IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 28, 2019, accepted September 9, 2019, date of publication September 13, 2019,
date of current version September 25, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2940997

One-Shot Learning Hand Gesture Recognition Based on Lightweight 3D Convolutional Neural Networks for Portable Applications on Mobile Systems

ZHI LU[1], SHIYIN QIN[1], LIANWEI LI[1], DINGHAO ZHANG[1], KUANHONG XU[2], AND ZHONGYING HU[2]

[1]School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China
[2]Artificial Intelligence Research Department, Sony China Research Laboratory, Beijing 100028, China

Corresponding authors: Zhi Lu (by1603117@buaa.edu.cn) and Shiyin Qin (qsy@buaa.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61731001 and Grant U1435220, and in part by Sony.

ABSTRACT Though deep convolutional neural networks (CNNs) have made great breakthroughs in the field of vision-based gesture recognition, however it is challenging to deploy these high-performance networks to resource-constrained mobile platforms and acquire large numbers of labeled samples for deep training of CNNs. Furthermore, there are some application scenarios with only a few samples or even a single one for a new gesture class so that the recognition method based on CNNs cannot achieve satisfactory classification performance. In this paper, a well-designed lightweight network based on I3D with spatial-temporal separable 3D convolutions and Fire module is proposed as an effective tool for the extraction of discriminative features. Then some effective capacity by deep training of large samples from related categories can be transferred and utilized to enhance the learning ability of the proposed network instead of training from scratch. In this way, the implementation of one-shot learning hand gesture recognition (OSLHGR) is carried out by a rational decision with distance measure. Moreover, a kind of mechanism of discrimination evolution with innovation of new sample and voting integration based on multi-classifiers is established to improve the learning and classification performance of the proposed method. Finally, a series of experiments and tests on the IsoGD and Jester datasets are conducted to demonstrate the effectiveness of our improved lightweight I3D. Meanwhile, a specific dataset of gestures with variant angles and directions, BSG 2.0, and the ChaLearn gesture dataset (CGD) are used for the test of OSLHGR. The results on different experiment platforms verify and validate the performance advantages of satisfied classification and real-time response speed.

INDEX TERMS 3D convolutional neural networks, discrimination evolution, multimodal feature fusion, one-shot learning hand gesture recognition, similarity measure, lightweight I3D.

I. INTRODUCTION

In recent years, non-contact human-computer interaction (HCI) is becoming more and more popular and gradually changing people's interaction mode. Therefore, as an important interactive medium, gesture recognition technology has attracted extensive attention from both the academia and industry. In the last decade, many gesture recognition algorithms based on specific hand-crafted feature descriptors extracted from depth or RGB data were introduced. Unfortunately, handcrafted features cannot take all factors into account at the same time and cannot fully reveal the essential characteristics of objects. Fortunately, with the breakthrough of deep CNNs based methods, it have obtained the state-of-the-art performance in a variety of vision-based task, such as image classification [1], image segmentation [2], anomaly detection [3], face recognition [4], and human action/gesture recognition [5], etc.

With the development and evolution of deep CNN architecture over the years, the classification performance of the

The associate editor coordinating the review of this manuscript and approving it for publication was Donato Impedovo.

network has been significantly improved at a tremendous speed. However, two challenging issues have hindered the application of deep CNNs in portable mobile systems. One is the acquisition of large-scale datasets and the other is the limitation of computing resources. Generally, higher classification performance heavily depends on the fact that optimization of deep and high-capacity models requires many iterative updates across a large number of labeled training samples. Moreover, there are many unavoidable problems especially with limited computational budget in the application of novel network architectures in portable mobile systems, such as the large number of trainable parameters, the high computation complexity of model and the large storage space, etc.

In this paper, we proposed a new framework for OSLHGR based on lightweight 3D CNN for portable applications on mobile systems. With the release of Kinetics dataset [6], Carreira and Zisserman [7] proposed a new model, named Inflated 3D ConvNets (I3D), which has the ability to take advantage of pre-training on Kinetics and can achieve state-of-the-art performance on several benchmark datasets. First, motivated by this paper, we used I3D as a reference baseline model and combined the 3D convolution kernel decomposition technology to design a lightweight network for spatiotemporal feature learning without significant loss of performance. Then, the large-scale isolated gesture dataset (IsoGD) and Jester dataset were used to evaluate the classification performance of lightweight I3D. Furthermore, multimodal feature fusion based on canonical correlation analysis achieved the state-of-the-art performance compared with other approaches on the benchmark datasets. On the basic of the analysis, some useful knowledge from deep training with big datasets of relative objects can be transferred and utilized to initialize the target network trained on the collected dataset. In this way, it can reduce the risk of over-fitting and accelerate the convergence of network. Furthermore, the mechanism of discrimination evolution with innovation of new sample was proposed for OSLHGR based on the extraction of discriminative spatiotemporal features with lightweight I3D. To further improve the classification results, voting mechanism based on multiple nearest neighbor classifiers was used. Finally, we implemented the OSLHGR method with robust classification performance, faster response speed and smaller storage consumption.

