phase-change memory cells using carbon tube electrodes have shown that write currents can reach $1\ \mu\text{A}$ at the 2 nm node (Liang et al., 2011). The resistance drift caused by amorphous relaxation limits the data retention ability of PCRAM, especially for MLC. Therefore, complex circuit compensation schemes are needed. PCRAM has good process compatibility with silicon CMOS technology, regarded as the most mature process technology in the post-silicon nano-electronic device industry (Yu and Chen, 2016).

Spin-transfer-torque magnetic random-access memory (STT-RAM)

Spin-transfer-torque magnetic random-access memory (STT-RAM) is a kind of memory that stores data by changing the resistance through the magnetoresistance effect of magnetic materials. The basic unit of STT-RAM is a sandwich structure composed of an insulating barrier layer sandwiched between two magneto-resistive materials, which is called a magnetic tunnel junction (MTJ). At the bottom is the fixed layer with fixed polarity, and at the top is the free layer with changeable polarity. The magnetic moment of the free layer is written

under the action of the current of the upper and lower wires at the same time. When the magnetic moments of the fixed magnetic layer and the free magnetic layer are parallel in the same direction, the resistance of the magnetic tunnel junction is small. At this time, the device shows a low-resistance state. When the magnetic moments of the fixed magnetic layer and the free magnetic layer are parallel in the opposite direction, electrons are not easy to pass through the magnetic tunnel junction, and the MTJ structure shows a high resistance state, as shown in Figure 7E. The resistance-voltage characteristic of STT-RAM is shown in Figure 7F. STT-RAM stores data “0” and “1” through two different resistive states.

Resistive random-access memory (RRAM)

RRAM is a kind of post-silicon nano-electronic device that can realize the reversible conversion between high-resistance and low-resistance states under the action of an external electric field based on the resistance of non-conductive material, thus completing the storage of binary data, as shown in Figure 7A. The current-voltage characteristic of RRAM is shown in Figure 7B. According to the different conductive media, it can be divided into two categories: OxRAM (Oxide-RAM), which conducts with oxygen holes, and CBRAM (Conductive Bridge RAM), which conducts with metal ions. The write operation of RRAM includes unipolar and bipolar modes, depending on the oxide as well as the electrode material system. The unipolar mode generally requires larger write currents and has poorer endurance; therefore, the bipolar mode is preferred. A key challenge in the design of the RRAM cell structure is the variability of switching parameters. The significant variation in resistance distribution (perhaps one or two orders of magnitude) presents a challenge to the design of sensitive readout circuits, requiring write-verify techniques to program to the target state, which may at the same time cause delays in MLC operation. RRAM typically has superior process compatibility with mainstream silicon CMOS technologies.

Ferroelectric random-access memory (FeRAM)

Ferroelectric memory is a post-silicon nano-electronic device with a special process, which is formed by using synthetic lead zirconium titanium (PZT) materials to form memory crystals, as shown in Figure 7G. The polarization-voltage hysteretic characteristic of FeRAM is shown in Figure 7H. When an electric field is applied to a ferrotransistor, the central atom follows the electric field and stops at the low-energy state I. Conversely, when a reverse electric field is applied to the same ferrotransistor, the central atom moves in the crystal along

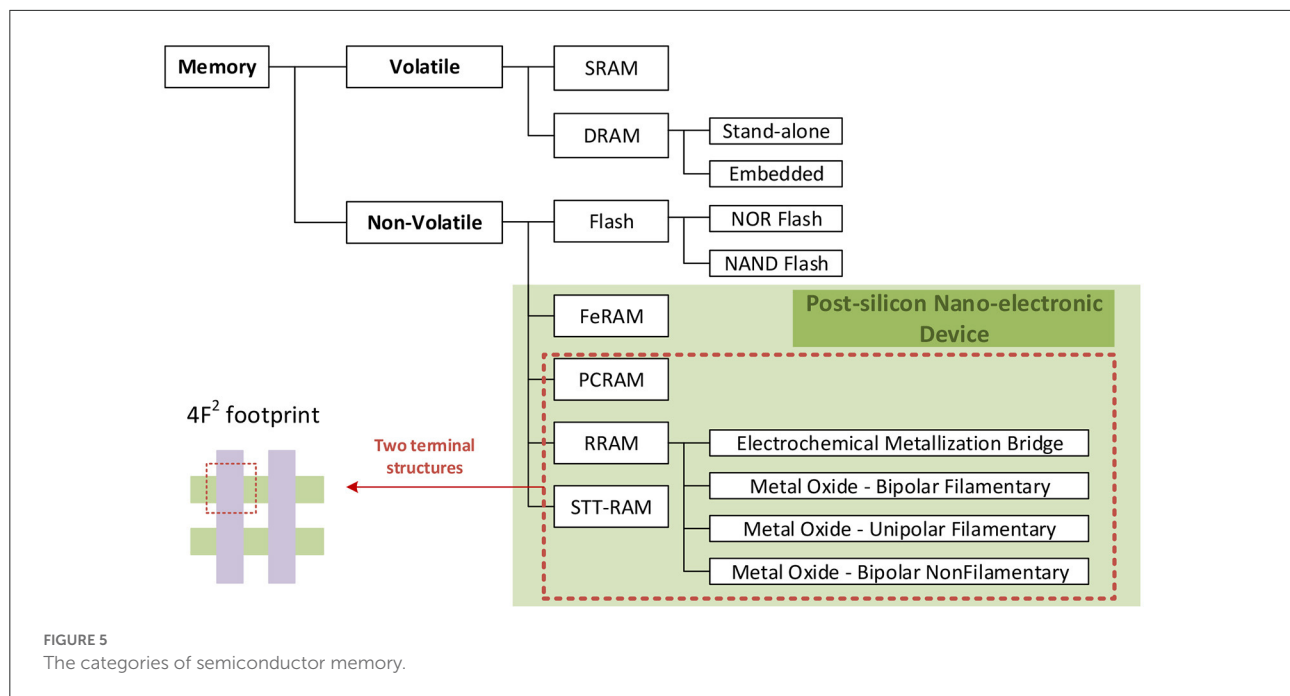


TABLE 2 The performance comparison of post-silicon nano-electronic device (Lai and Lowrey, 2001; Song et al., 2008; Sheu et al., 2009; Kim et al., 2011; Tamura et al., 2011; Bez and Cappelletti, 2012; Bez et al., 2013; Zangeneh and Joshi, 2014; Roy et al., 2020; Saxena, 2020).

Devices	MRAM	FeRAM	PCRAM	RRAM
Non-volatile	Yes	Yes	Yes	Yes
Cell size (F^2)	8	15–34	4	4
Read latency	30 ns	45 ns	50 ns	8.5 ns
Write/Erase latency	30 ns/30 ns	10 ns/10 ns	10 ns/20 ns	5 ns/5 ns
Endurance	$>10^{12}$	10^{14}	$>10^{12}$	10^8
Write power	High	Low	High	Low
High voltage required (V)	3	2–3	1.5–3	1.5–3
CMOS compatibility	Medium	Medium	Good	Good
Multi-level	No	No	Yes	Yes
3D Xpoint	Yes	Yes	Yes	Yes
Cost	Medium	High	Low	Low

the direction of the electric field and stops in another low-energy state II. A large number of central atoms move and the couples in the crystal unit cell form ferroelectric domains, and the ferroelectric domains form polarized charges under the action of an electric field. The polarization charge formed by the reversal of the ferroelectric domain under the electric field is higher, and the polarization charge formed by the ferroelectric domain without reversal under the electric field is lower. FeRAM combines the advantages of RAM and ROM. Compared with traditional non-volatile memory, FeRAM has the characteristics of high speed, low-power consumption, and long life.

Comparison of major post-silicon nano-electronic device

The above four major emerging trends are summarized as key strengths and challenges of post-silicon nano-electronic device. PCRAM, RRAM, and MRAM are called resistive memory, while FeRAM is a new memory equivalent to charge memory. Table 2 shows the performance comparison of post-silicon nano-electronic device (Lai and Lowrey, 2001; Song et al., 2008; Sheu et al., 2009; Kim et al., 2011; Tamura et al., 2011; Bez and Cappelletti, 2012; Bez et al., 2013; Zangeneh and Joshi, 2014; Roy et al., 2020; Saxena, 2020). From the table, we can conclude that phase-change memory shows great advantages in terms of high read and write speed, high-density integration, low-energy consumption, low cost, and compatibility with CMOS processes. It can replace the current co-storage structure of DRAM and Flash memory, and its potential in high-speed and high-density storage cannot be underestimated.

Research on construction of brain-inspired chips based on post-silicon nano-electronic device

Synapse

Combined with the design and application of brain-inspired chips, different types of non-volatile memory devices have been proposed. In the application of neural networks, according to the relationship between the adjustment of weight and the reading

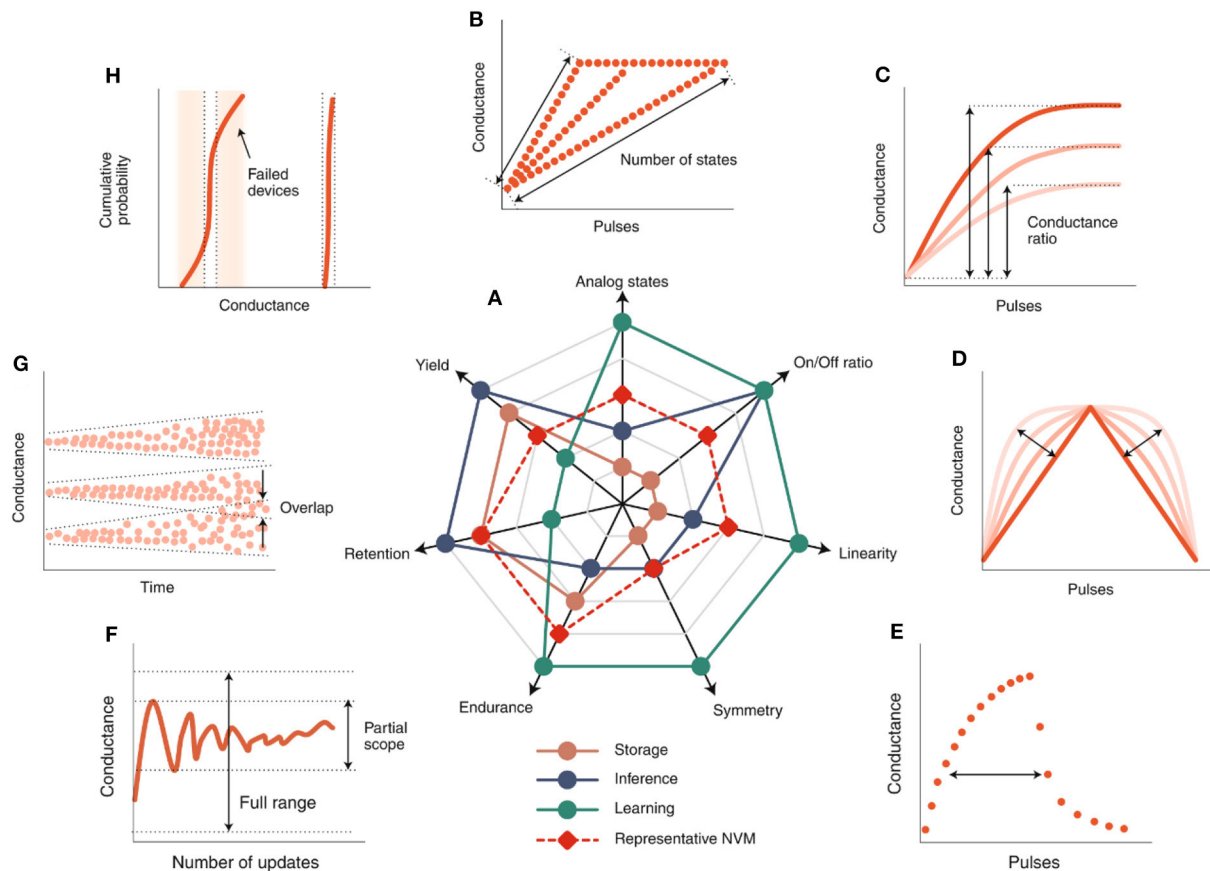


FIGURE 6

Application-dependent device metric requirements (Zhang W. Q. et al., 2020). (A) Ranking of qualitative device requirements for three potential applications and NVM. (B96–H), schematic diagram of computing device requirements: (B) simulation state, (C) on/off ratio, (D) linearity, (E) symmetry, (F) durability, (G) retention rate and (H) yield.

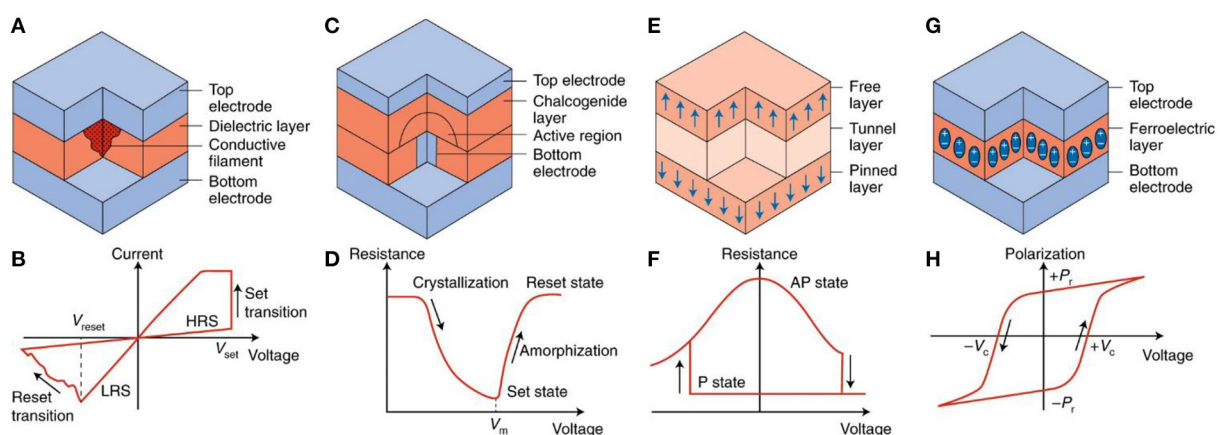
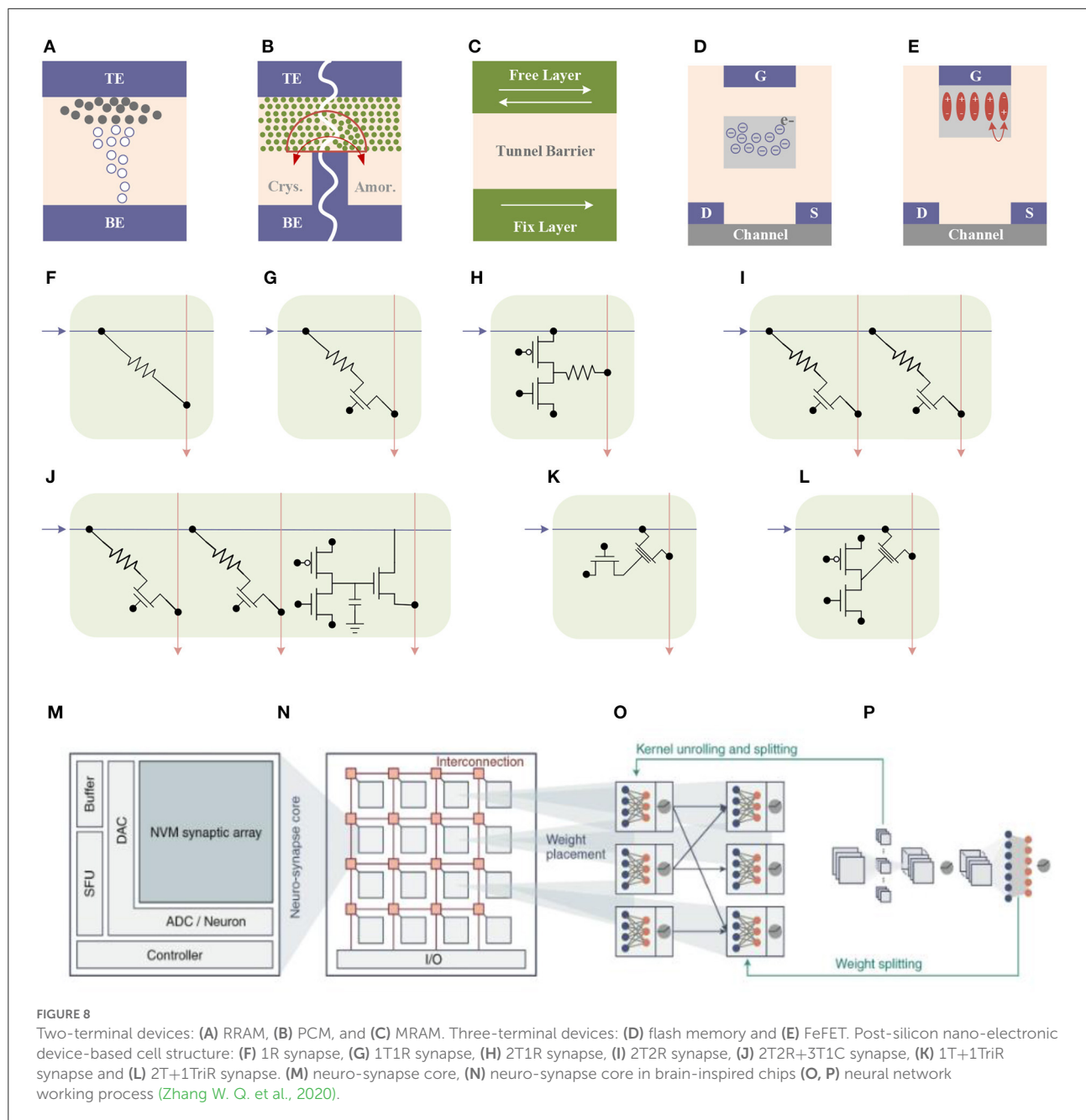


FIGURE 7

Post-silicon nano-electronic device. (A) Conductive filament resistive memory (B) corresponding polar current-voltage characteristics (C) Phase-change memory (D) phase-change memory characteristics (E) Spin-transfer torque magnetic random-access memory (F) resistance-voltage characteristics of Spin-transfer torque magnetic random-access memory (G) ferroelectric random-access memory (H) polarization-voltage hysteresis characteristics (Ielmini and Wong, 2018).



of weight, these devices can be divided into two categories: two-terminal devices and three-terminal devices. The two-terminal devices mainly include PCRAM, RRAM, and MRAM. Three-terminal devices mainly include flash memory and ferroelectric memory as shown in Figure 8.

PCRAM

Work on a PCM-based device was first proposed in 2012 (Kuzum et al., 2012). By applying a series of incremental excitation pulses to the device, the resistance of the device can

change under about 100 resistance states, and under appropriate pulses, the learning rule of spiking-time-dependent plasticity (STDP) can be realized under waveform. Subsequently, different research groups proposed various excitation pulse programming schemes to reduce the complexity and power consumption of PCM-based neuromorphic circuits (Suri et al., 2011; Jackson et al., 2013; Li et al., 2013; Stefano et al., 2016). However, a major challenge of PCM devices is the asymmetry of the resistance switching process, which is mainly because the process of melting the material at a high temperature to form an amorphous state is more difficult to control than the process of

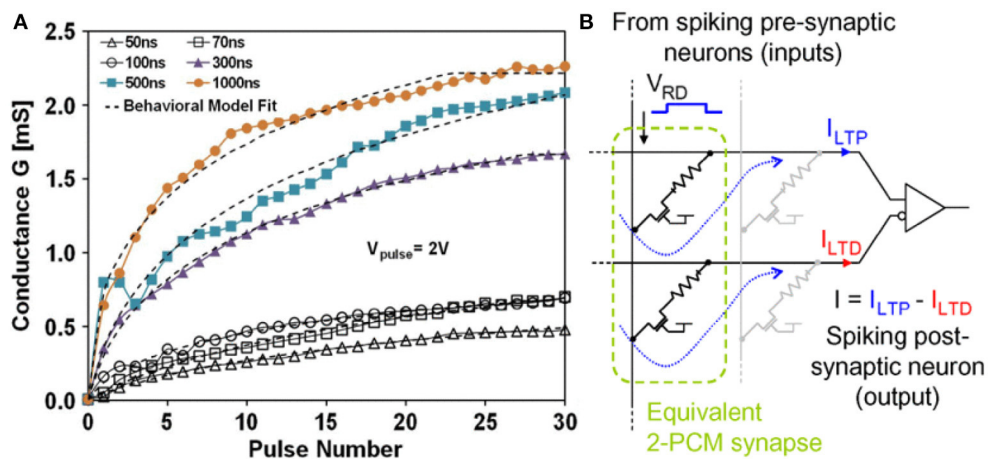


FIGURE 9 (A) Experimental LTP characteristics of $\text{Ge}_2\text{Sb}_2\text{Te}_5$ (GST) PCM devices. (B) 2-PCM synapse principle (Bichler et al., 2012).