The main contributions of our work can be summarized as follows:

(1) A lightweight I3D is proposed for learning and extracting spatiotemporal features of various data, leveraging the benefits of I3D.

(2) Multimodal feature fusion framework based on the proposed lightweight I3D with linear SVM classifier achieves a performance advantage on a par with the state-of-the-art methods on the IsoGD and Jester datasets.

(3) In order to evaluate the classification performance of the proposed method for OSLHGR, a new specific gesture dataset—BSG 2.0 has been built, which contains 36 predefined gesture categories and a set of disturbance gesture data.

(4) A new method based on discrimination evolution with innovation of new sample is proposed for OSLHGR. The experimental results on the CGD and BSG 2.0 datasets demonstrate that the proposed method achieves a good balance between model size, computational complexity and classification performance.

The remaining of this paper is organized as follows. In Section II, a short review of technical background and related works on gesture recognition is presented. In Section III, a new high performance classification method is proposed for OSLHGR with lightweight I3D toward portable applications. In Section IV, a series of experiments and test results are presented which demonstrate the effectiveness of our approach. In the last section, the work done has been summarized.

II. TECHNICAL BACKGROUND AND RELATED WORKS

Since the fundamental breakthrough of AlexNet research work, the classification accuracy of ImageNet has continued to improve by the novel network architectures, including VGGNet [8], GooLeNet [9], ResNet [10], DenseNet [11] and SE-Net [12]. For many vision-based recognition tasks, the performance of these networks can be superior to that of humans due to the emergence of large numbers of labeled data samples and the improvement of computing power. However, there are two challenging research issues that arise in the application of DNNs in portable mobile systems. One of the difficulties is the acquisition of large numbers of training samples in some special areas and the other is the high computational overhead of these deep CNNs along with the large storage requirements. Recently, there have been lots of exploratory work in terms of one-shot learning and network compression and acceleration.

A. HANDCRAFTED FEATURE-BASED ONE-SHOT LEARNING HAND GESTURE RECOGNITION

One-shot learning problem of object categorization was first proposed by Li *et al.* [13]. Until the Chalearn gesture challenge held in 2011 based on Chalearn gesture dataset [14] in which OSLHGR got the attention of researchers and a lot of relevant papers were published. Wu *et al.* [15] adopted extended motion-history-image (Extended-MHI) for extracting the RGB and depth videos features and then multi-view spectral embedding (MSE) algorithm was used for the fusion of extracted features. After that maximum correlation coefficient is used for classification. Another method proposed by Wan *et al.* in [16] for spatiotemporal feature representation is known as 3D enhanced motion scale-invariant feature transform (3D EMoSIFT), which fuses RGB-D data. It extracts richer video representation despite of having one training sample for each class and these features are scale and rotation invariant. Later, Goussies *et al.* [17] proposed a novel method for transfer learning which utilized decision forests to recognize gestures and characters. Furthermore, Konečný *et al.* extracted histograms of oriented gradients (HOG) and histogram of optical flow (HOF) [18] features from RGB and

depth images and used dynamic time warping (DTW) to recognize gestures. In [19], Wan *et al.* first adopted DTW algorithm to split continuous gestures into some isolated gestures. Then 3D EMoSIFT features are extracted for each isolated gesture. After that, the class-specific maximization of mutual information (CSMMI) is used for learning a compact and discriminative dictionary for each class. Furthermore, a novel feature, namely mixed feature around sparse keypoints (MFSK), has been proposed to calculate various descriptors, [e.g., 3D sparse motion scale-invariant feature transform (3D SMoSIFT), HOG, HOF and motion boundary histograms (MBH)] around keypoint volumes from RGB-D data. They produced very promising results under one-shot learning in Wan *et al.* [20].

B. DEEP NEURAL NETWORK-BASED ONE-SHOT LEARNING HAND GESTURE RECOGNITION

In practical applications, there are some cases with only one single sample for a new gesture class so that conventional recognition method cannot be qualified with a satisfactory classification performance. In this section, several typical one-shot learning methods based on DNNs are introduced. In [21], Koch *et al.* proposed a siamese neural network for one-shot learning image recognition. The experimental results outperformed all available baselines by a large margin on the Omniglot data set. Xu *et al.* [22] described a novel memory networks architecture to tackle few-shot learning problem on object recognition and the test results demonstrated that the proposed model with machine-labeled image annotations are very effective for object recognition on new categories. Recently, Kaiser *et al.* [23] presented a large-scale memory module for use in deep learning, which can be easily added to the supervised learning of neural network so as to achieve outstanding performance for one-shot learning recognition on the Omniglot dataset. Compared with the problem of image-based one-shot learning, we focus more on one-shot learning in video-based domain. In our previous research work [24], we proposed to train the convolutional network structure on a large dataset with different categories by deep training and then conducted continuous fine-tuning on a relatively small dataset. The experimental results show that the proposed method can classify new gestures effectively. Another related work of our research group, Li *et al.* [25] proposed an OSLHGR algorithm based on evolution of discrimination with successive memory. It also achieved high classification performance on the same dataset.

On the basis of these studies, we further explore the implementation of high performance OSLHGR based on lightweight 3D CNNs for portable applications on mobile systems.

C. FEASIBLE APPROACHES AND FRONTIER PROGRESSES ABOUT MODEL COMPRESSION AND ACCELERATION

With the great success of CNNs in the field of computer vision, the deployment of deep neural network models in portable mobile systems has become an acute application requirement. However, large memory consumption and dense computation seriously hinder the application of these deep models in mobile platforms. In this paper, the overarching goal of our research work is to design a lightweight model that has few parameters and less computational complexity while maintaining high classification performance in the case of small number of training samples. To solve these problems, several effective methods of model compression and acceleration have been proposed in recent years.

1) NETWORK PRUNING

With the popularity of deep learning, pruning methods have been widely studied in recent years. In [26], Denil *et al.* has been demonstrated that a large number of redundant parameters exist in some depth network models. Therefore, the pruning methods are used to remove unimportant parameters to compress CNN models. According to the granularity of pruning, the pruning methods can be categorized into fine-grained pruning and course-grained pruning. In fine-grained pruning, Hu *et al.* [27] utilized average percentage of zeros (APoZ) to measure the percentage of zero activation and neurons with high APoZ exceeding the threshold were pruned. The coarse-grained pruning is divided into five groups: vector-level pruning [28], kernel-level pruning [29], group-level pruning [30], channel-level pruning [31] and filter-level pruning [32], [33]. Usually, channel-level and filter-level pruning methods are widely used in model compression. Liu *et al.* [31] skillfully took the scale factor of batch normalization (BN) layer as the index to evaluate the importance of channels. In the training process, the insignificant channels were automatically identified and then removed, resulting in a thin and compact model. Luo *et al.* [33] proposed to convert the pruning of filters into an optimization problem. The unimportant filters were pruned based on statistic information computed from its next layer. Then the proposed method achieved 3.31 $\times$ FLOPs reduction and 16.63 $\times$ compression on VGG-16, with only 0.5% top-5 accuracy declination.

2) ACHITECTURE DESIGN OF EFFICIENT NETWORKS

Different from the compression of pre-trained models, another effective way for model compression and acceleration is to design more efficient but lightweight network architecture. In recent years, a large number of excellent network structures have emerged and gradually deployed in mobile devices. For example, Lin *et al.* [34] proposed a novel network structure, called network in network (NIN), to improve model discriminability for local areas. Two effective strategies for implementing lightweight models, 1 $\times$ 1 convolution and global average pooling (GAP), are widely used by many state-of-the-art CNN models like GoogLeNet and ResNet. In [35], the SqueezeNet achieved 50 $\times$ compression with AlexNet-level accuracy by using 1 $\times$ 1 convolution and branching structure. Since then, the depthwise separable convolutions and pointwise convolutions were proposed in MobileNet [36] to build lightweight neural networks.

MobileNet has almost the same accuracy as VGG-16 on the ImageNet while being 32 × smaller and 27 × less computation. Furthermore, in order to solve the problem of accuracy drop caused by excessive use of pointwise convolution in MobileNet, ShuffleNet [37] was proposed to utilize pointwise group convolution and channel shuffle to reduce the computational consumption while maintaining accuracy. For the slimming of 3D convolutional network structure, Xie *et al.* [38] presented a new network architecture, named separable 3D CNN (S3D), which is 1.5 × more computationally efficient than I3D by replacing 3D convolutions with spatiotemporal separable 3D convolutions.

D. TIME COMPLEXITY ANALYSIS TOWARD ARCHITECTURE DECOMPOSITION OF 3D CNN

The computational expressions of space and time complexity of standard 3D convolution are given by Eqs. (1) and (2). Specifically, all 3D convolutional kernels in the network are assumed to the cube structures. The total space complexity of all convolutional layers is:

$$Space \sim \mathcal{O}\Big(\sum_{l=1}^{D} k_l^3 \cdot C_{l\text{-}1} \cdot C_l\Big) \tag{1}$$

where D is the number of convolutional layers in network, and l is the index of convolution layer. k_l is the height/width of convolution kernel. $C_{l\text{-}1}$ is the number of input channels in the l-th convolution layer. C_l is the number of output channels of the l-th convolution layer.

Then, the time complexity of all convolutional layers is:

$$Time \sim \mathcal{O}\Big(\sum_{l=1}^{D} f_l \cdot M_l^2 \cdot k_l^3 \cdot C_{l\text{-}1} \cdot C_l\Big) \tag{2}$$

where M_l is the height/width of output feature map and f_l is the number of frames of the image sequence. It can be seen from Eq. (2) that the time complexity will decrease correspondingly while the space complexity decreases. Therefore, decomposing the 3D convolution kernel and reducing the number of input channels of the convolution layers are effective methods to reduce the complexity of network space.

III. THE FRAMEWORK OF OSLHGR BASED ON LIGHTWEIGHT I3D MODEL

In this section, we mainly focus on the design of the proposed lightweight I3D model, and based on that, the framework of OSLHGR is proposed.