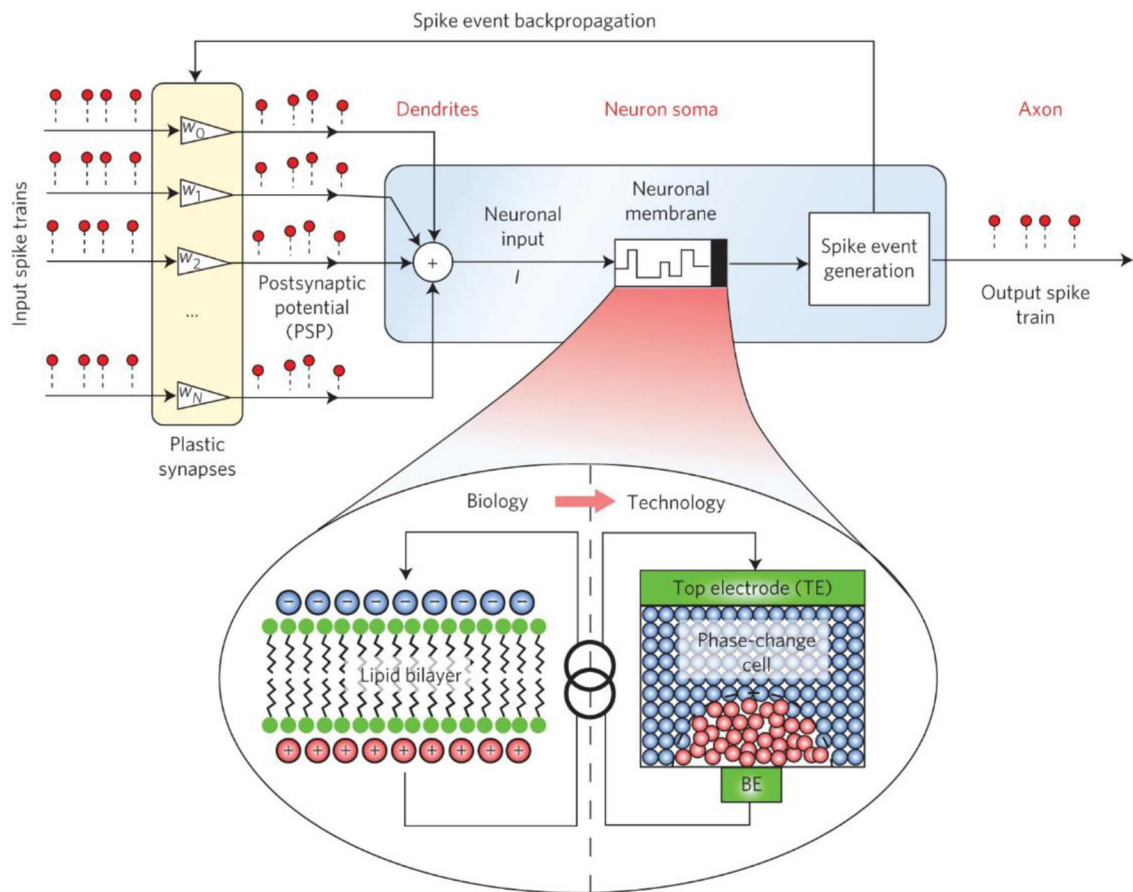


FIGURE 10 Basic structure diagram of IBM phase-change neuron (Tuma et al., 2016).

recrystallization of its amorphous state. Phase-change memory can achieve multilevel resistance states by the programming pulse. Only two resistance states can be achieved during reset using the same pulse. To this end, Bichler et al. proposed a 2-PCM synapse design to deal with this problem in their work (Bichler et al., 2012), in which one PCM was used as a synaptic potentiation (Long-term potentiation, LTP), and the other was used as a synaptic depression (Long-term depression, LTD). In this design, both PCM devices are partially crystallized. During LTP and LTD, the conductance of the device is increasing. The current through the LTP device plays a positive role and the current through the LTD device plays a negative role. The current through the LTD is subtracted at the output, ultimately resulting in synaptic inhibition, as shown in Figure 9.

RRAM

In the early RRAM device design, the artificial synapse device based on HfO_x material adopted the one-way reset learning mode (Yu et al., 2013). To make this process smoother, multiple conductive filaments can be formed under the electric field through the design of multilayer oxides implemented in the device. In the RRAM device with an interface mechanism, the resistance changes during the set and reset process are relatively gentle (Park et al., 2012, 2013; Gao et al., 2015b; Wang et al., 2015). In addition, multi-resistance states can also be achieved by regulating the capture and release of interfacial oxygen vacancies (Yang et al., 2017). The resistive switching device exhibited multistate resistance behavior, which enables 2-bit storage capacity in a single device providing a method for logic in-memory and neuromorphic computing (Sun B. et al., 2021b). A memristive device and a hybrid system composed of CMOS neurons and RRAM synapses were experimentally demonstrated to realize essential synaptic functions such as STDP (Jo et al., 2010).

Depending on the application, different excitation pulse programming schemes are applied for online or offline training with RRAM, so the requirements for device characteristics may vary. For example, in the offline training process, the resistance state can be iteratively programmed into the specified target layer by the write-verify method. Since the programming process is one-time, accuracy is more critical than speed in the writing process. Alibart et al. simulated this programming process by firing a series of pulses (Alibart et al., 2011), where pulses with smaller amplitudes approach the state in smaller steps but take longer than pulses with larger amplitudes. Therefore, the use of a pulse train of variable amplitude can approach the desired state in small steps within a reasonable time frame. In the absence of a change in switching state, the pulse amplitude becomes progressively smaller, resulting in smaller steps as the device gets closer to the desired state. However, due to the fluctuation of the device itself, the process of determining the initial pulse value often starts with a small non-disturbing pulse

and gradually increases, and the conductance of the device is confirmed by applying the read pulse after the write pulse until the required accuracy is achieved. When using this method, because the initial state is very close to the desired state, the maximum amplitude of the voltage pulse written in the new sequence is smaller than that of the previous sequence, which can ensure that the device is closer to the desired state. For a single $\text{Pt/TiO}_{2-x}/\text{Pt}$ device, this method can adjust the conductance to any expected value in the dynamic range of the device with an error of only 1% (Alibart et al., 2011). For the Ag/a-Si/Pt single device, the tuning accuracy for the low-resistance state is also close to 1%. A similar iterative algorithm has also been demonstrated in HfO_x devices (Gao et al., 2015a). For online training, since the synaptic weights need to be dynamically trained, the programming speed becomes a more important factor, therefore, smooth conductance adjustment without write verification becomes the preferred solution (Yu, 2018). Some examples of state-of-the-art based on RRAM are given in the literature, all of which show bidirectional graded conductance tuning under the same programming voltage pulse (Mulaosmanovic et al., 2017; Yu, 2018). Although these devices can all reach tens or hundreds of resistive states, there are still non-linearities and asymmetries in the tuning. They used $\text{W/MgO/SiO}_2/\text{Mo}$ memristive device as the synapse of speech recognition and completed the hardware implementation of SNN using the improved supervised tempotron algorithm on the TIDIGITS dataset (Al-Shedivat et al., 2015; Wu et al., 2022).

FeFET

FeFET synapse devices use a three-terminal structure, which is characterized by decoupling the write and read paths for the resistive state of the device. In FeFET, the programming voltage applied to the gate determines the resistance change of the device. The current is given by the drain-source current read. As mentioned earlier, as a three-terminal device, FeFET is designed for weighted summation as pseudo cross arrays. In terms of physical structure, FeFET is to apply short voltage pulses through the gate through the multi-domain effect in ferroelectric materials, so as to gradually adjust the capacitance of the gate, and finally complete the adjustment of threshold voltage and channel conductance (Oh et al., 2017). Recently, (Jerry et al., 2017) simulated FeFET synaptic devices using a gate-last manufacturing process flow of n-channel FeFETs, whose gates were formed by stacking 10 nm $\text{Hf}_{0.5}\text{Zr}_{0.5}\text{O}_2$ (HZO) materials by atomic deposition and annealed at 600°C to generate multiple ferroelectric domains in HZO nanocrystals. Compared to RRAM devices, FeFETs have advantages in on-off ratio and available program pulse range with less variation in the weight update curve.

Neuron

Neuromorphic computing systems need to simulate not only synapses, but also neuronal dynamics, including membrane potential maintenance, transient dynamics, and neurotransmission processes (Burr et al., 2016). In human neurons, the maintenance of membrane potential depends on the ion pump and ion channel in the middle of the membrane lipid bilayer. The excitation or inhibition of post-synaptic potentials of neuronal dendrites can change their state. In neurons composed of phase-change memory, the membrane potential is represented by an amorphous state of high resistance, and the firing frequency of phase-change neurons is controlled by the amplitude width and time interval of a series of voltage pulses. Connecting the plasticity of synapses, such neurons can complete complex calculations such as detecting time correlation in parallel data streams.

When a post-silicon nano-electronic device is used to build a neuron, the goal of the device is not the continuity of its conductance state, but rather a cumulative behavior that fires after receiving a certain number of pulses. Since each conductance state of a post-silicon nano-electronic device affects its behavior between accumulation and emission pulses, changes in these conductance states will be the focus of research.

The use of PCM devices to construct neurons was first reported in the work of Ovshinsky and Wright (Wright et al., 2011). In their work, Tuma et al. changed the membrane potential of neural components through phase encoding, and then experimentally proved that neurons based on PCM devices can integrate post-synaptic input signals (Tuma et al., 2016). A system in which both neuron and synaptic devices were implemented using PCM devices was reported by Pantazi et al. (2016). Studies by Averbeck et al. have shown that stochastic behaviors in neuronal dynamics, such as ionic conductance noise and thermal noise-induced chaotic motion of charge carriers, morphological variation between neurons, and other background noise can also affect neuronal signaling. Encoding and transmission play a key role (Averbeck et al., 2006). Therefore, simulating these random behaviors in artificial neurons can achieve many interesting functions (Maass, 2014). The random behavior in the device is due to the inhomogeneity of the thickness of the amorphous region and the internal atomic configuration during melt quenching of different batches of materials, and these random behaviors can lead to multiple integrations of the signal generated by a phase transition in the PCM neuron. The interval is generated between the transmitted signals to facilitate some statistical calculations based on these transmitted signals. At the same time, however, the melt quenching process of PCM device materials, especially the elemental migration therein, limits the device's durability. Likewise, in RRAM devices, large changes in conductance can also result in reduced device durability. Therefore, extending the lifetime of the device requires ensuring that neurons accumulate

and fire the number of spiking signals or fabricating the device with high-durability materials.

Figure 10 (Tuma et al., 2016) shows the basic structure diagram of the IBM phase-change neuron. The synapses consist of phase-change units that are responsible for weighting incoming excitation signals. Multiple excitation signals are input into the synaptic array, and after the signals pass through the synapse, they are input into the phase-change unit that functions as a neuronal membrane (neuronal membrane, which can also be understood as a neuron). When the threshold is reached, the IF event is triggered, and the excitation signal is emitted. The excitation signal is firstly conducted to the outside for further data processing, and at the same time, it is back-propagated for comparison with the previous input excitation signal. For positive delays, synaptic conductance is increased, and for negative delays, synaptic conductance is decreased. These functions of synapses can be achieved with SET and RESET operations. Through the above analysis, it can be found that this system has met the main requirements of the bionic neural network.

Al-Shedivat et al. have proposed to use TiO_x -based RRAM to construct random artificial neurons (Al-Shedivat et al., 2015). In an RRAM, integrating the input signal of neurons increases the voltage across the capacitive device, that is, increases the membrane potential of neurons, causing the device as a whole to switch to the low-resistance state and the generated increased current is converted into digital by an external circuit signal or analog pulse. Meanwhile, random switching of resistive states in RRAM results in random firing of neurons (Nessler et al., 2013). Jang et al. also implemented a similar principle on a $\text{Cu/Ti/Al}_2\text{O}_3$ -based conductive bridge random-access memory (conductive-bridging RAM; CBRAM) (Jang et al., 2017).

Resistive memory has also been used in the simulation of axonal behavior. The neuron resistor (Neuristor) was first proposed as an analog device for the Hodgkin-Huxley axon (Hodgkin and Huxley, 1952; Crane, 1962), but it could not be mass-produced in the early stages of the concept. Pickett et al. fabricated a neuron resistor composed of two nanoscale Mott memristors based on the Joule heat-driven insulation-conductor phase transition principle (Pickett et al., 2013). This neuron utilizes the dynamic resistance switching behavior of the Mott memristor and the functional similarity between Na^+ and K^+ channels in the Hodgkin-Huxley model to make the resistor have all-or-nothing pulse signal gain, periodicity, etc. important neuron features.

Many research works provide more references for the practical application of memristor. A. Chandrasekar et al. studied impulsive synchronization of stochastic memristor-based recurrent neural networks with time delay and concluded that the memristive connection weights have a certain relationship with the stability of the system (Chandrasekar and Rakkiyappan, 2016). Researchers have also done a lot of research on the complete definition of the brain elicitation

system and learning mode. The definition of completeness for brain-inspired systems was put forward by Zhang et al. (Zhang Y. et al., 2020), which is composed of Turing-complete software abstract model and a versatile abstract brain-inspired architecture, providing convenience for ensuring the portability of programming language, the completeness of hardware and the feasibility of compilation. By introducing a brain-inspired meta-learning paradigm and a differentiable spike model combining neuronal dynamics and synaptic plasticity, Wu et al. proposed a brain-inspired global-local cooperative learning model. It achieves higher performance than a single learning method (Wu et al., 2020). Associative memory is an important mechanism to describe the process of biological learning and forgetting. It is of great significance to construct neural morphological computing systems and simulate brain-inspired functions. The design and implementation of associative memory circuits have become a research hotspot in the field of artificial neural networks. Pavlov's conditioned reflex experiment is one of the classical cases of associative memory. The implementation of its hardware circuit still has some problems, such as complex circuit design, imperfect function, and unclear process description. Based on this, researchers combined the classical conditional reflection theory and nanoscience and technology to study its circuit. Sun et al. put forward a memristive neural network circuit that can realize Pavlov associative memory with time delay achieving learning, forgetting, fast learning, slow forgetting, and time-delay learning (Sun et al., 2020). A memristor-based learning circuit that can realize Pavlov associative memory with dual-mode switching, auditory mode, and visual mode, was designed and verified by Sun et al. (2021a). Sun et al. proposed a memristor-based neural network circuit of emotion congruent memory, which considers various memory and emotion functions, achieving the functions of learning, forgetting, changing speed, and emotion generation (Sun et al., 2021b). Gao et al. experimentally demonstrated the *in situ* learning ability of the sound localization function in a 1K analog memristor array with the proposed multi-threshold-update scheme (Gao et al., 2022), representing a significant advance toward memristor-based auditory localization system with low-energy consumption and high performance.

In 2016, Sengupta et al. proposed a deep spiking neural system based on magnetic tunnel junction (MTJ), which lead to a fully trained deep neural network (DNN) transformed into an SNN on forwarding inference (Sengupta et al., 2016). The input signal of DNN is encoded as a Poisson spike sequence of SNN according to the rate and is regulated by the synaptic weights, resulting in a post-synaptic current flowing through heavy metals under the MTJ device, which causes the switching of the device state in the MTJ device, the probability of which is the distribution is approximated by the DNN sigmoid function, again with a 50% probability of zero input by adding a constant bias current. Stochastic micromagnetic simulations of large-scale deep learning neural network architectures show

that SNN forward inference can achieve a test accuracy of up to 97.6% on the MNIST handwritten digit database. Sharad et al. also suggested using lateral spin valves and domain wall magnets (DWMs) as neural components to achieve multiply-accumulate functions (Roy et al., 2013). Initially conceived, this work connects two input magnets with opposite polarities, a stationary magnet, and an output magnet through a metal channel. The transmission of spin torque makes the output magnet switch to a flexible axis parallel to the polarity of the input magnet, which is detected by MTJ.

In a later envision, the device instead uses two magnets with fixed and opposite polarities, which are connected through a DWM device with an integrated MTJ. One magnet is grounded and the other is used to receive the difference between the excitatory and inhibitory currents plus the bias current to center the response of DWM. Such current differences determine the direction of the current flowing through the DWM and the resulting magnetic polarity, which is then induced by the MTJ. Sharad et al. also proposed circuit integration schemes of unipolar and bipolar neurons, as well as device-circuit joint simulation of some common image processing applications. Moon et al. realized pattern recognition neuromorphic systems by combining Mo/PCMO synaptic devices with NbO₂ insulator-metal transition neuronal devices, in which the Mo/PCMO devices exhibited excellent performance due to their high activation energy during oxidation reliability (Moon et al., 2015).

Conclusion

The development of artificial intelligence is highly dependent on massive amounts of data. Meeting the data processing requirements of high-performance machine learning is the most important factor for brain-inspired chips.

This study summarizes the development of brain-inspired and post-silicon nano-electronic device and its applications in brain-inspired chips. The current representative post-silicon nano-electronic device artificial synaptic devices include PCM, RRAM, and FeRAM. In addition, the post-silicon nano-electronic device can also be used to construct neural components. As CMOS technology is approaching its physical limits, post-silicon nano-electronic device-based brain-inspired chips offer a promising path forward.

The brain-inspired system has a broad application prospect in the field of artificial intelligence and cognitive computing because of its low-power consumption and fast parallel computing speed (Sun B. et al., 2021a). The research on brain-inspired chips has made phased progress, but there is still no intelligent system that can approach the human level. In the next period, the research on brain-inspired chips will focus on enhancing the universality of neural computing circuit modules, as well as reducing the difficulty of design and manufacturing. In addition, there is an urgent need to solve the power consumption

problem of brain-inspired computing chips, such as exploring ultra-low-power materials and computing structures, to lay a foundation for further improving the performance of brain-inspired chips.

Future device research should focus on implementing simulated post-silicon nano-electronic device with improved performance and exploring more bio-trustworthy properties.

1. Post-silicon nano-electronic device represented by phase-change memory is continuously optimized. In the future, they will continue to improve device performance, develop large-scale integration technology, and realize heterogeneous integration and three-dimensional high-density integration of various neuromorphic devices.

2. Small-scale brain-inspired chip circuits continue to improve in terms of synaptic structure and neuron function. In the future, the collaborative design will be opened to develop large-scale scalable, and versatile post-silicon nano-electronic device-based brain-inspired chips to realize massive data processing.

3. SNN still lacks effective learning algorithms, lacks dedicated hardware platforms, and has few commercial products, which only have theoretical advantages. The research space is relatively large, and the realization of learning algorithms and hardware has broad research prospects.

Brain-inspired chips have propelled the development of brain-inspired supercomputers, giving them extreme computing speeds and massive data processing capabilities. In the future, they can also “cognition” and “thinking,” which will change the traditional working mode of computers.

Author contributions

YL, HC, and ZS brought up the core concept and architecture of this manuscript. YL, HC, QW, XL, CX, and ZS wrote the article. All authors contributed to the article and approved the submitted version.

References

- Alibart, F., Gao, L., Hoskins, B. D., and Strukov, D. B. (2011). High-precision tuning of state for memristive devices by adaptable variation-tolerant algorithm. *Nanotechnology* 23:075201. doi: 10.1088/0957-4484/23/7/075201
- Al-Shedivat, M., Naoos, R., Neftci, E., Cauwenberghs, G., and Salama, K. N. (2015). “Inherently stochastic spiking neurons for probabilistic neural computation,” in *2015 7th International IEEE/Embs Conference on Neural Engineering (NER)*, 356–359 (Montpellier).
- Averbeck, B. B., Latham, P. E., and Pouget, A. (2006). Neural correlations, population coding and computation. *Nat. Rev. Neurosci.* 7, 358–366. doi: 10.1038/nrn1888
- Benjamin, B. V., Gao, P., McQuinn, E., Chou D Hary, S., Chandrasekaran, A. R., Bussat, J., et al. (2014). “Neurogrid: a mixed-analog-digital multichip system for large-scale neural simulations,” in *Proceedings of the IEEE* 102, 699–716.
- Bez, R., and Cappelletti, P. (2012). “Emerging memory technology perspective,” in *Proceedings of Technical Program of 2012 VLSI Technology, System and Application, Hsinchu, Taiwan*. doi: 10.1109/VLSI-TSA.2012.6210106
- Bez, R., Cappelletti, P., Servalli, G., and Pirovano, A. (2013). “Phase change memories have taken the field,” in *Memory Workshop* (Monterey, CA). doi: 10.1109/IMW.2013.6582084
- Bichler, O., Suri, M., Querlioz, D., Vuillaume, D., DeSalvo, B., and Gamrat, C. (2012). Visual pattern extraction using energy-efficient “2-PCM synapse” neuromorphic architecture. *IEEE Trans. Electron Dev.* 59, 2206–2214. doi: 10.1109/TED.2012.2197951
- Burr, G. W., Narayanan, P., Shelby, R. M., Sidler, S., and Leblebici, Y. (2015). “Large-scale neural networks implemented with non-volatile memory as the synaptic weight element: comparative performance analysis

Funding

This work was supported by the National Natural Science Foundation of China (92164302, 61874129, 91964204, 61904186, 61904189, 61874178), 0Strategic Priority Research Program of the Chinese Academy of Sciences (XDB44010200), Science and Technology Council of Shanghai (17DZ2291300, 19JC1416801, 2050112300), by the Youth Innovation Promotion Association CAS under Grant 2022233 and in part by the Shanghai Research and Innovation Functional Program under Grant 17DZ2260900.

Acknowledgments

This work is done in the State Key Laboratory of Functional Materials for Informatics, Laboratory of Nanotechnology, Shanghai Institute of Microsystem and Information Technology, and Chinese Academy of Sciences. The authors express their thanks for the help provided by the lab.

Conflict of interest

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

Publisher's note

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

(accuracy, speed, and power),” in *IEEE International Electron Devices Meeting* (Washington, DC).

Burr, G. W., Shelby, R. M., Sebastian, A., Kim, S., Kim, S., Sidler, S., et al. (2016). Neuromorphic computing using non-volatile memory. *Adv. Phys. X* 2, 89–124. doi: 10.1080/23746149.2016.1259585

Chandrasekar, A., and Rakkiyappan, R. (2016). Impulsive controller design for exponential synchronization of delayed stochastic memristor-based recurrent neural networks. *Neurocomputing* 173, 1348–1355. doi: 10.1016/j.neucom.2015.08.088

Cheng, H. Y., Wu, J. Y., Cheek, R., Raoux, S., Brightsky, M., Garbin, D., et al. (2012). “A thermally robust phase change memory by engineering the Ge/N concentration in (Ge, N)xSbTe z phase change material,” in *2012 International Electron Devices Meeting* (San Francisco, CA).

Crane, H. D. (1962). Neuristor - a novel device and system concept. *Proc. Inst. Radio Eng.* 50, 2048–2060. doi: 10.1109/JRPROC.1962.288234

Davies, M., Srinivasa, N., Lin, T. H., Chinya, G., Cao, Y. Q., Choday, S. H., et al. (2018). Loihi: a neuromorphic manycore processor with on-chip learning. *IEEE Micro* 38, 82–99. doi: 10.1109/MM.2018.112130359

Davison, A. P., Müller, E., Schmitt, S., Vogginger, B., Lester, D., Pfeil, T., et al. (2020). *HBP Neuromorphic Computing Platform Guidebook*. Available online at: <https://www.humanbrainproject.eu/en/silicon-brains/how-we-work/hardware/> (accessed June 7, 2022).

Froemke, R. C., and Dan, Y. (2002). Spike-timing-dependent synaptic modification induced by natural spike trains. *Nature* 416, 433–438. doi: 10.1038/416433a

Fukushima, K. (1980). Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biol. Cybern.* 36, 193–202. doi: 10.1007/BF00344251

Gao, B., Zhou, Y., Zhang, Q., Zhang, S., Yao, P., Xi, Y., et al. (2022). Memristor-based analogue computing for brain-inspired sound localization with *in situ* training. *Nat. Commun.* 13, 1–8. doi: 10.1038/s41467-022-29712-8

Gao, L., Chen, P. Y., and Yu, S. (2015a). Programming protocol optimization for analog weight tuning in resistive memories. *IEEE Electr. Dev. Lett.* 36, 1157–1159. doi: 10.1109/LED.2015.2481819

Gao, L., Wang, L.-T., Chen, P.-Y., Sarma, V., and Seo, J.-S. (2015b). Fully parallel write/read in resistive synaptic array for accelerating on-chip learning. *Nanotechnology* 26, 455204–455204. doi: 10.1088/0957-4484/26/45/455204

Hodgkin, A. L., and Huxley, A. F. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. Physiol.* 117, 500–544. doi: 10.1113/jphysiol.1952.sp004764

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. U.S.A.* 79, 2554–2558. doi: 10.1073/pnas.79.8.2554

Ielmini, D., and Wong, H. S. P. (2018). In-memory computing with resistive switching devices. *Nat. Electr.* 1, 333–343. doi: 10.1038/s41928-018-0092-2

Jackson, B. L., Rajendran, B., Corrado, G. S., Breitwisch, M., Burr, G. W., Cheek, R., et al. (2013). Nanoscale electronic synapses using phase change devices. *ACM J. Emerg. Technol. Comput. Syst.* 9, 1–20. doi: 10.1145/2463585.2463588

Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., et al. (2017). “Quantization and training of neural networks for efficient integer-arithmetic-only inference,” in *Proceedings of the IEEE Conference On Computer Vision and Pattern Recognition* (Salt Lake City, UT). doi: 10.48550/arXiv.1712.05877

Jang, W. J., Lee, M. K., Yoo, J., Kim, E., Yang, D. Y., Park, J., et al. (2017). Low-resistive high-work-function gate electrode for transparent a-IGZO TFTs. *IEEE Trans. Electron Devices* 64, 164–169. doi: 10.1109/TED.2016.2631567

Jerry, M., Chen, P. Y., Zhang, J., Sharma, P. and Datta, S. (2017). “Ferroelectric FET analog synapse for acceleration of deep neural network training,” in *2017 IEEE International Electron Devices Meeting (IEDM)* (San Francisco, CA).

Jo, S. H., Chang, T., Ebong, I., Bhadviya, B. B., Mazumder, P., and Lu, W. (2010). Nanoscale memristor device as synapse in neuromorphic systems. *Nano Lett.* 10, 1297–1301. doi: 10.1021/nl904092h

Kim, B., Song, Y. J., Ahn, S., Kang, Y., Jeong, H., Ahn, D., et al. (2011). “Current status and future prospect of Phase Change Memory,” in *IEEE, Current status and future prospect of Phase Change Memory* (Xiamen). doi: 10.1109/ASICON.2011.6157176

Kuzum, D., Jeyasingh, R. G., Lee, B., and Wong, H. S. (2012). Nanoelectronic programmable synapses based on phase change materials for brain-inspired computing. *Nano Lett.* 12, 2179–2186. doi: 10.1021/nl201040y

Lai, S., and Lowrey, T. (2001). “OUM - A 180 nm nonvolatile memory cell element technology for stand alone and embedded applications,” in *International*

Electron Devices Meeting. Technical Digest (Cat. No.01CH37224) (Washington, DC: IEEE).

Li, Y., Zhong, Y., Xu, L., Zhang, J., Xu, X., Sun, H., et al. (2013). Ultrafast synaptic events in a chalcogenide memristor. *Sci. Rep.* 3, 1–7. doi: 10.1038/srep01619

Liang, J., Jeyasingh, R., Chen, H. Y., and Wong, H. (2011). “A 1.4μA reset current phase change memory cell with integrated carbon nanotube electrodes for cross-point memory application,” in *Digest of Technical Papers - Symposium on VLSI Technology* (Kyoto), 100–101.

Maass, W. (1997). Networks of spiking neurons: the third generation of neural network models. *Neural Netw.* 10, 1659–1671. doi: 10.1016/S0893-6080(97)00011-7

Maass, W. (2014). Noise as a resource for computation and learning in networks of spiking neurons. *Proc. IEEE* 102, 860–880. doi: 10.1109/JPROC.2014.2310593

Merolla, P. A., Arthur, J. V., Alvarez-Icaza, R., Cassidy, A. S., Sawada, J., Akopyan, F., et al. (2014). A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science* 345, 668–673. doi: 10.1126/science.1254642

Moon, K., Cha, E., Park, J., Gi, S., and Hwang, H. (2015). “High density neuromorphic system with Mo/Pr0.7Ca0.3MnO3 synapse and NbO2 IMT oscillator neuron,” in *2015 IEEE International Electron Devices Meeting (IEDM)* (Washington, DC). doi: 10.1109/IEDM.2015.7409721

Mulaosmanovic, H., Ocker, J., Muller, S., Noack, M., and Slesazek, S. (2017). “Novel ferroelectric FET based synapse for neuromorphic systems,” in *2017 Symposium on VLSI Technology* (Kyoto). doi: 10.23919/VLSIT.2017.7998165

Nessler, B., Pfeiffer, M., Buesing, L., and Maass, W. (2013). Bayesian computation emerges in generic cortical microcircuits through spike-timing-dependent plasticity. *PLoS Comput. Biol.* 9:e1003037. doi: 10.1371/journal.pcbi.1003037

Nirschl, T., Philipp, J. B., Happ, T. D., Burr, G. W., Rajendran, B., Lee, M. H., et al. (2007). “Write strategies for 2 and 4-bit Multi-Level Phase-Change Memory,” in *2007 IEEE International Electron Devices Meeting* (Washington, DC: IEEE).

Oh, S., Kim, T., Kwak, M., Song, J., Woo, J., Jeon, S., et al. (2017). HfZrOx-based ferroelectric synapse device with 32 levels of conductance states for neuromorphic applications. *IEEE Electr. Dev. Lett.* 38, 732–735. doi: 10.1109/LED.2017.2698083

Pantazi, A., Wozniak, S., Tuma, T., and Eleftheriou, E. (2016). All-memristive neuromorphic computing with level-tuned neurons. *Nanotechnology* 27:355205. doi: 10.1088/0957-4484/27/35/355205

Park, S., Kim, H., Choo, M., Noh, J., and Hwang, H. (2012). RRAM-based synapse for neuromorphic system with pattern recognition function. *Electron Devices Meeting* doi: 10.1109/IEDM.2012.6479016

Park, S., Sheri, A., Kim, J., Noh, J., and Hwang, H. (2013). “Neuromorphic speech systems using advanced ReRAM-based synapse,” in *Electron Devices Meeting* (San Francisco, CA).

Pei, J., Deng, L., Song, S., Zhao, M., Zhang, Y., Wu, S., et al. (2019). Towards artificial general intelligence with hybrid Tianjic chip architecture. *Nature* 572, 106–111. doi: 10.1038/s41586-019-1424-8

Pi, S., Li, C., Jiang, H., Xia, W., Xin, H., Yang, J. J., et al. (2019). Memristor crossbar arrays with 6-nm half-pitch and 2-nm critical dimension. *Nat. Nanotechnol.* 14, 35–39. doi: 10.1038/s41565-018-0302-0

Pickett, M. D., Medeiros-Ribeiro, G., and Williams, R. S. (2013). A scalable neuristor built with Mott memristors. *Nat. Mater.* 12, 114–117. doi: 10.1038/nmat3510

Rast, A., Galluppi, F., Xin, J., and Furber, S. (2010). “The Leaky Integrate-and-Fire neuron: a platform for synaptic model exploration on the SpiNNaker chip,” in *International Joint Conference on Neural Networks* (Barcelona). doi: 10.1109/IJCNN.2010.5596364

Roy, K., Chakraborty, I., Ali, M., Ankit, A., and Agrawal, A. (2020). “In-memory computing in emerging memory technologies for machine learning: an overview,” in *2020 57th ACM/IEEE Design Automation Conference (DAC)* (San Francisco, CA).

Roy, K., Sharad, M., Fan, D. L., and Yogendra, K. (2013). “Beyond charge-based computation: boolean and non-boolean computing with spin torque devices,” in *2013 IEEE International Symposium on Low Power Electronics and Design (ISLPED)* (Beijing), 139–142.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back propagating errors. *Nature* 323, 533–536. doi: 10.1038/323533a0

Saxena, V. (2020). Mixed-signal neuromorphic computing circuits using hybrid cmos-tram integration. *IEEE Trans. Circuits Syst. II Express Briefs* 68, 581–586. doi: 10.1109/TCSII.2020.3048034

- Schuller, I. K., Stevens, R., Pino, R., and Pechan, M. (2015). *Neuromorphic Computing – From Materials Research to Systems Architecture Roundtable*. USDOE Office of Science (SC) (United States).
- Sengupta, A., Parsa, M., Han, B., and Roy, K. (2016). Probabilistic deep spiking neural systems enabled by magnetic tunnel junction. *IEEE Trans. Electron Devices* 63, 2963–2970. doi: 10.1109/TED.2016.2568762
- Shen, J., Ma, D., Gu, Z., Zhang, M., and Pan, G. (2015). Darwin: a neuromorphic hardware co-processor based on Spiking Neural Networks. *Science China Inform. Sci.* 59, 1–5. doi: 10.1007/s11432-015-5511-7
- Sheu, S. S., Chiang, P. C., Lin, W. P., Lee, H. Y., and Tsai, M. J. (2009). “A 5ns fast write multi-level non-volatile 1 K bits RRAM memory with advance write scheme,” in *2009 Symposium on VLSI Circuits* (Kyoto).
- Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., et al. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature* 529, 484–489. doi: 10.1038/nature16961
- Song, Z. T., Liu, B., and Feng, S. L. (2008). Development of nano phase change storage technology. *J. Funct. Mater. Dev.* 14:14. Available online at: https://www.researchgate.net/publication/288703901_Development_of_nano_phase_change_storage_technology
- Stefano, A., Nicola, C., Mario, L., Valerio, M., Agostino, P., Paolo, F., et al. (2016). Unsupervised learning by spike timing dependent plasticity in Phase Change Memory (PCM) synapses. *Front. Neurosci.* 10:56. doi: 10.3389/fnins.2016.00056
- Sun, B., Guo, T., Zhou, G., Ranjan, S., and Wu, Y. A. (2021a). Synaptic devices based neuromorphic computing applications in artificial intelligence. *Mater. Today Phys.* 18:100393. doi: 10.1016/j.mtphys.2021.100393
- Sun, B., Ranjan, S., Zhou, G., Guo, T., and Wu, Y. A. (2021b). Multistate resistive switching behaviors for neuromorphic computing in memristor. *Materials Today Adv.* 9:100125. doi: 10.1016/j.mtadv.2020.100125
- Sun, B., Zhou, G., Sun, L., Zhao, H., Chen, Y., Yang, F., et al. (2021c). ABO 3 multiferroic perovskite materials for memristive memory and neuromorphic computing. *Nanoscale Horizons* 6:939. doi: 10.1039/D1NH00292A
- Sun, J., Han, G., Zeng, Z., and Wang, Y. (2020). Memristor-based neural network circuit of full-function pavlov associative memory with time delay and variable learning rate. *IEEE Trans. Cybern.* 50, 2935–2945. doi: 10.1109/TCYB.2019.2951520
- Sun, J., Han, J., Liu, P., and Wang, Y. (2021a). Memristor-based neural network circuit of pavlov associative memory with dual mode switching. *AEU Int. J. Electr. Commun.* 129:153552. doi: 10.1016/j.aeue.2020.153552
- Sun, J., Han, J., Wang, Y., and Liu, P. (2021b). “Memristor-based neural network circuit of emotion congruent memory with mental fatigue and emotion inhibition,” in *IEEE Transactions on Biomedical Circuits and Systems*, 15, 606–616.
- Sun, K. X., Chen, J. S., and Yan, X. B. (2021). The future of memristors: materials engineering and neural networks. *Adv. Funct. Mater.* 31:2006773. doi: 10.1002/adfm.202006773
- Suri, M., Bichler, O., Querlioz, D., Cueto, O., and Desalvo, B. (2011). “Phase change memory as synapse for ultra-dense neuromorphic systems: application to complex visual pattern extraction,” in *2011 IEEE International Electron Devices Meeting (IEDM)* (Washington, DC).
- Tamura, S., Yamanaka, N., Saito, T., Takano, I., and Yokoyama, M. (2011). “Electrically switchable graphene photo-sensor using phase-change gate filter for non-volatile data storage application with high-speed data writing and access,” in *2011 International Electron Devices Meeting* (Washington, DC).
- Thomas, A. (2013). Memristor-based neural networks. *J. Phys. D Appl. Phys.* 46:093001. doi: 10.1088/0022-3727/46/9/093001
- Tuma, T., Pantazi, A., Le Gallo, M., Sebastian, A., and Eleftheriou, E. (2016). Stochastic phase-change neurons. *Nat. Nanotechnol.* 11, 693–699. doi: 10.1038/nnano.2016.70
- Wang, I. T., Lin, Y. C., Wang, Y. F., Hsu, C. W., and Hou, T. H. (2015). “3D synaptic architecture with ultralow sub-10 fJ energy per spike for neuromorphic computation,” in *IEEE International Electron Devices Meeting* (San Francisco, CA).
- Wang, J., Mao, S., Zhu, S., Hou, W., Yang, F., and Sun, B. (2022). Biomemristors-based synaptic devices for artificial intelligence applications. *Org. Electr.* 106:106540. doi: 10.1016/j.orgel.2022.106540
- Wang, Z., Yin, M., Zhang, T., Cai, Y., Wang, Y., Yang, Y., et al. (2016). Engineering incremental resistive switching in TaOx based memristors for brain-inspired computing. *Nanoscale* 8, 14015–14022. doi: 10.1039/C6NR00476H
- Wright, C. D., Liu, Y. W., Kohary, K. I., Aziz, M. M., and Hicken, R. J. (2011). Arithmetic and biologically-inspired computing using phase-change materials. *Adv. Mater.* 23, 3408–3413. doi: 10.1002/adma.201101060
- Wu, X. L., Dang, B. J., Wang, H., Wu, X. L., and Yang, Y. C. (2022). Spike-enabled audio learning in multilevel synaptic memristor array-based spiking neural network. *Adv. Intelligent Syst.* 4:2100151. doi: 10.1002/aisy.202100151
- Wu, Y., Zhao, R., Zhu, J., Chen, F., Xu, M., Li, G., et al. (2020). Brain-inspired global-local learning incorporated with neuromorphic computing. *Nat. Commun.* 13, 1–14. doi: 10.1038/s41467-021-27653-2
- Yang, Z., Yi, L., Wang, X., and Friedman, E. G. (2017). “Synaptic characteristics of Ag/AgInSbTe/Ta-based memristor for pattern recognition applications,” in *IEEE Transactions on Electron Devices*, 64, 1–6. doi: 10.1109/TED.2017.2671433
- Yu, S. (2018). “Neuro-inspired computing with emerging nonvolatile memories,” in *Proceedings of the IEEE*, 106, 260–285. doi: 10.1109/JPROC.2018.2790840
- Yu, S., and Chen, P. Y. (2016). Emerging memory technologies: recent trends and prospects. *IEEE Solid State Circuits Mag.* 8, 43–56. doi: 10.1109/MSSC.2016.2546199
- Yu, S., Gao, B., Fang, Z., Yu, H., Kang, J., and Wong, H. S. P. (2013). A low energy oxide-based electronic synaptic device for neuromorphic visual systems with tolerance to device variation. *Adv. Mater.* 25, 1774–1779. doi: 10.1002/adma.201203680
- Zangeneh, M., and Joshi, A. (2014). Design and optimization of nonvolatile multibit 1T1R resistive RAM. *IEEE Trans. Very Large Scale Integr. Syst.* 22, 1815–1828. doi: 10.1109/TVLSI.2013.2277715
- Zhang, W. Q., Gao, B., Tang, J. S., Yao, P., Yu, S. M., Chang, M. F., et al. (2020). Neuro-inspired computing chips. *Nat. Electr.* 3, 371–382. doi: 10.1038/s41928-020-0435-7
- Zhang, X. (2020). *Research on the Neuromorphic Computing and System Applications With Memristors*. Chinese Academy of Sciences.
- Zhang, Y., Qu, P., Ji, Y., Zhang, W., Gao, G., Wang, G., et al. (2020). A system hierarchy for brain-inspired computing. *Nature* 586, 378–384. doi: 10.1038/s41586-020-2782-y



OPEN ACCESS

EDITED BY

Song Deng,
Nanjing University of Posts
and Telecommunications, China

REVIEWED BY

Zhuo Yan,
Shenyang Aerospace University, China
Fengmin Yu,
Chongqing University of Posts
and Telecommunications, China

*CORRESPONDENCE

Lei Feng
fenglei@cigit.ac.cn

†These authors have contributed
equally to this work

RECEIVED 29 June 2022

ACCEPTED 12 August 2022

PUBLISHED 29 August 2022

CITATION

Yan H, Feng L, Yu Y, Liao W, Feng L,
Zhang J, Liu D, Zou Y, Liu C, Qu L and
Zhang X (2022) Cross-site scripting
attack detection based on a modified
convolution neural network.
Front. Comput. Neurosci. 16:981739.
doi: 10.3389/fncom.2022.981739

COPYRIGHT

© 2022 Yan, Feng, Yu, Liao, Feng,
Zhang, Liu, Zou, Liu, Qu and Zhang.
This is an open-access article
distributed under the terms of the
[Creative Commons Attribution License](#)
(CC BY). The use, distribution or
reproduction in other forums is
permitted, provided the original
author(s) and the copyright owner(s)
are credited and that the original
publication in this journal is cited, in
accordance with accepted academic
practice. No use, distribution or
reproduction is permitted which does
not comply with these terms.