A. IMPLEMENTATION OF LIGHTWEIGHT NETWORK WITH GUARANTEED COST BY OPTIMIZING COMPRESSION OF INFLATED 3D CONVNET

At present, the deep 3D CNNs are widely used for video-based recognition tasks and many more network structures have replaced the fully connected layers with a large number of parameters. Hence, reducing the complexity of convolutional layer is an effective way to realize a lightweight

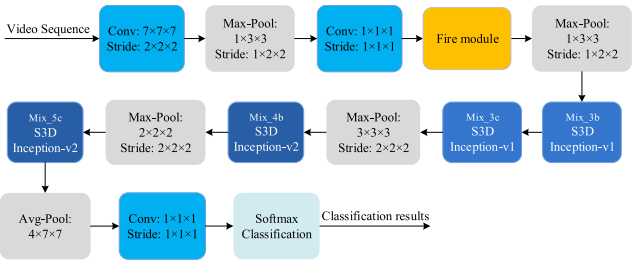


FIGURE 1. An illustration of the proposed lightweight I3D model for gesture recognition.

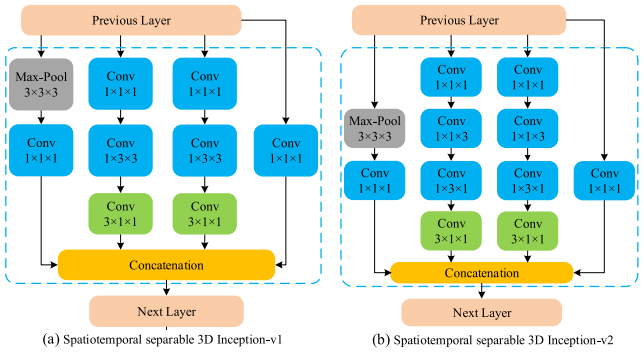


FIGURE 2. Two kinds of spatiotemporal separable 3D Inception.

network. In this section, three strategies are employed to reduce the computational cost of the network.

Strategy1: decrease the number of modules in network with the same output shape.

Strategy2: according to [35], reduce the number of input channels of large convolution kernels.

Strategy3: replace the standard 3D convolution with spatiotemporal separable 3D convolution. Besides, further factorization of spatial dimension is applied to the convolution layer which outputs smaller feature map [39].

Based on the design strategies of lightweight network, the I3D benchmark model was modified to reduce the storage consumption and improve processing speed. A detailed illustration of this new lightweight network architecture is shown in Fig. 1. The orange box is the Fire module. The blue boxes are spatiotemporal separable 3D inception-v1 modules (S3D Inception-v1), as shown in Fig. 2(a). The dark blue boxes are variants of S3D Inception-v1 modules (S3D Inception-v2), as shown in Fig. 2(b). Furthermore, we also compared the differences of each layer between the proposed lightweight I3D and the original I3D network structure, as illustrated in Table 1. On this basis, a multimodal feature fusion framework based on lightweight I3D is proposed to improve the performance of large-scale isolated gesture classification tasks. The implementation methods related to the fusion scheme can be referred in [40] and [41].

B. HIGH PERFORMANCE OSLHGR WITH LIGHTWEIGHT I3D TOWARD PORTABLE APPLICATIONS

As gesture recognition is becoming an important part of our daily life, it is requirement that gesture recognition system be

TABLE 1. Comparison of network structure between I3D and lightweight I3D.

layer name	output size	I3D	lightweight I3D
conv1	112×112×16	3D conv 7×7×7, stride 2×2×2	
maxpool1	56×56×16	3D max pool 1×3×3, stride 1×2×2	
conv2	56×56×16	3D conv 1×1×1, stride 1×1×1	
conv3	56×56×16	3D conv 3×3×3, stride 1×1×1	Fire module, reference [35]
maxpool2	28×28×16	3D max pool 1×3×3, stride 1×2×2	
Mixed_3b	28×28×16	Inception Module, reference [7]	As in Fig. 2(a)
Mixed_3c	28×28×16	same as above	As in Fig. 2(a)
maxpool3	14×14×8	3D max pool 3×3×3, stride 2×2×2	
Mixed_4b	14×14×8	Inception Module, reference [7]	As in Fig. 2(b)
Mixed_4c	14×14×8	same as above	-
Mixed_4d	14×14×8	same as above	-
Mixed_4e	14×14×8	same as above	-
Mixed_4f	14×14×8	same as above	-
maxpool4	7×7×4	3D max pool 2×2×2, stride 2×2×2	
Mixed_5b	7×7×4	Inception Module, reference [7]	-
Mixed_5c	7×7×4	same as above	As in Fig. 2(b)
	1×1×1	average pooling3D, 249-d fc, softmax	
params (M)	-	12.55	3.17
FLOPs (G)	-	55.58	17.63

tailored to the newly generated gesture vocabularies. However, it is impractical to supply a lot of training samples in many applications where collecting data and/or labeling them is tedious or expensive. In this context, based on the well-designed lightweight I3D network, a new method of gesture recognition using one-shot learning from a small sample set of gestures is proposed. In this paper, the proposed OSLHGR algorithm focuses only on the depth video data collected by SoftKinetic DS325.