Cross-site scripting attack detection based on a modified convolution neural network

Huyong Yan^{1,2,3,4†}, Li Feng^{5†}, You Yu⁶, Weiling Liao⁵,
Lei Feng^{7,8*}, Jingyue Zhang⁴, Dan Liu⁹, Ying Zou⁹,
Chongwen Liu^{1,2,3}, Linfa Qu¹⁰ and Xiaoman Zhang¹⁰

¹Chongqing Engineering Laboratory for Detection Control and Integrated System, Chongqing Technology and Business University, Chongqing, China, ²Chongqing Key Laboratory of Intelligent Perception and Blockchain Technology, Chongqing, China, ³School of Computer Science and Information Engineering, Chongqing Technology and Business University, Chongqing, China, ⁴School of Big Data and Artificial Intelligence, Chongqing Polytechnic Institute, Chongqing, China, ⁵Chongqing Academy of Eco-Environmental Science, Chongqing, China, ⁶Chongqing Ecological Environment Big Data Application Center, Chongqing, China, ⁷Online Monitoring Center of Ecological and Environmental of The Three Gorges Project, Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Chongqing, China, ⁸College of Environment and Ecology, Chongqing University, Chongqing, China, ⁹Chongqing Polytechnic Institute, Chongqing, China, ¹⁰School of Mathematics and Statistics, Chongqing Technology and Business University, Chongqing, China

Cross-site scripting (XSS) attacks are currently one of the most threatening network attack methods. Effectively detecting and intercepting XSS attacks is an important research topic in the network security field. This manuscript proposes a convolutional neural network based on a modified ResNet block and NiN model (MRBN-CNN) to address this problem. The main innovations of this model are to preprocess the URL according to the syntax and semantic characteristics of XSS attack script encoding, improve the ResNet residual module, extract features from three different angles, and replace the full connection layer in combination with the 1*1 convolution characteristics. Compared with the traditional machine learning and deep learning detection models, it is found that this model has better performance and convergence time. In addition, the proposed method has a detection rate compared to a baseline of approximately 75% of up to 99.23% accuracy, 99.94 precision, and a 98.53% recall value.

KEYWORDS

XSS, URL, ResNet, word vector, code injection

Introduction

The worldwide web has become the most common, least expensive and fastest communication medium in the world today (Cao et al., 2021; Kotzur, 2022). Tens of millions of people are using it for their daily activities due to its convenient access and variety of available services. Social networking sites, online shopping sites,

and cloud storage services are becoming increasingly popular. In this case, a typical feature that attracts internet customers is a user-friendly, attractive and dynamic web page (Lu et al., 2022; Luo et al., 2022). Server and client-side scripts play an important role in providing a better experience for web users. In contrast, malicious users or attackers use these scripts to construct direct or indirect attack vectors to attack network users (Yu et al., 2021; Deng et al., 2022). Their main purpose is to steal account credentials such as usernames and passwords, personal details, session cookies, gain access to remote systems and spread malware (Zhang et al., 2020, 2021).

Cross-site scripting (XSS) has become one of the main attack vectors for various websites (Lee et al., 2022). As shown in **Figure 1**, in the statistical survey recently conducted by OWASP, XSS attacks are still the most harmful attacks. Among the top ten security threats, XSS attacks rank from seventh in 2017 to third in 2021, just behind broken access control and cryptographic failures. XSS attacks are a very common security problem that exists in nearly two-thirds of applications, and their threat level is always at the forefront. An XSS attack consists of malicious code execution by attackers exploiting the XSS vulnerability left during web application development. The attacker injects malicious script content into the web application so that when a normal user accesses the web application, the malicious script is embedded in the response of the traffic data and then returned to the browser to be executed. The hazards of XSS vulnerabilities include the following (Schuckert et al., 2022): obtaining normal users' website cookie information, intercepting browser session information, and arbitrarily using the identities of other users to manifest a series of malicious behaviors. Such behaviors may lead to website hanging and controlling normal users' computers as well as phishing scams to obtain users' private information, such as bank card passwords, maliciously controlling other users' computers to carry out various distributed attacks and spreading worm scripts on the network, thereby endangering the network environment (Zhao et al., 2021).

Improving XSS vulnerability detection has become a research hotspot in the network and information security field (Kalouptoglou et al., 2022). The current XSS detection methods still have the following problems. In feature engineering, it takes too much time to manually extract features, and a lack of professional knowledge limits the feature extraction quality. In addition, the deep logical features of complex semantics are not easy to extract (Zheng and Yin, 2022). There are many encryption and obfuscation methods, and the obfuscated data greatly increase detection difficulty. In complex XSS data, there are semantic features with strong relevance, which are difficult to mine and extract by traditional techniques. With the continuous development of network technology, there will be a large number of unknown attacks that are not easy to detect. Therefore, we must pay attention to the technology of detecting XSS attacks for in-depth research. To avoid the harm caused by XSS attacks on web applications, we should use XSS

attack detection technology to regularly scan web applications. Once XSS attacks are found, we must immediately repair the corresponding XSS vulnerabilities.

Related work

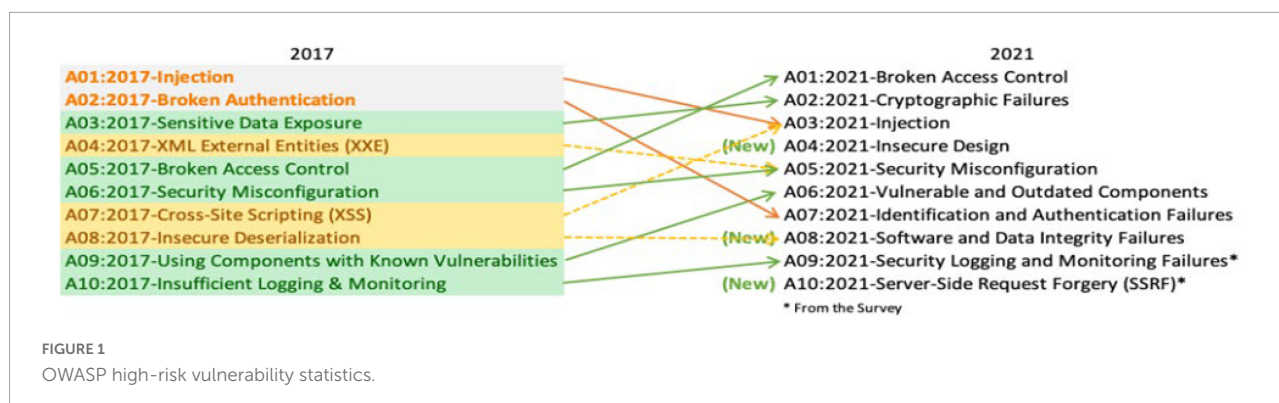
According to the HackAgon report (Hackagon, 2016), 12.75% of network attacks are XSS attacks, and almost 70% of network vulnerabilities are classified as being related to XSS vulnerabilities. Therefore, many researchers have proposed analysing web page codes to discover XSS attacks in networks. The methods used consist of static detection, dynamic detection, machine learning and deep learning.

Static detection

Static detection can directly find possible vulnerabilities by analysing the program source code when the program is not running (Liu et al., 2019). Shar and Tan (2012) proposed an automated method for statically removing XSS attacks from program code based on static analysis and pattern matching techniques. This method used static analysis and pattern matching techniques to track user input while identifying potentially vulnerable statements, discovered the location of XSS vulnerabilities and removed them. Its limitation is that it is only for the server side and cannot detect document object model (DOM)-type XSS attacks. Ahmed and Ali (2016) proposed a genetic algorithm to generate a set of test data to detect XSS attacks. They stored the data with three types of XSS attacks in the database and found the optimal method in these data through a genetic algorithm to mark all XSS attacks and verified whether these attacks were successful. This test method is used for web applications developed by PHP and MySQL. The final test results showed that the generated test data can well identify various types of XSS attacks.

Dynamic detection

Dynamic detection requires inputting test data to test the program and analysing the results and the response content of the page returned by the server. If there are specific data in the response content, then there is a vulnerability (Hou et al., 2018). Fazzini et al. (2015) proposed CSP-based web application automation technology. This technology has four parts: dynamic detection, web page analysis, CSP analysis and source code conversion. It collected the web application and test data accepted by CSP, marked the encoded value in the server-side code as trusted data, and ran the web program when performing dynamic detection analysis. Experiments showed that it can effectively detect XSS attack vulnerabilities.



Parameshwaran et al. (2015) designed a DOM XSS test platform based on taint analysis. The platform includes a detection engine and a vulnerability generator. First, it accepts the browser's request and obtains the website URL, finds the script that exists in the response and modifies it, and uses taint analysis to automatically verify the vulnerability. Then, when the platform receives a URL, it inspects the source code of the application, analyses the data stream to find potential threats, and sends it to the vulnerability generator to determine its location. Finally, a link is created to verify the original website. This method has a good effect on detecting DOM XSS attacks.

Cross-site scripting detection based on machine learning

The traditional XSS detection method usually extracts some features based on experience and then detects whether it is an XSS attack based on the rule-based matching method. However, this method cannot identify increasingly complex XSS attack sentences. With the rapid development of machine learning, an increasing number of researchers have attempted to solve problems in network security through machine learning algorithms, especially XSS attack detection, and have made corresponding progress (Wu et al., 2020, 2021a,b,c,d, 2022; Yan et al., 2021). Zhou et al. (2019) proposed a cross-site script detection model based on the combination of a multilayer perceptron and a hidden Markov model. This model preprocesses the data through a natural language processing method and then uses a multilayer perceptron to adjust the initial observation matrix of the hidden Markov model (HMM). The improved HMM improves the detection efficiency compared with the unmodified hidden Markov model. Wang et al. (2019) proposed an XSS attack detection method based on a Bayesian network. First, the nodes in the network are obtained, and 17 XSS attack characteristics are extracted. Then, malicious IP and malicious domain name information are used to improve the model. This method has achieved good detection results for nonpersistent XSS attacks. Zhao et al.

(2018) established an improved SVM classifier to identify XSS attacks and extracted typical five-dimensional features for model optimization. This method improved the detection efficiency of deformed XSS attacks.

Cross-site scripting detection based on deep learning

In recent years, researchers have applied deep learning to XSS attack detection. Luo et al. (2018, 2020) designed a URL feature representation method by analysing the existing URL attack detection technology and proposed a multisource fusion method based on a deep learning model, which can improve the detection accuracy and system stability of the entire XSS detection system. Abaimov and Bianchi (2019) presented a CODDLE model against web-based code injection attacks such as XSS. Its main novelty consists of improving the convolutional deep neural network's effectiveness via a tailored preprocessing stage that encodes XSS-related symbols into value pairs. The results showed that this model can improve the detection rate from a baseline of approximately 92% recall value, 99% precision, and 95% accuracy.

Timely detection and interception of possible attacks is an effective method for preventing XSS. Traditional vulnerability detection methods, such as static detection and dynamic detection, are unsatisfactory in the face of diverse attack loads and require considerable manual participation. The integrity of attack vectors will also have an important impact on the results. The machine learning detection method requires artificially defined features. Hence, it requires relatively high amounts of prior knowledge, and the detection effect depends heavily on the accuracy of the predefined features. The continuous maturation of deep learning in various fields provides new research directions for the XSS attack problem but also faces many challenges. The first is the automatic feature definition and extraction of deep learning, which ignores the characteristics of the security field and cannot completely retain the valid information in the URL. Second, deep learning models are

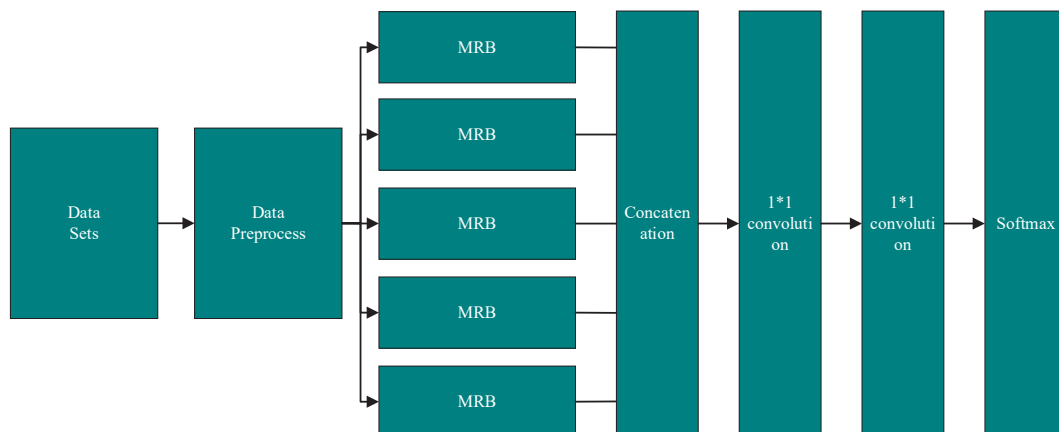


FIGURE 2
Model schematic.

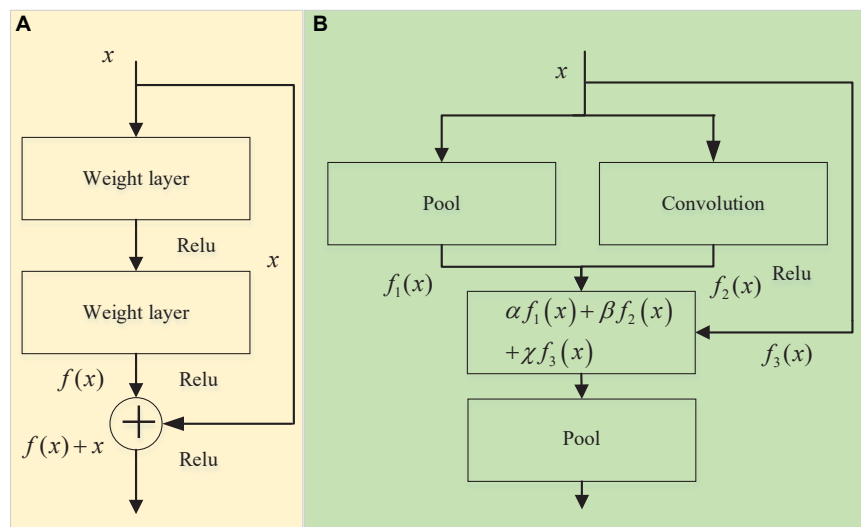


FIGURE 3
Module structure. Panel (A) is the residual block and panel (B) is the modified residual block (MRB).

usually time consuming, and stacking models increase the convergence time while improving the detection accuracy (Fan et al., 2022). On the premise of considering the characteristics of the security field, how to build a deep learning security detection model and realize the rapid detection of malicious code in URLs is a problem that needs to be considered in the current network security field.

Our approach

This manuscript analyses the hidden XSS attack in the URL from a new perspective. It treats the URL as a text language, performs word segmentation on the URL script, and then

understands the intent of the entire URL from the perspective of syntax and semantics to find the attack loaded in the URL. We modify the residual block in ResNet (MRB) and combine the 1*1 convolutional layer of NiN to replace the fully connected layer to build a modified convolution neural network-based ResNet block and NiN (MRBN-CNN).

Overall model

The overall structure of the MRBN-CNN is similar to that of the traditional CNN and is shown in Figure 2. The inputs of the entire model are normal website script data and XSS malicious attack sample data, and the feature vector is obtained

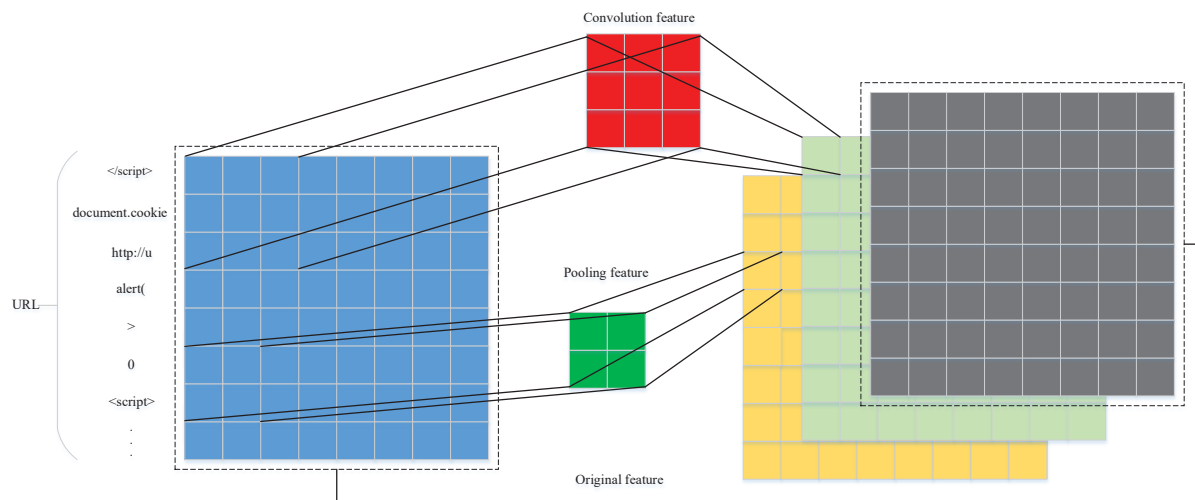


FIGURE 4
Modified residual block (MRB) module processing.

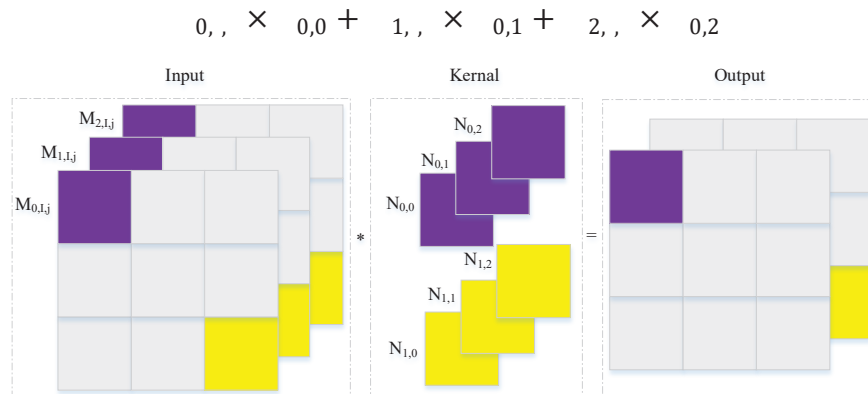


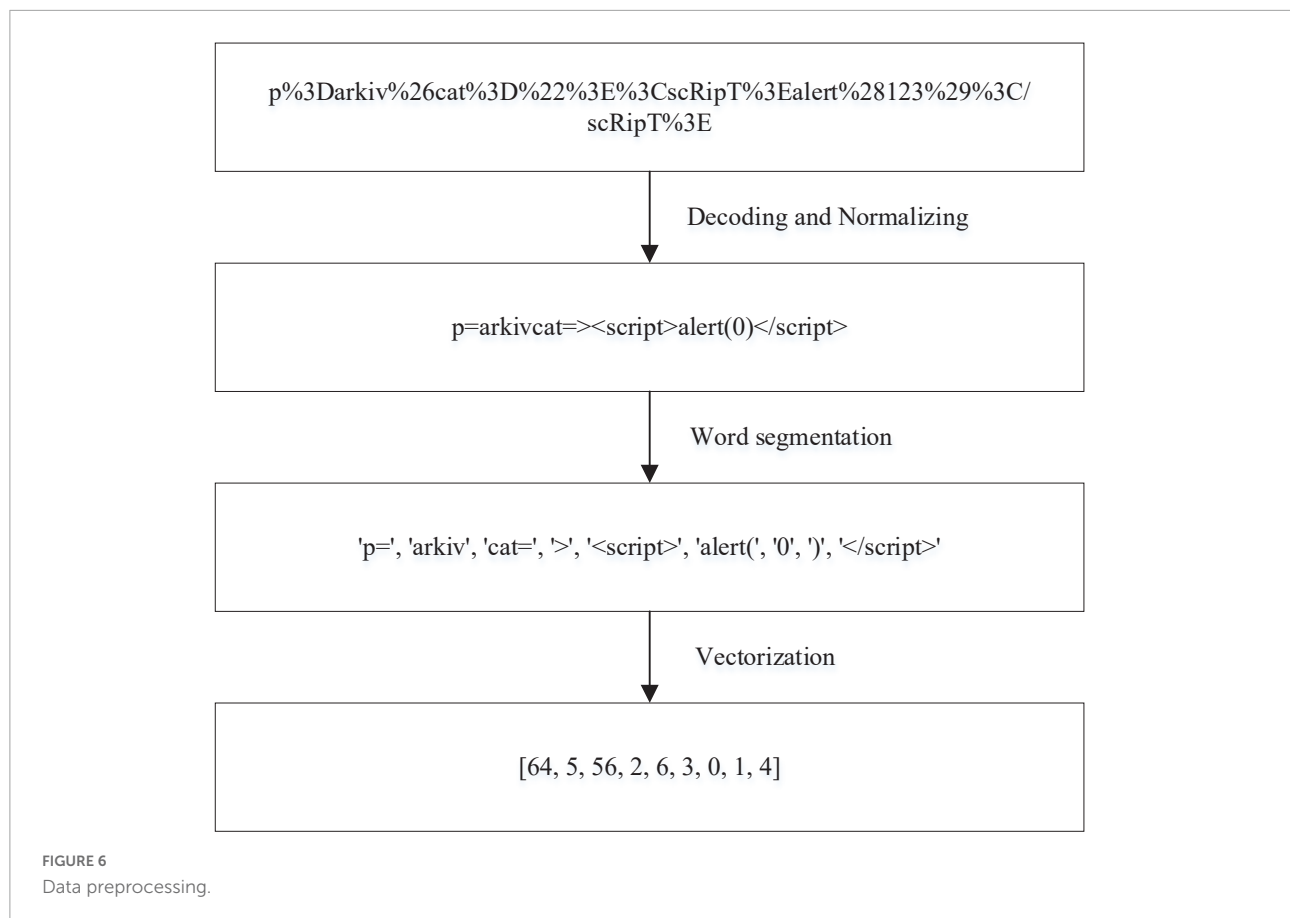
FIGURE 5
1*1 Convolution calculation.

after data preprocessing. In the deep learning model, five MRB modules are combined in parallel. By stacking multiple different convolutions, the adaptability of the whole deep learning network to different features and the comprehensiveness of feature extraction increase, but the depth of the whole neural network does not increase. In the MRB stacking part, the convolution operations in each MRB network structure use different convolution kernels for feature extraction, and the parameters of the pooling layer are different. Each MRB structure in the feature extraction layer outputs multiple feature maps, which are used to represent the effective features extracted by the MRB from the feature vector. These feature maps are concatenated and fed into a convolutional layer combination, which consists of three convolutional layers, the last two of which use a 1*1 convolution kernel. As the output layer,

softmax normalizes the final decision result and estimates the probability.

Core ideas

The model needs to learn the characteristics of normal URL scripts and XSS attack scripts from the feature vector. On the one hand, it needs to retain as much information of the entire URL as possible, and on the other hand, it needs to analyse the position and semantic relationship between words. XSS attack scripts and normal URL scripts reflect whether the grammatical and semantic relationship between various words will produce malicious operations and the different positions of various words or symbols in attack scripts and normal scripts.



The entire MRB module is designed based on these two factors. The $f_1(x)$ pooling branch and the $f_2(x)$ convolution branch in the MRB module are used to analyse the grammatical semantic relationship and positional relationship between words in the URL. The location information and semantic information hidden in the feature vector are extracted, and $f_3(x)$ is used to retain the frequency information and location information, which will compensate for the loss of URL part information in the pooling branch and convolution branch feature extraction.

ResNet is a well-known deep learning model (He et al., 2016), and its core residual module is shown in Figure 3A. The output of its module is $x + f(x)$, where $f(x)$ is composed of two convolutional layers. The entire module extracts the feature information in the input x through the convolution layer while retaining the information in the original feature vector x to avoid the loss of important features in the convolution operation during feature extraction. In this manuscript, the residual module is improved (Figure 3B). The input of the module is processed in three parts: $f_1(x)$, $f_2(x)$ and $f_3(x)$. $f_1(x)$ and $f_2(x)$ are used to learn the feature part of the input data, and their purpose is to ensure that the entire training process more easily fits the objective function. The difference from the ResNet residual module is that there is an additional pooling branch for feature extraction, while $f_3(x)$ is a high-speed channel that

maintains the input and is directly connected to the output and retains the integrity of the original input information to a certain extent. The original input feature vector x is effectively extracted from different angles, and three coefficients α , β , and χ are added when the last three branches are merged so that the entire network can learn the best combination of the three branches (Figure 4). The modified residual block (MRB) structure can be expressed as follows:

$$f_1(x) = \text{pool}(x) \quad (1)$$

$$f_2(x) = \text{Relu}(\text{Conv}(x)) \quad (2)$$

$$f_3(x) = x \quad (3)$$

$$F(x) = \text{pool}(\alpha f_1(x) + \beta f_2(x) + \chi f_3(x)) \quad (4)$$

In the classic CNN classification model, the local features obtained by the convolution operation are often connected through a fully connected layer before the output results to consider the global features of the data. However, because the fully connected layer has many parameters, it will make the model calculation more complicated. The convolution layer

TABLE 1 HTML code table.

Character	Name	Entity encoding	Decimal encoding	Hexadecimal encoding
"	Quotation marks	"	"	"
&	Logical AND	&	&	&
>	Greater than sign	>	>	>
<	Less than sign	<	<	<

TABLE 2 URL code table.

Character	Description	URL encoding
%	Special characters	%25
#	Bookmark	%23
&	The separator between the specified parameters in the URL	%26
space	Code or use the symbol '+'	%20
?	Separate the actual URL from the parameters	%3F
=	The value of the specified parameter in the URL	%3D
/	Separate directories and subdirectories	%2F
+	Space	%2B

generally needs to set the height and width, and it will identify the features in the convolution window. If the height and width of the convolutional layer are exactly 1 (Lin et al., 2013), then the calculation mode will be as shown in Figure 5. The convolution kernel has three input channels and two output channels; $(N_{0,0})$, $(N_{0,1})$, $(N_{0,2})$ corresponds to the parameters of the first channel of the output, and $(N_{1,0})$, $(N_{1,1})$, $(N_{1,2})$ correspond to the parameters of the second channel of the output. The output is multiplied by the purple part of the input and the purple part of the convolution kernel one by one, as shown in Formula 5. $(M_{0,i,j})$, $(M_{1,i,j})$, $(M_{2,i,j})$ and other input vectors on different channels are features in the MLP network, and $(N_{0,0})$, $(N_{0,1})$, $(N_{0,2})$ are weight parameters in the MLP network. The features and weights are multiplied one by one, which is almost the same as the operation of the fully connected layer. Therefore, the work required for the fully connected layer can be performed by 1*1 convolution. The experiments

use a 1*1 convolutional layer instead of fully connected layers. The convolutional neural network has the characteristics of parameter sharing, so the use of a 1*1 convolutional layer can reduce the parameters in the model under the condition of ensuring the effect of the model, thereby reducing the model complexity.

$$M_{0,i,j} N_{0,0} + M_{1,i,j} N_{0,1} + M_{2,i,j} N_{0,2} \quad (5)$$

Dataset preprocessing

Data preprocessing cannot only greatly affect the final detection ability of a model but also determine the difficulty of training a model. To improve the modeling quality, the collected positive sample data and negative sample data need to be preprocessed. Due to the particularity of XSS attacks, the collected dataset is in the form of text. Hence, natural language processing is used to process the data. The process is roughly divided into three steps: data coding and normalization, word segmentation and vectorization. All data preprocessing steps are shown in Figure 6.

The purpose of data encoding and normalization is to exclude noncritical information and minimize the impact of nonimportant information on the algorithm model construction. To ensure the safety and reliability of the data, noncritical information regarding the protocol, domain name, port, etc., in the URL request is excluded. Instead, only the virtual directory, file name and parameters are retained as valid information to train the model. XSS attacks are encoded to evade detection, including URL encoding, HTML encoding and JavaScript encoding. The HTML encoding includes HTML entity encoding and HTML system encoding. HTML entity encoding can distinguish itself from semantic markup. This entity code begins with an "&" symbol and ends with a semicolon. For example, to encode "<", the HTML entity encodes it as "<". HTML system encoding, starting with the "&#" symbol and ending with a semicolon. Normally, only HTML decimal and HTML hexadecimal are recognized. For example, to encode "<", HTML decimal encodes it as "<" and HTML hex encodes it as "<". Common HTML encodings are shown in Table 1. The URL encoding method is very simple, and attackers can easily complete XSS attacks by using URL encoding. For example, angle brackets "<", URL-encoded as "%3C". Table 2 shows the common

TABLE 3 JavaScript code table.

Different forms	Function code
JavaScript octal	<script>eval("\163\163\57\51\164\50\57\170\141\154\145\162");</script>
JavaScript hexadecimal coding	<script>eval("\x73\x73\x2f\x29\x74\x28\x2f\x78\x61\x6c\x65\x72");</script>
Junicode coding	<script>eval("\u0073\u0073\u002f\u0029\u0074\u0028\u002f\u0078\u0061\u006c\u0065\u0072");</script>

URL-encoded characters in XSS attacks. There are many forms of JavaScript coding, including JavaScript hexadecimal coding, JavaScript octal coding and Jsunicode coding. For example, “<” is encoded by JavaScript hex as “\x3c”, JavaScript octal as “\074”, and Jsunicode as “\u003c”. JavaScript coding will not be parsed in HTML tags in browsers, because Jsunicode can be used for coding, but only function names can be coded. The onerror event in Javascript coding is special. Onerror event can capture JavaScript errors in web pages, so the content in onerror event can be parsed by JavaScript. Several JavaScript codes are shown in **Table 3**. According to these three codes, the XSS attack adopts malicious deformation to avoid detection, and direct feature extraction will lose the attack code characteristics, which is not conducive to detection accuracy. Thus, the corresponding decoding must be performed first. After decoding, to reduce the number of word segmentations, it is necessary to normalize numbers and hyperlinks; for example, “0” is used to replace numbers, and “http://u” is used to replace hyperlinks.

According to the characteristics of the XSS attack script, we design the word segmentation principles that meet the syntax and semantics requirements: single and double quotation marks, http/https hyperlinks, end tag, start tag, attribute name, and function body. These six word segmentation principles are matched with their corresponding regular expressions. The word segmentation rules are shown in **Table 4**.

Vectorization uses the CBOW model in word2vec to convert text into digital vectors that can be recognized by computers. The converted word vectors cannot only represent words as distributed word vectors but also capture the similarity between words. To verify the effect of the trained word vector, t-SNE is used to visualize the word vector (**Figure 7**).

Experiments and results

Dataset

The data of normal samples (negative samples) come from the DMOZ database, and 75,428 pieces of standard data are obtained after data preprocessing. The malicious samples (positive samples) come from the XSSed database and the tested payload (Payload) in the penetration test. Additionally, 75,428 pieces of standard data are obtained to ensure a balanced selection of samples (Zheng et al., 2021; Cai et al., 2022). In the experiment, the training set and the test set are randomly selected from the samples at a ratio of 7:3. Our experiment was performed using a notebook computer with a 3.20 GHz AMD Ryzen 7 5800H, 32 GB of RAM, NVIDIA GTX3070 of GPU, Ubuntu16.04 operating system. The Keras framework based on Tensorflow-Gpu is used.

TABLE 4 Word segmentation rules.

Word segmentation rules	Regular expression
Function body	(?x)[\w\.\.]+\?
Attribute name	\w+=
Start tag	<\w+>
End tag	</\w+>
http/https hyperlinks	http://s+,https://s+
Single and double quotation marks	['"]+['"]+

Metrics

We use four indicators of recall, precision, accuracy, and F1 as the evaluation criteria for the model performance results. The formulas for the indicators are as follows:

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

$$Precision = \frac{TP}{TP + FP} \quad (7)$$

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP} \quad (8)$$

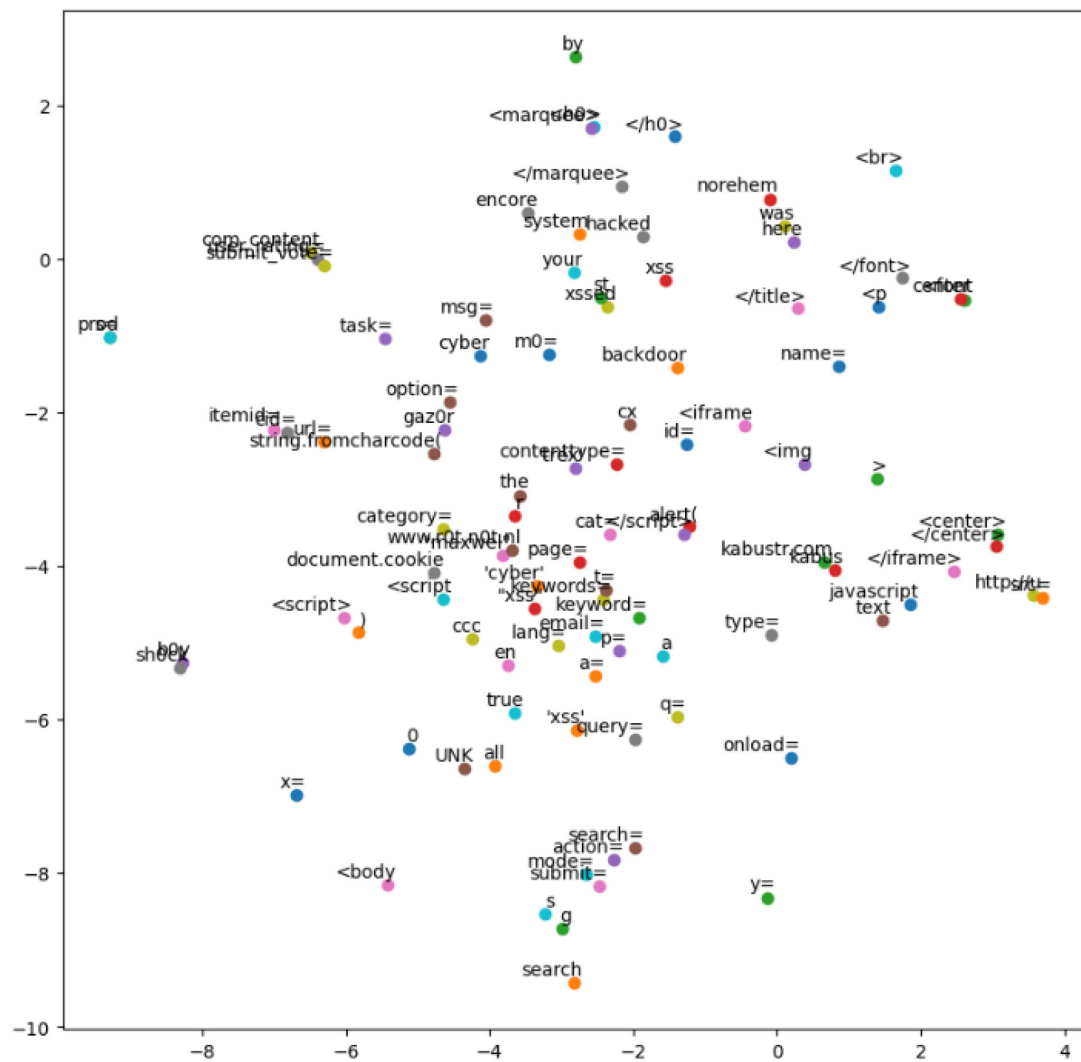
$$F1 = \frac{2 * (Precision * Recall)}{Precision + Recall} \quad (9)$$

In these formulas, FN is the abbreviation for false negatives, which means that malicious samples are identified as normal samples, FP is the abbreviation for false positives, which means that normal samples are identified as malicious samples, TN is the abbreviation for true negatives, which means that normal samples are identified as normal samples, TP is the abbreviation for true positives, which means that malicious samples are identified as malicious samples.

Model training

The effect of vector dimensions on model performance

Model training needs to choose a suitable vector dimension to make full use of the sample information. If the vector dimension is too short, a large amount of effective information will be lost, and the detection accuracy will be reduced. In contrast, if the vector dimension is too long, the training time will greatly increase, the accuracy cannot be improved, and the real-time detection performance will be affected. To obtain a suitable vector dimension, this manuscript compares the effects of different vector dimensions on the accuracy and training time, and the results are shown in **Figure 8**. The experimental results show that the accuracy does not change significantly when the dimension exceeds 100, but the training time increases



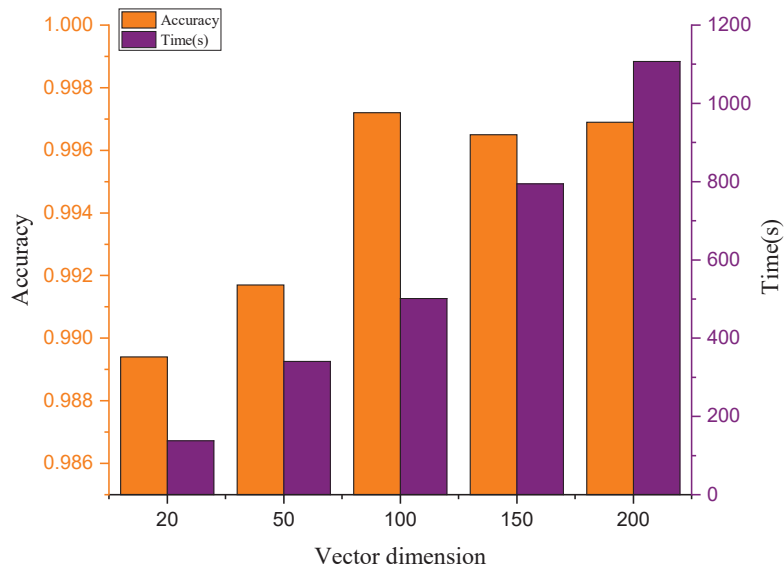


FIGURE 8
Training time and accuracy under different vector dimensions.

as malicious URLs by the model. When the convolution kernel combination is modified to Group-D, it can be seen that the values of the three evaluation indicators of accuracy, recall and precision significantly improve based on the convolution kernel combination to Group-C, and both accuracy and recall reach the maximum values of their respective records. This indicates that when the convolution kernel combination is Group-D, the semantic and grammatical information between the words of the URL can be extracted more accurately. In contrast, when the convolution kernel combination is modified to Group-E, recall decreases significantly, and accuracy and precision also decrease to a certain extent. This indicates that the omission rate of the whole model increases significantly, and more malicious URLs are recognized as normal URLs by the model. As we continue to modify the size of the convolution kernel, from Group-F to Group-G, it can be seen that the gap between the three evaluation indicators of accuracy, recall and precision becomes increasingly obvious. Based on the accuracy, recall and precision of the seven groups of experiments, we adjusted the convolution kernel combination in the MRBN neural network model according to Group-D.

Model testing

To verify the effectiveness and advantages of the MRBN model, we design comparative experiments involving machine learning and deep learning.

Machine learning comparison experiments

Three classic machine learning algorithms, namely, AdaBoost (Freund and Schapire, 1997), ADTree (Freund and

Mason, 1999), and SVM (Cortes and Vapnik, 1995), were selected for comparative experiments. AdaBoost trains multiple weak classifiers and then aggregates the weak classifiers into

TABLE 5 Details of convolution kernel groupings.

Experimental grouping Combination of convolution kernels

Group-A	3*3, 2*2, 5*5, 4*2, 2*1
Group-B	2*1, 3*5, 5*5, 3*1, 3*4
Group-C	2*1, 3*1, 3*4, 3*5, 5*1
Group-D	2*1, 5*5, 7*7, 4*4, 3*1
Group-E	2*1, 3*2, 5*5, 4*4, 3*5
Group-F	2*1, 5*5, 3*5, 4*4, 3*1
Group-G	2*1, 5*5, 3*5, 4*4, 3*2

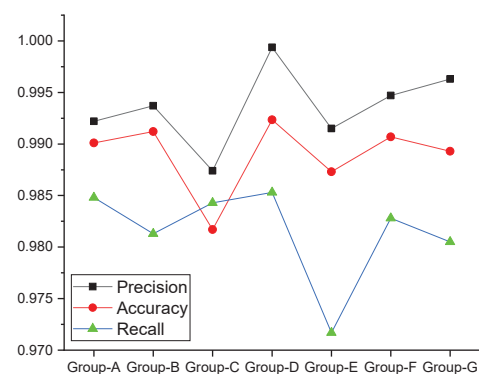


FIGURE 9
The influence of the convolution kernel on the MRBN.

TABLE 6 The result of comparing machine learning.

Models	Precision (%)	Accuracy (%)	Recall (%)	F1 (%)
SVM	95.71	91.35	86.59	90.92
ADTree	96.47	92.37	87.96	92.02
AdaBoost	98.48	93.41	88.18	93.05
MRBN-CNN	99.94	99.23	98.53	99.23

The best results are highlighted in bold.

TABLE 7 The result of comparing deep learning.

Models	Precision (%)	Accuracy (%)	Recall (%)	F1 (%)
GRU	98.89	92.68	86.32	92.18
CNN	98.56	94.53	90.38	94.29
LSTM	99.15	96.43	93.67	96.33
BiLSTM	98.47	96.18	93.81	96.09
BiLSTM-CNN	99.99	97.34	94.69	97.27
MRBN-CNN	99.94	99.23	98.53	99.23

The best results are highlighted in bold.

a strong classifier (Hastie et al., 2009). ADTree is a decision tree learning algorithm based on boosting, and its classification performance is better than other decision trees. Support vector machine (SVM) is a linear classifier that performs binary classification on data according to supervised learning. The experimental results are shown in Table 6. The three machine learning models have good results and reasonable accuracy values, but the recall value is not very good, and the false negative rate in the detection results is high. This indicates that the three models have not truly learned the characteristics that can identify malicious URLs and normal URLs. The accuracy

of the MRBN-CNN model reaches 99.23%, the precision is 99.94%, the recall is 98.53%, and the F1 value is 99.23%. Compared with the three machine learning algorithms, the proposed model greatly improves the detection effect. This is because it can learn relevant features in URLs very accurately from three perspectives.