1) PREPROCESSING OF INPUT GESTURE VIDEO DATA

In order to improve the classification performance of the model, all gesture samples must be preprocessed in the following three ways before input into the lightweight I3D network for deep training and feature extraction.

a: IMAGE PREPROCESSING

Due to the limitation of camera height, the depth values of four corners of the image are quite different from those of the middle area. Therefore, we replace each pixel with a fixed value whose depth value is higher than the threshold value in each corner.

b: ANGLE ROTATION

In order to reduce the difficulty of network training due to the difference in data distribution, all gesture videos are rotated in one direction during the network training and online testing.

c: DATA AUGMENTATION

A number of new gesture samples are generated from the original videos using the method proposed in [42], e.g. rotate 10° Clockwise and rotate 10° Counterclockwise.

2) FEATURE EXTRACTION BASED ON DEEP LEARNING AND CAPACITY TRANSFER OF LIGHTWEIGHT I3D

Generally, how to extract discriminative features from a single sample is the key to the success of OSLHGR. However, traditional handcrafted methods are often limited to extracting several typical features. Fortunately, the feature extraction methods based on CNNs have achieved a great breakthrough in video classification tasks, but the success of these tasks depends on the optimization of high-capacity network parameters with a lot of training samples. Therefore, there is a serious overfitting problem when only a small number of samples are used to train the network. Within this context, Yosinski *et al.* [43] provided a direct experiment evidence that the transfer learning method can effectively alleviate the risk of overfitting.

Inspired by the stunning success of transferred features from a well-trained baseline network to a target network described in [43], [44], a variant of the lightweight I3D (adding an additional layer of Mix_5b) which is slightly deeper than the lightweight I3D, is first fully trained on the training subset of IsoGD dataset and achieved effective recognition results on the validation subset. After that, the network parameters and model structure are transferred to the new target network excluding the final classification layer. Then, the parameters of target network are full fine-tuned on the new dataset to compensate for the differences (e.g., type of objects and acquisition conditions) between the source and target data. Finally, the network parameters of lightweight I3D are transferred and fixed. The remaining layers will be randomly initialized and fine-tuned on the new dataset again. On this basis, nonparametric classification methods, e.g., 1-nearest neighbor and maximum correlation coefficient, are utilized for OSLHGR because they are essentially attributed to the template matching and will not suffer from overfitting problems.

3) DISCRIMINATION EVOLUTION WITH INNOVATION OF NEW SAMPLE TOWARDS OSLHGR

In previous research [24], [25], we found that the quality of the first newly registered gesture sample in each class have a significant impact on the experimental results. Therefore,

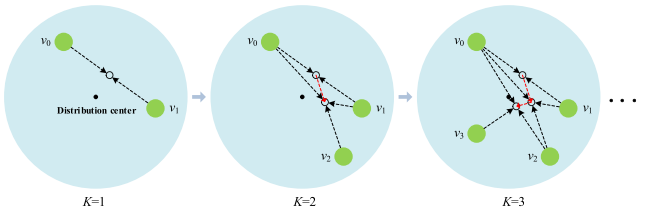


FIGURE 3. The discrimination evolution process of root sample with innovation of new samples. The green balls represent the feature points of gestures in space. The dotted circles indicate the central points of evolution.

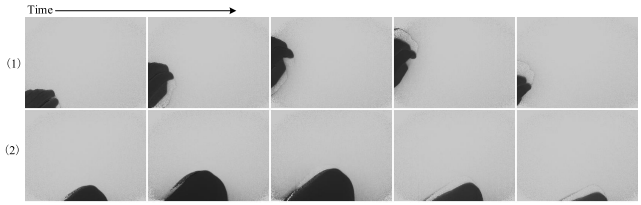


FIGURE 4. Examples of disturbances and maloperations.

a new method based on the discrimination evolution with innovation of new sample towards OSLHGR is proposed. We try to continuously adjust the position of this kind of gestures in high-dimensional space by utilizing the feature information of multiple test samples. In this way, the updated feature points can well depict the statistical information of this kind of gesture and avoid the performance degradation caused by abnormal sample.