Deep learning comparison experiments

The GRU, CNN, LSTM, BiLSTM, and BiLSTM-CNN are selected for comparison experiments with our model. The experimental results are shown in Table 7 and Figure 10. It can be seen that the accuracy and precision of the GRU model are good, but the recall is poor, indicating that the system shows a high false negative rate in the experiment. This means that the system does not accurately learn the characteristics of XSS attacks in URLs, resulting in identifying many URLs with attack payloads as normal URL requests. The CNN, LSTM, and BiLSTM models have better performance and achieve better accuracy. These systems have been able to learn the characteristics of XSS attacks in URLs to a certain extent. The precision of the BiLSTM-CNN model is as high as 99.99%, and its accuracy also reaches 97.34%, but the recall is slightly worse, indicating that this model can better learn the relevant features in the URL. The MRBN-CNN model performs better, and the values of the three indicators are very close. It is a stable system. It learns the characteristics of XSS attacks in URLs very accurately. It cannot only detect malicious URLs but also ensure fewer false positive and false negatives. Experiments show that the improved method proposed in this work can accurately learn the potential XSS attack features in URLs and can fit a very suitable high-dimensional function to correctly classify URLs. Compared with other works, it shows a certain superiority.

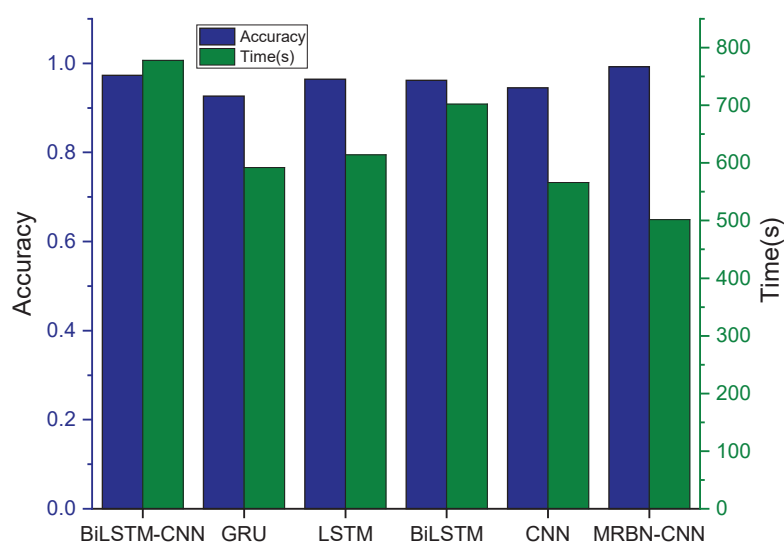


FIGURE 10

The result of comparing deep learning.

Because the deep learning model is usually time consuming, the stacking model will increase the convergence time while improving the detection accuracy (Zhou et al., 2022). Hence, we compare the convergence time of the above deep learning models, and the results are shown in Figure 10. It can be seen that the CNN convergence time is relatively short, while the convergence time of the other models gradually increases, and the BiLSTM-CNN model has the longest convergence time. In contrast, MRBN-CNN replaces the fully connected layer by a 1×1 convolution, the model parameters are greatly reduced, the training difficulty is reduced, and its convergence time is the least.

Conclusion

This manuscript proposes an MRBN-CNN model. Its significance is as follows. First, by applying natural language processing technology to URLs for attack detection, learning the semantics and syntax in URLs and performing feature representation can filter out irrelevant information. Second, in the deep learning model design, combined with the traditional ResNet module modification for the XSS attack scenario, the MRB module was designed and proposed. It can obtain the semantic and grammatical information of the feature vector without losing the relevant position, frequency and other basic information and can realize the accurate identification of the attack with a low false-positive rate. Third, by replacing the fully connected layer with a 1×1 convolution, the model parameters can be reduced, the training difficulty can be reduced, and the phenomenon that too many parameters cause overfitting can be avoided. This manuscript only uses the MRBN-CNN model to detect XSS vulnerability attacks. In the future, we will study the applicability of this model to various web vulnerability detection and vulnerability mining, such as buffer overflow, SQL injection, and cross-site request forgery.

Data availability statement

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

References

- Abaimov, S., and Bianchi, G. (2019). CODDLE: Code-injection detection with deep learning. *IEEE Access* 7, 128617–128627. doi: 10.1109/ACCESS.2019.2939870
- Ahmed, M. A., and Ali, F. (2016). Multiple-path testing for cross site scripting using genetic algorithms. *J. Syst. Arch.* 64, 50–62. doi: 10.1016/j.sysarc.2015.11.001
- Cai, L., Xiong, L., Cao, J., Zhang, H., and Alsaadi, F. E. (2022). State quantized sampled-data control design for complex-valued memristive neural networks. *J. the Franklin Inst.* 359, 4019–4053. doi: 10.1016/j.jfranklin.2022.04.016
- Cao, K., Wang, B., Ding, H., Lv, L., Tian, J., Hu, H., et al. (2021). Achieving reliable and secure communications in wireless-powered NOMA systems. *IEEE Trans. Vehicular Technol.* 70, 1978–1983. doi: 10.1109/TVT.2021.3053093

Author contributions

HY: conceptualization and editing. HY, LeF, LiF, and CL: funding acquisition. YY: project administration. WL: supervision. DL and YZ: validation. LQ and XZ: data curation. JZ: visualization. All authors read and agreed to the published version of the manuscript.

Funding

This study was supported by Youth Project of Science and Technology Research Program of Chongqing Education Commission of China (grant no. KJQN202100812), the Youth Fund of Chongqing Technology and Business University (grant no. 1952033), the Scientific and Technological Research Program of Chongqing Municipal Education Commission under Grant KJQN202000841, the Opening Research Platform of Chongqing Technology and Business University (grant no. KFJJ2018058), and the Funds for Creative Research Groups of Chongqing Municipal Education Commission under Grant CXQT21034.

Conflict of interest

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

Publisher's note

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

- Cortes, C., and Vapnik, V. (1995). Support-vector networks. *Machine Learn.* 20, 273–297. doi: 10.1007/BF00994018
- Deng, S., Zhang, J., Wu, D., He, Y., Xie, X., and Wu, X. (2022). A quantitative risk assessment model for distribution cyber physical system under cyber attack. *IEEE Trans. Indus. Inform.* 1–1. doi: 10.1109/TII.2022.3169456
- Fan, Q., Zhang, Z., and Huang, X. (2022). Parameter conjugate gradient with secant equation based Elman neural network and its convergence analysis. *Adv. Theor. Simulat.* doi: 10.1002/adts.202200047
- Fazzini, M., Saxena, P., and Orso, A. (2015). “AutoCSP: Automatically retrofitting CSP to web applications,” in *Proceedings of the 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*, Florence, 336–346. doi: 10.1109/ICSE.2015.53
- Freund, Y., and Mason, L. (1999). “The alternating decision tree learning algorithm,” in *Proceedings of the Sixteenth International Conference on Machine Learning*, (San Francisco, CA: Morgan Kaufmann Publishers Inc), 124–133.
- Freund, Y., and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.* 55, 119–139. doi: 10.1006/jcss.1997.1504
- Hackagon (2016). *XSS Attack*. Available online at: <http://hackagon.com/xss-attack> (accessed December 31, 2016).
- Hastie, T., Rosset, S., Zhu, J., and Zou, H. (2009). Multi-class adaboost. *Stat. Interf.* 2, 349–360. doi: 10.4310/SII.2009.v2.n3.a8
- He, K., Zhang, X., Ren, S., and Sun, J. (2016). “Deep residual learning for image recognition,” in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, 770–778. doi: 10.1109/CVPR.2016.90
- Hou, X., Zhao, X., Wu, M., Ma, R., and Chen, Y. (2018). “A dynamic detection technique for XSS vulnerabilities,” in *Proceedings of the 2018 4th Annual International Conference on Network and Information Systems for Computers (ICNISC)*, Wuhan, 34–43. doi: 10.1109/ICNISC.2018.00016
- Kaloutsoglou, I., Siavvas, M., Kehagias, D., Chatzigeorgiou, A., and Ampatzoglou, A. (2022). Examining the capacity of text mining and software metrics in vulnerability prediction. *Entropy* 24:651. doi: 10.3390/e24050651
- Kotzur, M. (2022). “Privacy protection in the world wide web—legal perspectives on accomplishing a mission impossible,” in *Personality and Data Protection Rights on the Internet: Brazilian and German Approaches*, eds M. Albers and I. W. Sarlet (Cham: Springer International Publishing). doi: 10.1007/978-3-030-90331-2_2
- Lee, S., Wi, S., and Son, S. (2022). “Link: Black-box detection of cross-site scripting vulnerabilities using reinforcement learning,” in *Proceedings of the ACM Web Conference 2022*, (Virtual Event, Lyon: Association for Computing Machinery), 743–754. doi: 10.1145/3485447.3512234
- Lin, M., Chen, Q., and Yan, S. (2013). Network in network. *Comput. Sci.* 1–10. doi: 10.48550/arXiv.1312.4400
- Liu, M., Zhang, B., Chen, W., and Zhang, X. (2019). A survey of exploitation and detection methods of XSS vulnerabilities. *IEEE Access* 7, 182004–182016. doi: 10.1109/ACCESS.2019.2960449
- Lu, S., Ban, Y., Zhang, X., Yang, B., Liu, S., Yin, L., et al. (2022). Adaptive control of time delay teleoperation system with uncertain dynamics. *Front. Neurobot.* 16:928863. doi: 10.3389/fnbot.2022.928863
- Luo, C., Wang, L., and Lu, H. (2018). *Analysis of LSTM-RNN Based on Attack Type of KDD-99 Dataset*. The Netherlands: ICCCS. doi: 10.1007/978-3-030-00006-6_29
- Luo, C. C., Su, S., Sun, Y., Tan, Q., Han, M., and Tian, Z. (2020). A convolution-based system for malicious URLs detection. *CMC- Computers, Materials & Continua* 62, 399–411. doi: 10.32604/cmc.2020.06507
- Luo, G., Zhang, H., Yuan, Q., Li, J., and Wang, F. Y. (2022). ESTNet: Embedded spatial-temporal network for modeling traffic flow dynamics. *IEEE Trans. Intellig. Trans. Syst.* 1–12. doi: 10.1109/TITS.2022.3167019
- Parameshwaran, I., Budianto, E., Shinde, S., Dang, H., Sadhu, A., and Saxena, P. (2015). “DexterJS: robust testing platform for DOM-based XSS vulnerabilities,” in *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering*, (Bergamo: Association for Computing Machinery), 946–949. doi: 10.1145/2786805.2803191
- Schuckert, F., Langweg, H., and Katt, B. (2022). “Systematic generation of XSS and SQLi vulnerabilities in PHP as test cases for static code analysis,” in *Proceedings of the 2022 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, Valencia, 261–268. doi: 10.1109/ICSTW55395.2022.00053
- Shar, L. K., and Tan, H. B. K. (2012). Automated removal of cross site scripting vulnerabilities in web applications. *Inform. Softw. Technol.* 54, 467–478. doi: 10.1016/j.infsof.2011.12.006
- Wang, P. C., Zhou, Y., Zhu, C., and Zhang, W. M. (2019). XSS attack detection based on Bayesian network. *J. Univ. Sci. Technol. China* 49, 166–172.
- Wu, D., He, Q., Luo, X., Shang, M., He, Y., and Wang, G. (2022). A posterior-neighborhood-regularized latent factor model for highly accurate web service QoS prediction. *IEEE Trans. Serv. Comput.* 15, 793–805. doi: 10.1109/TSC.2019.2961895
- Wu, D., He, Y., Luo, X., and Zhou, M. (2021a). A latent factor analysis-based approach to online sparse streaming feature selection. *IEEE Trans. Syst. Man Cybernet. Syst.* 1–15. doi: 10.1109/TSMC.2021.3096065
- Wu, D., Luo, X., Shang, M., He, Y., Wang, G., and Wu, X. (2022). A data-characteristic-aware latent factor model for web services QoS prediction. *IEEE Trans. Knowl. Data Eng.* 34, 2525–2538. doi: 10.1109/TKDE.2020.3014302
- Wu, D., Luo, X., Shang, M., He, Y., Wang, G., and Zhou, M. (2021b). A deep latent factor model for high-dimensional and sparse matrices in recommender systems. *IEEE Trans. Syst. Man Cybernet. Syst.* 51, 4285–4296. doi: 10.1109/TSMC.2019.2931393
- Wu, D., Shang, M., Luo, X., and Wang, Z. (2021c). An L1-and-L2-Norm-Oriented latent factor model for recommender systems. *IEEE Trans. Neural Netw. Learn. Syst.* 1–14. doi: 10.1109/TNNLS.2021.3071392
- Wu, X., Zheng, W., Chen, X., Zhao, Y., Yu, T., and Mu, D. (2021d). Improving high-impact bug report prediction with combination of interactive machine learning and active learning. *Inform. Softw. Technol.* 133:106530. doi: 10.1016/j.infsof.2021.106530
- Yan, H. Y., He, J., Xu, X., Yao, X., Wang, G., Tang, L., et al. (2021). Prediction of potentially suitable distributions of *Codonopsis pilosula* in China based on an optimized MaxEnt model. *Front. Ecol. Evol.* 9:773396. doi: 10.3389/fevo.2021.773396
- Yu, J., Lu, L., Chen, Y., Zhu, Y., and Kong, L. (2021). An Indirect Eavesdropping Attack of Keystrokes on Touch Screen through Acoustic Sensing. *IEEE Trans. Mobile Comput.* 20, 337–351. doi: 10.1109/TMC.2019.2947468
- Zhang, M., Chen, Y., and Lin, J. (2021). A privacy-preserving optimization of neighborhood-based recommendation for medical-aided diagnosis and treatment. *IEEE Internet Things J.* 8, 10830–10842. doi: 10.1109/JIOT.2021.3051060
- Zhang, M., Chen, Y., and Susilo, W. (2020). PPO-CPQ: A privacy-preserving optimization of clinical pathway query for E-healthcare systems. *IEEE Internet Things J.* 7, 10660–10672. doi: 10.1109/JIOT.2020.3007518
- Zhao, C., Jun-xin, C., and Ming-hai, Y. (2018). XSS attack detection technology based on SVM classifier. *Comput. Sci.* 45, 356–360.
- Zhao, S., Li, F., Li, H., Lu, R., Ren, S., Bao, H., et al. (2021). Smart and practical privacy-preserving data aggregation for fog-based smart grids. *IEEE Trans. Inform. Forensics Secur.* 16, 521–536. doi: 10.1109/TIFS.2020.3014487
- Zheng, W., Xun, Y., Wu, X., Deng, Z., Chen, X., and Sui, Y. (2021). A comparative study of class rebalancing methods for security bug report classification. *IEEE Trans. Reliab.* 70, 1658–1670. doi: 10.1109/TR.2021.3118026
- Zheng, W., and Yin, L. (2022). Characterization inference based on joint-optimization of multi-layer semantics and deep fusion matching network. *PeerJ Comput. Sci.* 8:e908. doi: 10.7717/peerj-cs.908
- Zhou, K., Wan, L., and Ding, H. W. (2019). A cross-site script detection method based on MLP-HMM. *Comput. Eng. Sci.* 41, 1413–1420.
- Zhou, L., Fan, Q., Huang, X., and Liu, Y. (2022). Weak and strong convergence analysis of Elman neural networks via weight decay regularization. *Optimization* 1–23. doi: 10.1080/02331934.2022.2057852



OPEN ACCESS

EDITED BY

Song Deng,
Nanjing University of Posts
and Telecommunications, China

REVIEWED BY

Huyong Yan,
Chongqing Technology and Business
University, China
Jia Chen,
Beihang University, China

*CORRESPONDENCE

Kun-hua Zhong
zhongkunhua@cigit.ac.cn

RECEIVED 19 July 2022

ACCEPTED 18 August 2022

PUBLISHED 07 September 2022

CITATION

Chen Y-w, Zhang J, Wang P, Hu Z-y
and Zhong K-h (2022)
Convolutional-de-convolutional
neural networks for recognition
of surgical workflow.
Front. Comput. Neurosci. 16:998096.
doi: 10.3389/fncom.2022.998096

COPYRIGHT

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

Convolutional-de-convolutional neural networks for recognition of surgical workflow

Yu-wen Chen¹, Ju Zhang¹, Peng Wang², Zheng-yu Hu¹ and
Kun-hua Zhong^{1*}

¹Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Chongqing, China, ²Southwest Hospital, Third Military Medical University, Chongqing, China

Computer-assisted surgery (CAS) has occupied an important position in modern surgery, further stimulating the progress of methodology and technology. In recent years, a large number of computer vision-based methods have been widely used in surgical workflow recognition tasks. For training the models, a lot of annotated data are necessary. However, the annotation of surgical data requires expert knowledge and thus becomes difficult and time-consuming. In this paper, we focus on the problem of data deficiency and propose a knowledge transfer learning method based on artificial neural network to compensate a small amount of labeled training data. To solve this problem, we propose an unsupervised method for pre-training a Convolutional-De-Convolutional (CDC) neural network for sequencing surgical workflow frames, which performs neural convolution in space (for semantic abstraction) and neural de-convolution in time (for frame level resolution) simultaneously. Specifically, through neural convolution transfer learning, we only fine-tuned the CDC neural network to classify the surgical phase. We performed some experiments for validating the model, and it showed that the proposed model can effectively extract the surgical feature and determine the surgical phase. The accuracy (Acc), recall, precision (Pres) of our model reached 91.4, 78.9, and 82.5%, respectively.

KEYWORDS

neural networks, convolutional-de-convolutional, transfer learning, surgical workflow, deep learning

Introduction

Computer-assisted surgery (CAS) emerged in the twentieth century, which means that computer technology is used to guide and assist surgeons. The application (Garg et al., 2005) provides decision-making support and planning tools in the preoperative. Intraoperative computer assistance includes robotic surgical system (Dergachyova, 2018), image guidance and navigation (Peters, 2006), augmented reality and visualization (Kersten-Oertel et al., 2013). Postoperative assistance provides tools to

analyze executed procedures and results, as well as to improve and optimize (Schumann et al., 2015). Despite all the advance and valuable assistance, the seamless integration of computer-aided equipment with operating room (OR) and surgical procedures has not yet been achieved. Existing ORs contain a set of unrelated independent systems and devices, most of which appear in isolation, disabling proper communication and interaction (Hübner et al., 2014). Current computer-aided equipment facilitates a number of individual surgical tasks, but their lack of synchronization with the surgical process hampers the work and resource management of the surgical team. It leads to higher stress levels (Agarwal et al., 2006), frequent misunderstandings among surgical staffs, resulting in risks and delays, as well as inefficient surgical groups that incur excessive costs for hospitals (Macario, 2010).

Context-aware Computer-assisted surgery (CA-CAS) has powerful artificial intelligence that understands or perceives the needs of clinicians. It should always be aware of the events that occur, the actions performed, and the current state by tracking the surgical procedure and constantly observing the surgical site. Examples of applications are: optimization of the surgical procedure (Franke et al., 2013; Guédon et al., 2016), prediction of the remaining time of surgery (Bhatia et al., 2007), intraoperative assistance (Nessi et al., 2015; Fard et al., 2016), automatic generation of surgical reports (Agarwal et al., 2006). A large number of studies have focused on IntelliSense intraoperative aids to reduce the pressure on surgeons and facilitate the surgical process (Meng et al., 2021; Liu et al., 2022). Automatic recognition of surgical procedures is an important part of this. Recognizing surgical procedures is a prerequisite for CAS applications. The study on this subject began about 10 years ago. Despite the great progress made, it remains a relatively new area that inspires scientists and clinicians to inspire. Due to the lack of automatic recognition, most applications use manual label of surgical activities, which is a very tedious and time-consuming process.

Today, artificial intelligence and deep learning technologies have developed rapidly (Li et al., 2017; Liu et al., 2020; Zhong et al., 2021; Fan et al., 2022) and have been successfully applied in many different fields, including image labeling, natural language modeling, text generation, image labeling, natural language modeling, text generation, classification (Zheng et al., 2021), medical care (Zhang et al., 2020, 2021), web service QoS prediction (Wu et al., 2022), and risk assessment (Deng et al., 2022). In most cases, their performance is superior to that of traditional machine learning methods. Comprehensive and accurate training data have been playing an important role in machine learning. The quantity and quality of data have become an important factor. The size of the massive data sets that serve as a basis for the training of deep learning model, such as the famous ImageNet (Deng et al., 2009), Microsoft COCO (Deng et al., 2009), the recently released Google's OpenImages

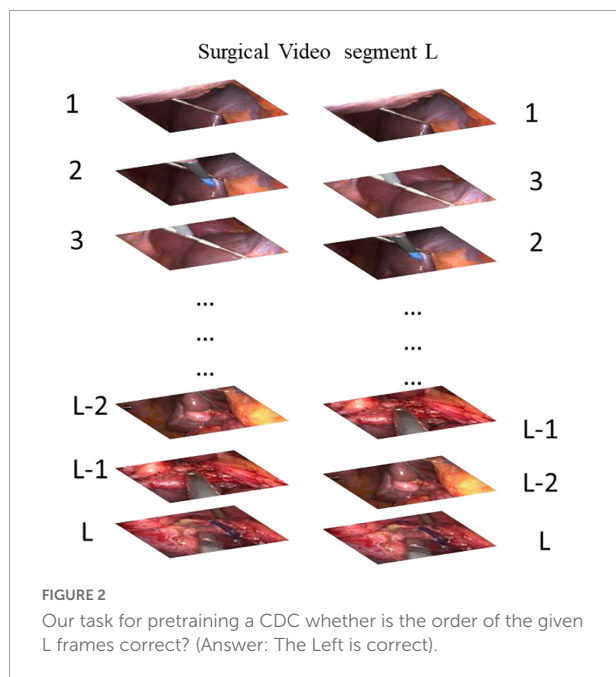
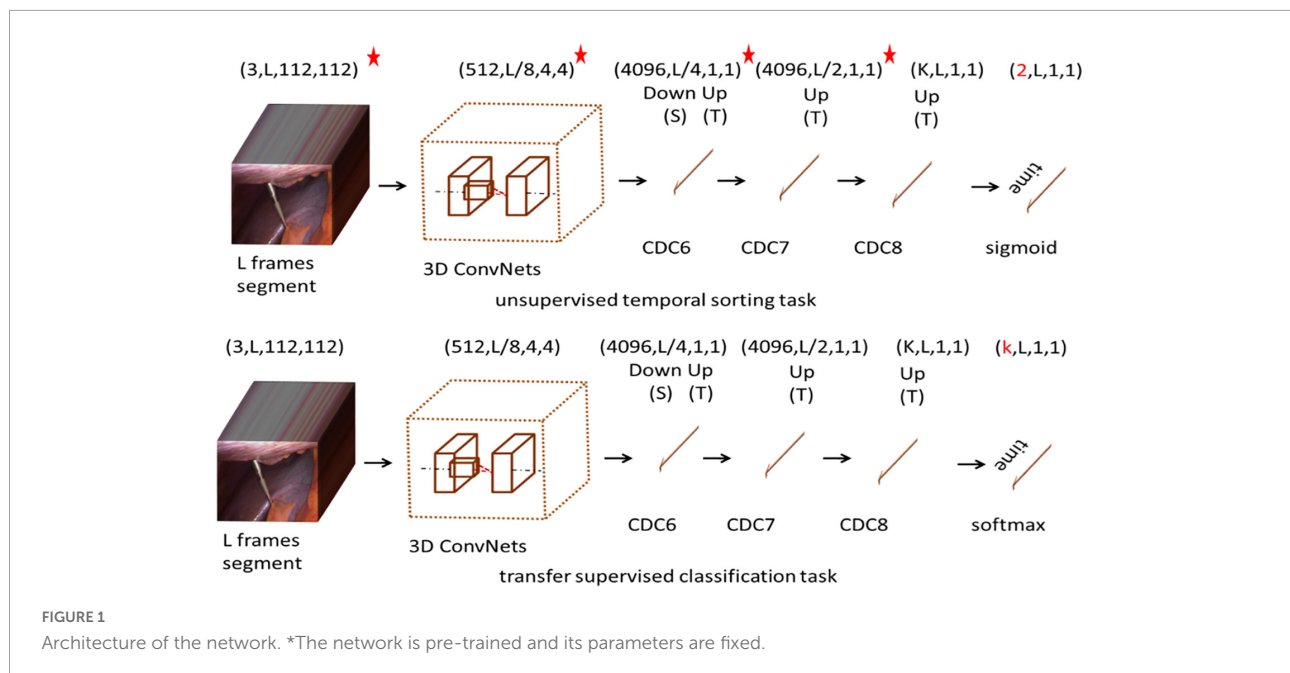
(Krasin et al., 2017; Kuznetsova et al., 2020), and YouTube-8M (Abu-El-Haija et al., 2016; YouTube-8M Dataset, 2018), is self-evident. They contain millions of samples representing thousands of categories. Unfortunately, sometimes learning tasks have to be carried out in an area of interest expressed by a small group of data, such as the field of surgery. A variety of constraints hinder proper data collection: Ethical approvals, the consent of patients and medical personnel, the limited number of cases, the installation of expensive data acquisition equipment, and time-consuming manual annotations that require medical experience. In these cases, the methods of transfer learning may play a role. To a large extent, transfer learning involves the use of methods from resources in other areas of interest, where data may be distributed differently and located in different feature spaces, thus improving the learning of the target task. Depth models make it easy to transfer knowledge of one network to another. Transfer learning is a knowledge transfer technology that is currently widely used with convolution neural networks (CNN) for tasks related to visual content, which benefits from a large number of free datasets. It is also widely used in speech and language processing (Huang et al., 2013), document classification (Dai et al., 2007), sentiment analysis (Glorot et al., 2011), and other sequence analysis tasks.

Therefore, in this paper, we proposed an unsupervised method for training Convolutional-De-Convolutional (CDC) networks to sort surgical workflow frames, which are simultaneously rolled out in space (for semantic abstraction) and temporal convolution (for frame-level resolution). It has unique property in modeling the spatio-temporal interactions between high-level semantics in space and fine-grained action dynamics in time. Specifically, the CDC has to extract features related to understanding the surgical workflow. The knowledge learned from the task is encoded into the weight matrix of the internal parameters of the representation layer. Then the Convolutional-De-Convolution network is fine-tuned to classify the surgical phase.

The contributions of this paper are summarized as follows:

- We proposed a model that can solve the problem of annotating data deficiency in medical field by using the transfer learning method.
- We used a CDC network to recognize the surgical workflow because of its property of spatio-temporal interactions in training.
- We try to achieve intelligent detection of surgical video phase at a low cost. Finally, based on M2CAI 2016 challenge dataset, we performed experiments for validating the model. It shows a good performance compared with other methods.

This paper is organized as follows: Section II presents related work. We summarize methodology and the proposed models



in section III. In section IV, we present the experiment and result of our method. In section V, we discuss conclusions and suggestions for future research.

Related work

The OR's understanding of surgical activities is a new field of research. Surgical workflow identification is closely

related to multi-target tracking. Wang et al. (2022) proposed a General Recurrent Tracking Unit (RTU++), which can be flexibly plugged into other trackers, to score track proposals by capturing long-term information. And the experiments showed the generalization ability of RTU++ trained by simulated data in various scenarios. Under the specific limitations and difficulties implied by the surgical environment, only a few jobs deal directly with the application. Since the problem of surgical process identification is a multidisciplinary problem, we have decided to propose different related fields. Surgical phase recognition is similar to time action recognition. We start with a brief introduction to literatures on temporal action recognition. Then, we will focus on the internal approval of the operation.

Temporal action recognition

Gaidon et al. (2011, 2013) introduced temporally action recognition in untrimmed videos, focusing on limited actions such as “drinking and smoking” (Calder and Siegel, 2009) and “opening the door to sit down” (Laptev and Perez, 2007). Later, researchers worked on building large datasets, including complex action categories such as THUMOS (Mexaction2, 2013), as well as datasets focused on fine-grained actions (Sigurdsson et al., 2016a,b) or high-level semantics activities (Heilbron et al., 2015). Recently, deep learning methods have shown better performance in localizing action instances. Franke et al. (2013) presented a temporal action proposal system based on Long-Short Term Memory (LSTM); Yeung et al. (2018) provided the MultiTHUMOS dataset of each frame multi-label annotations, and a LSTM network is defined to

TABLE 1 List of phases in the dataset.

ID	Phase
P0	Trocar placement
P1	Preparation
P2	Calot triangle dissection
P3	Clipping and cutting
P4	Gallbladder dissection
P5	Gallbladder packaging
P6	Cleaning and coagulation
P7	Gallbladder retraction

model multiple input and output connections; Shou et al. (2016) introduced a 3D CNN framework (S-CNN) based on end-to-end segmentation, which is superior to other RNN-based methods by capturing spatio-temporal information simultaneously. However, S-CNN lacks the ability to accurately predict time resolution and localize the exact time boundary of an action instance. In Shou et al. (2017), they proposed a CDC network for precise temporal action localization of untrimmed video, which provides a new CDC filter that can simultaneously perform spatial down-sampling (for spatio-temporal semantic abstraction) and temporal up-sampling (for precise time positioning). In this paper, we will use the CDC network structure to recognize the surgical phase by transfer learning. Details are described in the next section. Yang et al. (2018) proposed a Frame Segmentation Network (FSN), which placed a temporal CNN on top of the 2D spatial CNNs, and can make dense predictions at frame-level for a video clip using both spatial and temporal context information.

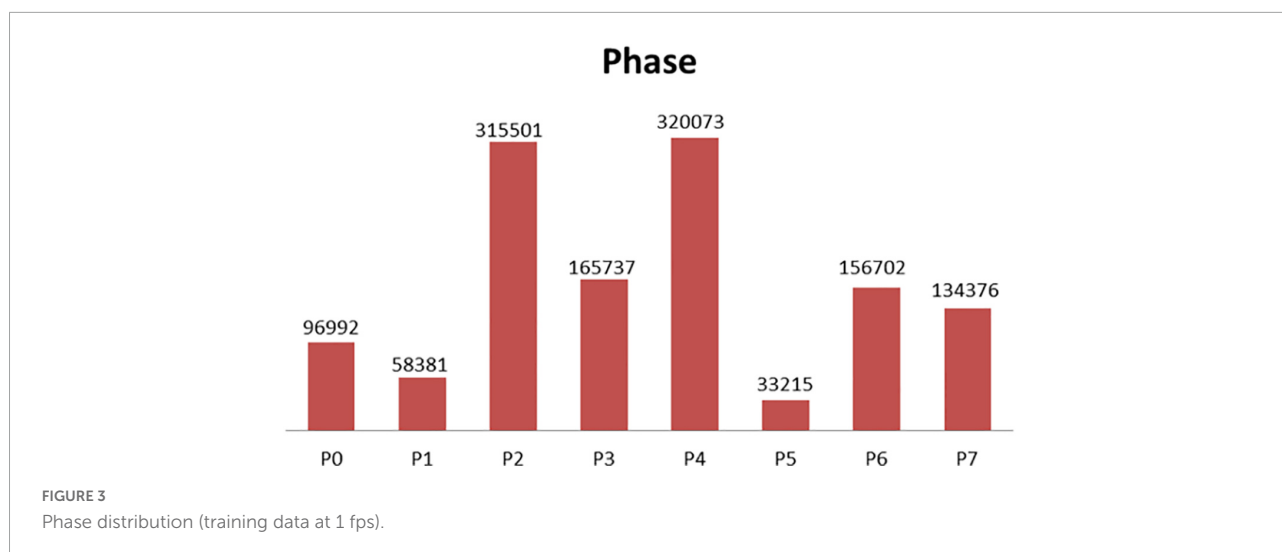
Surgical phase recognition

Mackenzie et al. (2001) were among the first to propose the creation of a process model. In Mackenzie et al. (2001), it is based on structured multi-level decomposition that describes the surgical action performed during surgery. In the same year, Jannin et al. (2001) also proposed a neural process model based on Uniform Mark-up Language decomposition. Subsequently, the concept of surgical workflow was introduced. Neumuth et al. (2006) proposed the concept of the general methodology described in the acquisition process from surgical intervention, clinical and technical analysis, and automatic processing of workflow schemes can drive a workflow management system as the future of OR process control. Klank et al. (2008) used the evolutionary reinforcement learning to classify the laparoscopic cholecystectomy into 6 stages for the first time, with an Acc rate of about 50%. Klank et al. (2008) presented a method that based on Hidden Markov Model (HMM) and dynamic time warping algorithm (DTW) to perform a dimensionality reduction on image features by using additional information about tool usage

for recognition of surgical workflow of laparoscopic video, the Acc of phase detection is 76.8%. Dergachyova et al. (2016) proposed a machine learning method. Specifically, they firstly described the input image by extracting the color, shape, and texture features of the image, and then they used several AdaBoost cascades for intermediate classification. Finally, a definite phase label is given by using the hidden semi-Markov Model. Based on visual features, the Acc of the model is close to 68%, and the Acc of fusion surgical instruments is close to 90%. The recent study in Dergachyova et al. (2016) is a method based on deep learning. The time smoothing convolution neural network and the classical HMM were used for phase recognition. The proposed network challenge is based on the residual network-200 pre-trained ImageNet, where the last layer is replaced by a new fully connected output layer, corresponding to 8 possible surgical phases. It was then fine-tuned on the M2CAI dataset using online data augmentation. The logarithmic probability output vector of the network was processed by temporal smoothing, and then passed to the HMM to correct possible classification errors for previously recognized frames. Twinanda et al. (2016) also proposed a method of deep learning based on pre-trained AlexNet, called PhaseNet, and they replaced the output layer and fine-tuned it using the M2CAI training dataset. At the second last layer of the PhaseNet, one-vs.-all linear SVM is obtained by using the image features extracted by CNN as input. Based on the Support Vector Machine classifier, the hierarchical HMM was introduced to reinforce the temporal constraint. The method was still based on two large datasets of laparoscopic cholecystectomy (Cholec 80 and EndoVis), which achieves better performance. The average Acc of offline analysis was highest, at 92.2% (Cholec80) and 86% (EndoVis), respectively. Shi et al. (2021) proposed a label-efficient Surgical workflow recognition method with a two-stage semi-supervised learning, named as SurgSSL which progressively leverages the inherent knowledge held in the unlabeled data to a larger extent. The SurgSSL method surpasses the state-of-the-art semi-supervised methods by a large margin.

Materials and methods

In this paper, we proposed a model for recognizing surgical workflow, as shown in Figure 1. Specifically, the top is an unsupervised time sorting task based on the CDC network, and the bottom is based on the top of the transfer supervised surgical phase classification task. The weights of the layers marked with a star can be passed. The first row shows the shape of the output data of each layer. First, the surgical video clip is fed into 3D ConvNets, and the temporal length is reduced from L to $L/8$. CDC6 has kernel size (4, 4, 4), Stride (2, 1, 1), padding (1, 0, 0), so the height and width are reduced to 1, while the temporal length increases from $L/8$ to $L/4$. CDC7 and CDC8 kernel size (4 1 1),

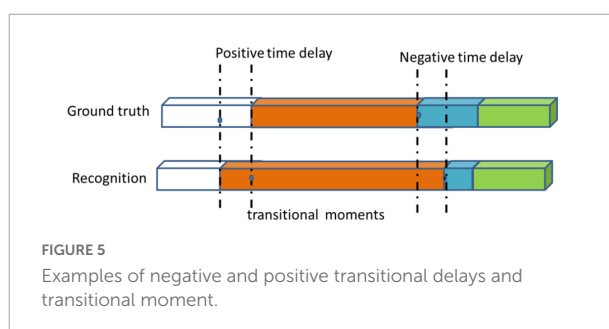
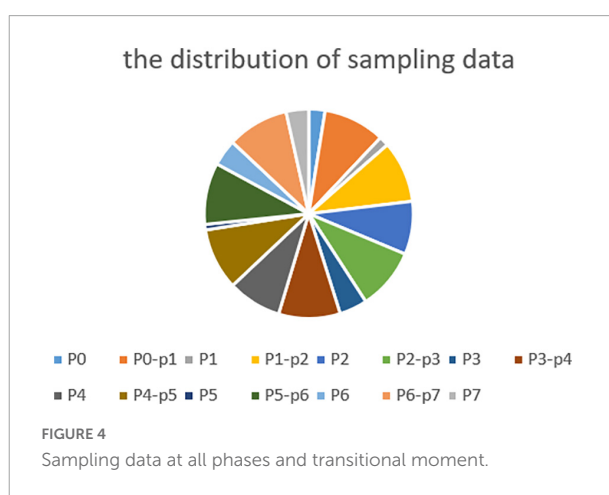


Step (2, 1, 1), padding (1, 0, 0), so CDC7 and CDC8 further perform up-sampling in time by a factor of 2, so the temporal length is back to L in the unsupervised temporal sorting task, sigmoid layer is added on top of CDC8 to determine whether is correct for the order of the given L frames. In the transfer supervised classification task, a frame-wise softmax layer is added on top of CDC8 to obtain confidence scores for every frame. Each channel stands for one class, obtaining confidence scores for every frame.

Unsupervised spatio-temporal context learning

In this section, we describe how to train the CDC network using unmarked video. We do this by addressing a task that requires the CDC to sort L given frames in the correct temporal order. For this, a large dataset from multiple surgical intervention is used. We assume that solving such a task requires CDC to learn to extract visual cues that describe the temporal flow of the surgical workflow.

The CDC (Shou et al., 2017) network is based on 3D convolution C3D network, which simultaneously carries out spatial convolution (for semantic abstraction) and temporal convolution (for frame-level resolution). It has a unique property in the spatio-temporal interactions between joint modeling and summarizing. The CDC network uses from conv1a to conv5b as the first part of the C3D network. For the remaining layers in the C3D, CDC keeps pool5 to perform max pooling in height and width by a factor of 2 but keeps the temporal length. The CDC sets the height and width of the network input as 112×112 . Given an input video segment with a temporal length L , the output data shape of the pool5 is (512, $L/8$, 4, 4). To maintain the original temporal resolution (frame level), the CDC makes up-sampling in time (back to L from



$L/8$) and down-sampling in space (from 4×4 to 1×1). More information is described in Shou et al. (2017).

Our CDC training tasks are shown in Figure 2: Given the same surgical video input for a video clip of temporal length L , what is the most relative order of L frames? That is, is the order of the given L frames correct? We uniformly sample L random frames from the video of the surgical intervention at the moment of transfer and enter them into our CDC. The

TABLE 2 ATD, TRR metrics for phase recognition.

Methods	ATD	TRR
Ours	[−15 s; 30 s]	6.0
Twinanda	[−23 s; 54 s]	3.8
Dergachyova	[−45 s; 70 s]	2.7

transfer moment is shown in **Figure 1**. The CDC must calculate the relative order of L frames in the original video. That is, determines whether the given L frame is in the correct order? That is, in the last layer of the network, we have two categories of L frames, the correct order is positive, otherwise it is negative. We assume that solving this task requires the CDC to extract visual cues related to the surgical process in order to understand the temporal flow of surgical intervention. At the same time, the learning of temporal information is carried out in this process.

The total loss is defined as:

$$L = -\sum_i label_i * \log(prob_i) + (1 - prob_i) * \log(1 - prob_i) \quad (1)$$

where $label_i$ is the ground truth for i -th segment, $prob_i$ is predictions for i -th segment.

When an unsupervised dataset is generated, data generation is primarily performed randomly at the time of conversion. Each phase is randomly sampled according to the ratio column, and the main sampling point is the transfer point. The specific sampling is related to the experimental dataset.

Knowledge transfer for recognition of surgical phase

The phase sequence indicating a surgical process encodes some form of abstract knowledge about a given procedure. The knowledge can be extracted and utilized to improve various operations on surgical process data, including analysis, recognition and prediction. It is particularly assumed that the knowledge gained from one procedure can improve the prediction of the surgical phase of another procedure. The knowledge involved may include dependencies between phases in a sequence, relationships between elements in an activity, and connections between individual elements of different activities. In view of the difficulty of formalizing the concealment of knowledge, the CDC network can extract features from time and space at the same time, so the CDC network is chosen as a method to extract and transfer knowledge.

Deep neural networks have an interesting property that enables networks to store extracted information in a distributed hierarchical manner. It means that the basic information that is more common for many areas stored separately from the features that describe the characteristics

of a particular domain. It also means that this information can be shared with other learning goal (e.g., other training task or area). In the deep model, the knowledge learned from the data is encoded into the weight matrix of the internal parameters of the representation layer. In order to establish the value of internal parameters, the domain containing a large number of training samples is first trained. Then, depending on the quantity and quality of data in the actual target domain, there are three transfer options. First, if the new data is close enough to the data used for training, and the task has not changed, we can use the same training model directly for the new data. The second option is to use the weights (in whole or in part) of the training model as the initialization of the new model. This applies where a reasonable amount of new data is available for training use. The third option, called fine-tuning, is typically used when the new domain contains only a small number of examples. It includes importing the trained weight matrix into the new model, but “freezes” some layers that usually contain more basic features during training. The weight setting of pre-training on other data is usually more optimized than random initialization. The network can benefit from what has been learned, thus, we should focus its “attention” on the specific characteristics of the new data. This section is based on the CDC time sorting network for knowledge transfer learning. Modify the final output layer of the CDC network to be L and classify each surgical step. In the transfer supervision classification task, the Softmax output is the vector of the K -value. Note that for the i -th class:

$$p_n^i[t] = \frac{e^{o_n^{(i)}[t]}}{\sum_{j=1}^K e^{o_n^{(j)}[t]}} \quad (2)$$

The total loss L is defined as:

$$L = \frac{1}{N} \sum_{n=1}^N \sum_{t=1}^L \left(-\log \left(p_n^{(z_n)}[t] \right) \right) \quad (3)$$

Where z_n is the ground truth class label for the n -th segment.