For the convenience of expression, the original BSG 2.0 was first artificially divided into three subsets to estimate the performance of the proposed method. These subsets include: (a) auxiliary base subset $\boldsymbol{B}$ (part of categories selected from BSG 2.0) which contains training data and test data for fine-tuning the lightweight I3D. (b) Support subset $\boldsymbol{S}$ with only a single video clip per new class (root sample). (c) Evaluation subset $\boldsymbol{E}$ including positive and negative samples. Note that the support subset and evaluation subset share the same data labels. However, the subset $\boldsymbol{B}$ has its own private label, which is disjoint from the support set. Then, in order to avoid the influence of random disturbance and maloperation gestures, the negative samples in the evaluation subset $\boldsymbol{E}$ are first filtered out. After that, the root sample with the highest similarity to test sample will update its position information in space according to Eq. (3).

$$\bar{f}_K = \frac{f(v_0) + \sum_{j=1}^{K} f(v_j)}{K+1} = \frac{K \cdot \bar{f}_{K-1} + f(v_K)}{K+1} \tag{3}$$

where $f(\cdot)$ represents the feature extraction function implemented by lightweight I3D. $\bar{f}_K$ denotes the weighted average of the feature values of root sample and K test samples. v_0 is the root sample in support subset $\boldsymbol{S}$ and v_j is the test sample in evaluation subset. K denotes the number of times the root sample has been updated. The progressive evolutionary process of root sample is shown in Fig. 3.

4) REJECTION RULES AGAINST DISTURBANCE AND MALOPERATION

At present, most of the research work on object recognition mainly focuses on the classification of predefined classes. However, a practical gesture recognition system should not only do this, but also be able to effectively reject some disturbances and maloperation. Moreover, for some samples which cannot be effectively recognized, they are classified as unknown class. This is because it is more reasonable to be recognized as an unknown category than misclassification. As shown in Fig. 4, some representative samples of disturbances are presented. These disturbance samples are used to validate the rejection performance of OSLHGR, while the predefined classes are used for verifying the classification performance.

Unlike [24], [25], which only use one rejection rule, three similarity discrimination methods based on normalized Euclidean distance (*ED*), cosine distance (*CD*), and maximum correlation coefficient (*CC*) are used for classification in this paper. Formally, the three similarity calculation expressions are shown in Eqs. (4)-(6), respectively.

$$ED(\boldsymbol{V_1}, \boldsymbol{V_2}) = \frac{\|\boldsymbol{V_1} - \boldsymbol{V_2}\|}{\|\boldsymbol{V_1}\| + \|\boldsymbol{V_2}\|} = \frac{\sqrt{\sum_{i=1}^{n_d} (V_{1i} - V_{2i})^2}}{\sqrt{\sum_{i=1}^{n_d} V_{1i}^2} + \sqrt{\sum_{i=1}^{n_d} V_{2i}^2}} \tag{4}$$

$$CD(\boldsymbol{V_1}, \boldsymbol{V_2}) = \frac{\boldsymbol{V_1} \cdot \boldsymbol{V_2}}{\|\boldsymbol{V_1}\| * \|\boldsymbol{V_2}\|} = \frac{\sum_{i=1}^{n_d} (V_{1i} \times V_{2i})}{\sqrt{\sum_{i=1}^{n_d} V_{1i}^2} * \sqrt{\sum_{i=1}^{n_d} V_{2i}^2}} \tag{5}$$

$$CC(\boldsymbol{V_1}, \boldsymbol{V_2}) = \frac{Cov(\boldsymbol{V_1}, \boldsymbol{V_2})}{S(\boldsymbol{V_1}) * S(\boldsymbol{V_2})} \tag{6}$$

where $\boldsymbol{V_1} = \left[V_1^1, V_1^2, \ldots, V_1^{n_d}\right]^{\mathrm{T}}$ and $\boldsymbol{V_2} = \left[V_2^1, V_2^2, \ldots, V_2^{n_d}\right]^{\mathrm{T}}$ with the same dimension of n_d represent the feature vectors extracted from different gesture video clips. $Cov(\boldsymbol{V_1}, \boldsymbol{V_2})$ is the covariance of two feature vectors and $S(\boldsymbol{V_1})$, $S(\boldsymbol{V_2})$ are the standard deviation.

5) IMPLEMENTATION OF HIGH PERFORMANCE OSLHGR WITH LIGHTWEIGHT I3D

In this section, a new method based on discrimination evolution with innovation of new sample towards OSLHGR is proposed and the overall architecture is shown in Fig. 5. As can been seen, it consists of five modules: raw video data input, image preprocessing and angle rotation, efficient extraction of spatiotemporal features based on the pre-trained lightweight I3D model, nonparametric classification and comprehensive integration based on multi-classifiers. The input of network contains the root samples in the support set $\boldsymbol{S}$ and the test samples in the evaluation set $\boldsymbol{E}$. Then, in order to recognize gestures from variant directions and angles effectively, all gesture videos are unified into the same

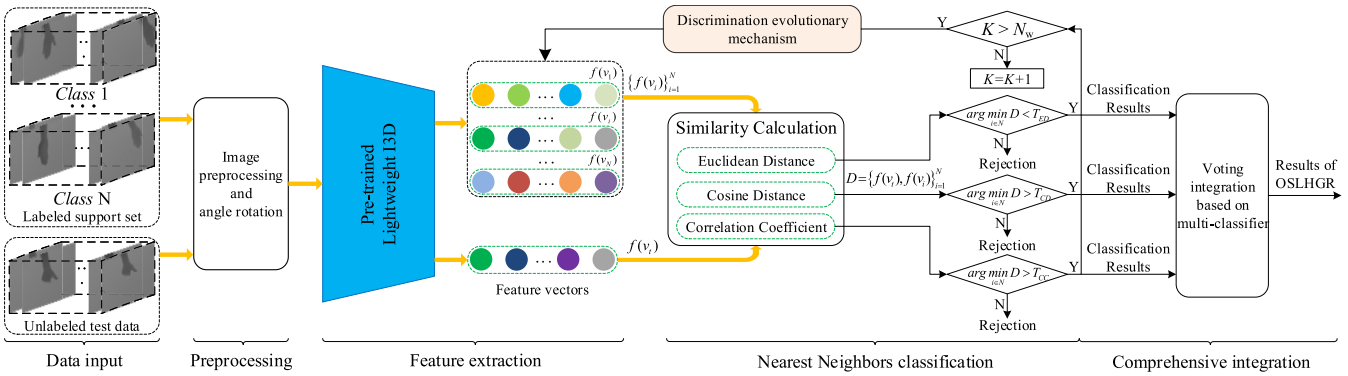


FIGURE 5. An overall architecture of the proposed OSLHGR method.

direction. After that, the proposed lightweight I3D model maps the raw gesture videos into high-dimensional feature space. On this basis, 1-nearest neighbors (NN) classifier based on similarity measure between different sample features and root sample updating mechanism based on the discrimination evolution are used to recognize unlabeled test gesture samples. Since three nonparametric classifiers are used to classify the test samples, voting integration based on multi-classifiers is established to fuse the classification results. Meanwhile, the implementation details of OSLHGR based on *CD* for similarity measurement are described in Algorithm 1. Specifically, the operator in line 11 of Algorithm 1 needs to be changed to '>' when *ED* is used to measure sample similarity.

IV. EXPERIMENTAL RESULTS AND COMPARATIVE ANALYSIS

In this section, two large-scale benchmark datasets and our collected gesture dataset are used to evaluate the classification performance of the proposed lightweight I3D. The proposed OSLHGR algorithm will be verified on the collected BSG 2.0 and CGD datasets respectively. First, the newly collected dataset will be described, including the collection environment settings, gesture categories and gesture data analysis. Then, experiment schemes and results will be fully demonstrated. Finally, the performance evaluation of experimental results will be discussed.

A. EXPERIMENT ENVIRONMENT AND DATA SETS

1) EXPERIMENT ENVIRONMENT TOWARDS PORTABLE APPLICATIONS

In this paper, we conducted experiments on two PCs: one is *Intel*® Core^{TM} i7-6800K CPU @ 3.40GHz × 12, 32GiB RAM and NVIDIA GeForce GTX 1080 GPU; the other is *Intel*® Xeon(R) CPU E5-2603 v2 @ 1.80GHz × 8, and Tesla K40c. The experiments of the lightweight I3D model training and feature extraction are processed under Keras framework on Linux Ubuntu 14.04 LTS. Canonical correlation analysis (CCA) based fusion is implemented with Matlab R2015b.

Algorithm 1 One-Shot Learning Hand Gesture Recognition With Lightweight I3D

Input: Auxiliary base subset: $\boldsymbol{B} = \{X, Y\}$. Support subset: $\boldsymbol{S} = \{(X_i, Y_i)\}_{i=1}^{N}$. Evaluation subset: $\boldsymbol{E} = \{(\hat{X}_j, \hat{Y}_j)\}_{j=1}^{n}$. Threshold: $T_{CD} \in (0, 1)$. Maximum number of waiting samples: N_w.

Output: Predict label: y.

1: **Initialize:** $D = \emptyset$, $K = 1$.
2: Image preprocessing, angle rotation and data augmentation based on the base subset $\boldsymbol{B}$.
3: Pre-training the lightweight I3D on the basis of step 2.
4: Feature extraction of support subset and evaluation subset, expressed as $F_s = \{f(v_i)\}_{i=1}^{N}$ and $F_e = \{f(\hat{v}_j)\}_{j=1}^{n}$.
5: **for** each $V_{2j} \in F_e$ **do**
6: **for** each $V_{1i} \in F_s$ **do**
7: $d_{\vec{ij}} = CD(V_{2j}, V_{1i})$ is calculated by Eq. (5);
8: $D \leftarrow D \cup d_{\vec{ij}}$;
9: **end for**
10: $D_{min} \leftarrow min(\mathrm{D})$;
11: **if** $D_{min} < T_{CD}$ **then**
12: Mark the j-th test sample as unknown class;
13: **else if** $K < N_w$ **then**
14: $y = i^* = arg\max_{i \in N}([D_{min}]_i)$;
15: $K = K + 1$;
16: **else**
17: $y = i^* = arg\max_{i \in N}([D_{min}]_i)$;
18: updating the root sample according to Eq. (3);
19: $K = K + 1$;
20: **end if**
21: **end for**
22: return y;

2) BENCHMARK DATASETS

To evaluate the performance of the proposed lightweight I3D, a series of experiments have been conducted on two publicly available datasets.