Experiment and result

Dataset and data sampling

The experiment in this paper is based on the M2CAI16-workflow dataset, which is available from <http://camma.u-strasbg.fr/m2cai2016/>. It contains videos of 41 cholecystectomy processes from the University Hospital of Strasbourg/IRCAD (Strasbourg, France) and Klinikum Rechts der Isar Hospital (Munich, Germany). The datasets are divided into two parts: the training subset (containing 27 videos) and the testing subset (14 videos). The videos are recorded

TABLE 3 Time delay standard scores metrics for phase recognition.

Methods	Scores			$d = 30$ s			$d = 60$ s		
	Acc	Rec	Pres	Acc	Rec	Pres	Acc	Rec	Pres
Ours	89.2	76.5	78.3	90.6	77.8	80.9	91.4	78.9	82.5
Twinanda	75.2	64.6	69.0	80.5	70.6	77.8	82.9	74.9	79.5
Dergachyova	68.6	60.9	64.1	72.1	65.3	66.2	76.6	71.4	78.1

Bold values indicate the optimal result in the algorithm comparison.

at 25 fps. All the frames are fully annotated with 8 defined phases: (1) trocarplacement, (2) preparation, (3) calot triangle dissection, (4) clipping and cutting, (5) gallbladder dissection, (6) galbladder packaging, (7) cleaning and coagulation, and (8) gallbladder retraction. The list of phases in the dataset is shown in **Table 1**. The distribution of the phases in dataset is shown in **Figure 3**.

In the case of a frame rate of 1, a total of 1.3 million frames are available. Depending on the distribution of the surgical phase, we randomly collected 250,000 surgical video clips from different surgical phases, 500,000 surgical video clips for the transition period, and 750,000 surgical video clips for unsupervised temporal learning. The sampling data for each stage and transition time is shown in **Figure 4**.

Comparison algorithms

We compared our method with several state-of-the-art method. **Dergachyova et al. (2016)** and **Twinanda et al. (2016)** are two of the methods submitted to the M2CAI 2016 challenge. CNN-biLSTM-CRF (**Yu et al., 2019**) is a semi-supervised method with 12 labeled vides and 15 unlabeled videos. The cnn-lstm-net and spatial-net are temporal and spatial models depicted in **Chen et al. (2018)**. In the CAE method (**Qi et al., 2020**), a convolutional auto-encoder network is trained first, and then surgical process segmentation is performed.

Metrics and result

As described in other literatures (**Chen et al., 2018**; **Qi et al., 2020**; **Shi et al., 2021**), the metrics includes standard accuracy (Acc), recall rate (Rec), precision (Pres), average conversion delay (ATD), and real transition ratio (TRR). Some applications do not require a frame-by-phase identification. They may tolerate a certain time delay, but have no fundamental impact on the assistance provided. We introduced the concept of a transition window that

TABLE 4 Comparison results with no time delay.

Methods	Rec	Pres
Dergachyova	60.9	64.1
Twinanda	64.6	69.0
CNN-biLSTM-CRF	69.9	74.5
Cnn-lstm-net	72.2	60.8
Spatial-net	72.9	73.4
CAE	68.3	72.7
Ours	76.5	78.3

Bold values indicate the optimal result in the algorithm comparison.

a time interval centered on a real transitional moment, at both ends, authorizing an acceptable delay d . If the time moment being checked is in the transition window and occurs because of a delay, it is considered true. In this experiment, we set up different delay time d to calculate the Acc, Rec, and Pres of the model. We called it a time delay standard score. ATD measures the latency generated during all conversions of all available interventions in order to make an average estimate of the delay (see **Figure 5**). The negative and positive delays are measured separately and used to define the range of values for the average transition delay. A negative delay indicates that the transition between phases is detected in a delayed manner with regards to the ground truth. Conversely, positive delay means that the system decides to switch phases prematurely before the actual transition, details in **Dergachyova (2018)**. The TRR Metric calculates the actual TRR detected between numbers. It is an indicator of system stability and reflects the robustness of the system, as systems with high TRR may have a lower tolerance for intrinsic changes in input data. This ratio also provides a simple and intuitive idea of how many incorrect transfer moments are detected with the number and actual number of transitional moments that they actually detect (see Equation 4).

$$TRR = \frac{s'}{s} \quad (4)$$

where the s is the real transfer moment, the s' is transfer moment detected by the model.

Based on the data collected randomly, we first carry out unsupervised temporal task learning, pre-training, and then use the transfer learning method to carry out phase supervision classification. The corresponding results are shown in **Tables 2, 3**.

As can be seen from the results in **Table 2**, our approach has the shortest transition delay [−15s; 30s]. As can be seen from the results in **Table 3**, the standard Acc, Rec, and Pres of our model reach 89.2, 76.5, and 78.3%, respectively. Based on these results, this is why our model improves Acc less than other usage time delay standard scores. Our approach is more suitable for applications that require rapid system response. However, it makes too many incorrect conversions between phases (6 times more than it should be). On the other hand, the Dergachyova method provides greater delays recognition, but less incorrect phase change peaks (TRR = 2.7). Compared with our method, its recognition is more consistent. The Twinanda method also has a lower TRR. This shows that our model is more suitable for online use, while the Twinanda method and the Dergachyova method are suitable for offline use. The results in **Table 3** show how to use the delay transition window to improve performance scores. This helps to make a clearer estimate of how close these methods are actually to clinical applications in specific applications. From the above analysis, it is also important that we do not use a single indicator to distinguish and objectively compare these surgical phases of the identification model. In **Table 4**, the experimental results of Rec and Pres with no time delay are compared. The results show that our method outperform the comparison methods.

Conclusion

The automatic recognition of the current surgical phase can provide the correct computer assistance at the right time, which is the basis of realizing the context-aware OR system. However, the lack of clinical data in this area is a well-known problem. This creates obstacles to the recognition and analysis of surgical workflow tasks that require significant amounts of data. In this paper, an unsupervised CDC network method is proposed, which simultaneously carries out spatial convolution (for semantic abstraction) and temporal convolution (for visual resolution) of surgical workflow frame sequences. Then through the transfer learning, the CDC network is fine-tuned to classify the operative stage. Based on M2CAI 2016 challenge dataset, experiments and comparisons have been made, and good results have been obtained. The transparency is a very important

attribute of the medical system. In this paper, we use a deep learning method has been criticized for the nature of its learning process that is poorly understood. This can cause distrust among doctors. In the future work, we want to visualize the learning processes of deep networks in order to understand exactly what they have learned.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: <https://mldta.com/dataset/m2cai-2016-challenge/>.

Author contributions

Y-WC and JZ: study concept and design. K-HZ, Y-WC, and PW: analysis and interpretation of data. Y-WC, Z-YH, and PW: technical support. Y-WC: obtain funding. Y-WC, K-HZ, and Z-YH: writing original manuscript. K-HZ and Y-WC: revision of manuscript. All authors contributed to the article and approved the submitted version.

Funding

This work was supported by the National Key R&D Program of China (No. 2018YFC0116704 to Y-WC) and the Youth Innovation Promotion Association of Chinese Academy of Sciences (No. 2020377 to Y-WC).

Acknowledgments

We thank all the people who participated in this study.

Conflict of interest

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

Publisher's note

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

References

- Abu-El-Haija, S., Kothari, N., and Lee, J. (2016). YouTube-8M: A large-scale video classification benchmark. *arXiv [preprint]*. *arXiv:1609.08675v1.
- Agarwal, S. K., Joshi, A., and Finin, T. (2006). *Context-Aware System to Create Electronic Medical Encounter Records*. UMBC, TR-CS-06-05.
- Bhatia, B., Oates, T., Xiao, Y., and Hu, P. (2007). "Real-time identification of operating room state from video," in *Proceedings of the 19th Conference on Innovative Applications of Artificial Intelligence (IAAI)*, 1761–1766.
- Calder, A., and Siegel, J. (2009). "Automatic annotation of human actions in video," in *Paper Presented at the IEEE International Conference on Computer Vision*.
- Chen, Y., Sun, Q., and Zhong, K. (2018). Semi-supervised spatio-temporal CNN for recognition of surgical workflow. *EURASIP J. Image Video Proc.* 2018:76. doi: 10.1186/s13640-018-0316-4
- Dai, W., Yang, Q., Xue, G. R., and Yu, Y. (2007). "Boosting for transfer learning," in *Paper Presented at the International Conference on Machine Learning*. doi: 10.1145/1273496.1273521
- Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., and Li, F. F. (2009). "ImageNet: A large-scale hierarchical image database," in *Paper Presented at the IEEE Conference on Computer Vision & Pattern Recognition*. doi: 10.1109/CVPR.2009.5206848
- Deng, S., Zhang, J., Wu, D., He, Y., Xie, X., and Wu, X. (2022). A quantitative risk assessment model for distribution cyber physical system under cyber attack. *IEEE Trans. Indust. Inform.* doi: 10.1109/TII.2022.3169456
- Dergachyova, O. (2018). *Knowledge-Based Support For Surgical Workflow Analysis And Recognition Ph. D. Thesis*. Rennes University.
- Dergachyova, O., Bouget, D., Huauilmé, A., Morandi, X., and Jannin, P. (2016). Automatic data-driven real-time segmentation and recognition of surgical workflow. *Int. J. Comput. Assist. Radiol. Surg.* 11, 1–9. doi: 10.1007/s11548-016-1371-x
- Fan, Q., Zhang, Z., and Huang, X. (2022). Parameter conjugate gradient with secant equation based elman neural network and its convergence analysis. *Adv. Theory Simulat.* 2022:2200047. doi: 10.1002/adts.202200047
- Fard, M. J., Pandya, A. K., Chinnam, R. B., Klein, M. D., and Ellis, R. D. (2016). Distance-based time series classification approach for task recognition with application in surgical robot autonomy. *International Journal of Medical Robotics & Computer Assisted Surgery Mrcas* 13:3. doi: 10.1002/rcs.1766
- Franke, S., Meixensberger, J., and Neumuth, T. (2013). Intervention time prediction from surgical low-level tasks. *J. Biomed. Inform.* 46, 152–159. doi: 10.1016/j.jbi.2012.10.002
- Gaidon, A., Harchaoui, Z., and Schmid, C. (2011). "Actom sequence models for efficient action detection," in *Paper Presented at the IEEE Conference on Computer Vision & Pattern Recognition*. doi: 10.1109/CVPR.2011.5995646
- Gaidon, A., Harchaoui, Z., and Schmid, C. (2013). Temporal localization of actions with actoms. *IEEE Trans. Pattern Anal. Mach. Intell.* 35, 2782–2795. doi: 10.1109/TPAMI.2013.65
- Garg, A., Adhikari, N., McDonald, H., Rosas Arellano, M., Devereaux, P. J., Beyene, J., et al. (2005). Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: A systematic review. *JAMA J. Am. Med. Assoc.* 293, 1223–1238. doi: 10.1001/jama.293.10.1223
- Glorot, X., Bordes, A., and Bengio, Y. (2011). "Domain adaptation for large-scale sentiment classification: A deep learning approach," in *Paper Presented at the International Conference on International Conference on Machine Learning*.
- Guédon, A. C., Paalvast, M., Meeuwse, F. C., Tax, D. M., van Dijke, A. P., Wauben, L. S., et al. (2016). 'It is time to prepare the next patient' real-time prediction of procedure duration in laparoscopic cholecystectomies. *J. Med. Syst.* 40:271. doi: 10.1007/s10916-016-0631-1
- Heilbron, F. C., Escorcia, V., Ghanem, B., and Niebles, J. C. (2015). "ActivityNet: a large-scale video benchmark for human activity understanding," in *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 961–970. doi: 10.1109/CVPR.2015.7298698
- Huang, J. T., Li, J., Dong, Y., Li, D., and Gong, Y. (2013). "Cross-language knowledge transfer using multilingual deep neural network with shared hidden layers," in *Paper Presented at the IEEE International Conference on Acoustics*. doi: 10.1109/ICASSP.2013.6639081
- Hübner, A., Hansen, C., Beuing, O., Skalej, M., and Preim, B. (2014). *Workflow Analysis for Interventional Neuroradiology Using Frequent Pattern Mining*. CURAC Medicine.
- Jannin, P., Raimbault, M., Morandi, X., Seigneuret, E., and Gibaud, B. (2001). Design of a neurosurgical procedure model for multimodal image-guided surgery. *Int. Cong.* 1230, 102–106. doi: 10.1016/S0531-5131(01)00025-5
- Kersten-Oertel, M., Jannin, P., and Collins, D. L. (2013). The state of the art of visualization in mixed reality image guided surgery. *Comput. Med. Imag. Graph.* 37, 98–112. doi: 10.1016/j.compmedimag.2013.01.009
- Klank, U., Padoy, N., Feussner, H., and Navab, N. (2008). Automatic feature generation in endoscopic images. *Int. J. Comput. Assist. Radiol. Surg.* 3, 331–339. doi: 10.1007/s11548-008-0223-8
- Krasin, I., Duerig, T., and Alldrin, N. (2017). *OpenImages: A Public Dataset for Large-Scale Multi-Label and Multi-Class Image Classification*. Available online at: <https://storage.googleapis.com/openimages/web/index.html> (accessed March 16, 2022).
- Kuznetsova, A., Rom, H., and Alldrin, N. (2020). The open images dataset V4: unified image classification, object detection, and visual relationship detection at scale. *arXiv [Preprint]*. doi: 10.1007/s11263-020-01316-z
- Laptev, I., and Perez, P. (2007). "Retrieving actions in movies," in *Paper Presented at the IEEE International Conference on Computer Vision*. doi: 10.1109/ICCV.2007.4409105
- Li, J., Xu, K., Chaudhuri, S., Yumer, E., Zhang, H., and Guibas, L. (2017). GRASS: Generative recursive autoencoders for shape structures. *ACM Trans. Graph.* 36, 1–14. doi: 10.1145/3072959.3073637
- Liu, F., Zhang, G., and Lu, J. (2020). Multi-source heterogeneous unsupervised domain adaptation via fuzzy-relation neural networks. *IEEE Trans. Fuzzy Syst.* 1:3018191. doi: 10.1109/TFUZZ.2020.3018191
- Liu, Y., Tian, J., and Hu, R. (2022). Improved feature point pair purification algorithm based on SIFT during endoscope image stitching. *Front. Neuror.* 2022:840594. doi: 10.3389/fnbot.2022.840594
- Macario, A. (2010). What does one minute of operating room time cost? *J. Clin. Anesth.* 22, 233–236. doi: 10.1016/j.jclinane.2010.02.003
- Mackenzie, L., Ibbotson, J. A., Cao, C. G. L., and Lomax, A. J. (2001). Hierarchical decomposition of laparoscopic surgery: A human factor approach to investigating the operating room environment. *Minimally Invasive Ther.* 10, 121–127. doi: 10.1080/136457001753192222
- Meng, Q., Lai, X., Yan, Z., Su, C., and Wu, M. (2021). Motion planning and adaptive neural tracking control of an uncertain two-link rigid-flexible manipulator with vibration amplitude constraint. *IEEE Trans. Neural Networks Learn. Syst.* 2021, 1–15. doi: 10.1109/TNNLS.2021.3054611
- Mexaction2 (2013). Available online at: <http://mexculture.cnam.fr/xwiki/bin/view/Datasets/Mex+action+dataset> (accessed March 5, 2022).
- Nessi, F., Beretta, E., Ferrigno, G., and De, M. E. (2015). "Recognition of user's activity for adaptive cooperative assistance in robotic surgery," in *Paper Presented at the International Conference of the IEEE Engineering in Medicine & Biology Society*. doi: 10.1109/EMBC.2015.7319582
- Neumuth, T., Strauß, G., Meixensberger, J., Lemke, H. U., and Burgert, O. (2006). "Acquisition of process descriptions from surgical interventions," in *Paper Presented at the Database & Expert Systems Applications, International Conference*. doi: 10.1007/11827405_59
- Peters, T. M. (2006). Image-guidance for surgical procedures. *Phys. Med. Biol.* 51:R505. doi: 10.1088/0031-9155/51/14/R01
- Qi, B. L., Zhong, K. H., and Chen, Y. W. (2020). Semi-supervised surgical video workflow recognition based on convolution neural network. *Comput. Sci.* 47, 172–175.
- Schumann, S., Bühligen, U., and Neumuth, T. (2015). Outcome quality assessment by surgical process compliance measures in laparoscopic surgery. *Artifi. Intell. Med.* 63, 85–90. doi: 10.1016/j.artmed.2014.10.008
- Shi, X., Jin, Y., Dou, Q., and Heng, P. (2021). Semi-supervised learning with progressive unlabeled data excavation for label-efficient surgical workflow recognition. *Med. Image Anal.* 73:102158. doi: 10.1016/j.media.2021.102158
- Shou, Z., Chan, J., Zareian, A., Miyazawa, K., and Chang, S. F. (2017). CDC: Convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos. *arXiv [preprint]*. *arXiv:1703.01515v2. doi: 10.1109/CVPR.2017.155
- Shou, Z., Wang, D., and Chang, S. F. (2016). "Temporal action localization in untrimmed videos via multi-stage CNNs," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1049–1058. doi: 10.1109/CVPR.2016.119

- Sigurdsson, G. A., Russakovsky, O., Farhadi, A., Laptev, I., and Gupta, A. (2016a). *Much Ado About Time: Exhaustive Annotation of Temporal Data*. *arXiv [Preprint]*. doi: 10.48550/arXiv.1607.07429
- Sigurdsson, G. A., Varol, G., Wang, X., Farhadi, A., Laptev, I., and Gupta, A. (2016b). "Hollywood in homes: Crowdsourcing data collection for activity understanding," in *Paper Presented at the European Conference on Computer Vision*. doi: 10.1007/978-3-319-46448-0_31
- Twinanda, A. P., Mutter, D., Marescaux, J., De Mathelin, M., and Padoy, N. (2016). Single- and multi-task architectures for tool presence detection challenge at M2CAI 2016. *arXiv [Preprint]*. doi: 10.48550/arXiv.1610.08851
- Wang, S., Sheng, H., Yang, D., Zhang, Y., Wu, Y., and Wang, S. (2022). Extendable multiple nodes recurrent tracking framework with RTU++. *IEEE Trans. Image Proc.* 2022:319206. doi: 10.1109/TIP.2022.3192706
- Wu, D., Zhang, P., He, Y., and Luo, X. (2022). A double-space and double-norm ensembled latent factor model for highly accurate web service QoS prediction. *IEEE Trans. Serv. Comput.* 2022:3178543. doi: 10.1109/TSC.2022.3178543
- Yang, K., Qiao, P., Wang, Q. (2018). "Frame segmentation networks for temporal action localization," in *Advances in Multimedia Information Processing – PCM 2018. Lecture Notes in Computer Science*, eds R. Hong, W. H. Cheng, T. Yamasaki, M. Wang, and C. W. Ngo (Cham: Springer). doi: 10.1007/978-3-030-00767-6_23
- Yeung, S., Russakovsky, O., Jin, N., Andriluka, M., Mori, G., and Li, F. F. (2018). Every moment counts: Dense detailed labeling of actions in complex videos. *Int. J. Comput. Vision* 126, 375–389. doi: 10.1007/s11263-017-1013-y
- YouTube-8M Dataset (2018). Available online at: <https://research.google.com/youtube8m/index.html> (accessed February 9, 2022).
- Yu, T., Mutter, D., Marescaux, J., and Padoy, N. (2019). "Learning from a tiny dataset of manual annotations: a teacher/student approach for surgical phase recognition," in *Proceeding of the International Conference on Information Processing in Computer-Assisted Interventions*.
- Zhang, M., Chen, Y., and Lin, J. (2021). A privacy-preserving optimization of neighborhood-based recommendation for medical-aided diagnosis and treatment. *IEEE Int. Things J.* 8, 10830–10842. doi: 10.1109/JIOT.2021.3051060
- Zhang, M., Chen, Y., and Susilo, W. (2020). PPO-CPQ: A privacy-preserving optimization of clinical pathway query for e-healthcare systems. *IEEE Int. Things J.* 7, 10660–10672. doi: 10.1109/JIOT.2020.3007518
- Zheng, W., Xun, Y., Wu, X., Deng, Z., Chen, X., and Sui, Y. (2021). A comparative study of class rebalancing methods for security bug report classification. *IEEE Trans. Reliabili.* 70, 1–13. doi: 10.1109/TR.2021.3118026
- Zhong, L., Fang, Z., Liu, F., Yuan, B., Zhang, G., and Lu, J. (2021). Bridging the theoretical bound and deep algorithms for open set domain adaptation. *IEEE Trans. Neural Networks Learn. Syst.* 2021, 1–15. doi: 10.1109/TNNLS.2021.3119965

Frontiers in Computational Neuroscience

Fosters interaction between theoretical and experimental neuroscience

Part of the world's most cited neuroscience series, this journal promotes theoretical modeling of brain function, building key communication between theoretical and experimental neuroscience.

Discover the latest Research Topics

[See more →](#)

Frontiers

Avenue du Tribunal-Fédéral 34
1005 Lausanne, Switzerland
frontiersin.org

Contact us

+41 (0)21 510 17 00
frontiersin.org/about/contact