IsoGD [45] is a large-scale isolated gesture dataset built by Wan *et al.* At present, this benchmark dataset is widely used to evaluate the classification performance of new models

and to provide pre-training models for specific application scenarios. The dataset contains 47933 RGB + D gesture videos divided into 249 kinds of gestures performed by 21 different individuals. All videos are artificially divided into three mutually exclusive subsets: training set, validation set and testing set. It should be noted that the optical flow and saliency videos are generated by ours from RGB videos and no samples performed by the same person appear both in the training and validation/testing subsets.

Jester [46] is a large collection of densely-labeled video clips. Each clip contains a pre-defined hand gesture performed by a user in front of a laptop camera or webcam. It contains 148,092 RGB video files of 27 kinds of gestures. It is the largest isolated gesture dataset in which each category has more than 5,000 instances on average. Therefore, this dataset was used to train our networks from scratch.

CGD dataset was first proposed in [47], and then widely used to evaluate the performance of OSLHGR algorithm. In the videos, each performer is portrayed in front of a fixed Kinect^{TM} camera. The videos are a collection of a large dataset of gestures including RGB and depth samples of 50,000 gestures with image sizes 320 × 240 pixels at 10 frames per second. And a lot of gestures presented separately and only once for training classifier, and then different combinations of one or more of these gestures performed in a sequence in each video for recognizing. This benchmark dataset is recorded by 20 different users and grouped in 500 batches of 100 gestures. Each batch including 47 sequences of 1 to 5 gestures is drawn from various small gesture vocabularies of 8 to 12 unique gestures. Thus the CGD dataset is undoubtedly one of the most complex gesture datasets due to the wide variation of types of actions, environment, and position of the performers.

3) DATASET BUILDING FOR A SPECIFIC APPLICATION

In this section, we used the established experiment platform to collect depth video samples for the performance verification of the proposed OSLHGR algorithm, as shown in Fig. 6. Meanwhile, it also provides an experimentation environment for users to test the response speed and classification ability of the model online.

Unlike most other existing gesture datasets, the proposed OSLHGR algorithm needs to be robust to the rotation and translation of gestures. That is to say, gestures in any position and direction in the receptive field can be well recognized. Therefore, we added some restrictions on the angle and position of one-handed movements and the position of two-handed movements when collecting gesture samples, as shown in Fig. 7. The area surrounded by red lines is the effective space for gesture data collection and testing. We showed the performers a demo of each gesture at the beginning of the capture and then let them express the gesture naturally.

For the convenience of description, we divided all the gesture categories into two parts: data for network training and

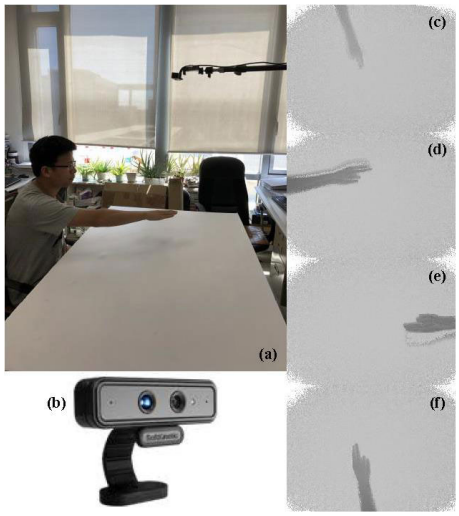


FIGURE 6. Experiment environment for data collection. (a) An illustration of experiments carried out for collecting samples. (b) DS325 is used for recording depth and RGB videos and capturing 320 × 240 and 640 × 480 pixels at 30fps, respectively. (c)-(f) respectively represent gesture samples collected from four directions.

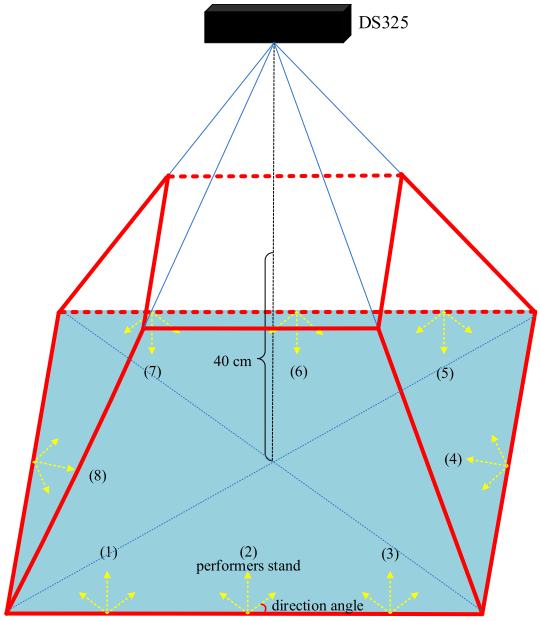


FIGURE 7. Demonstration of gesture data acquisition. (1)-(8) represent eight different data sampling points. At each point, three gesture videos in different directions (the direction of yellow arrow) are collected.

performance testing. For the first part, we collected 26 types of gestures. It contained a total number of 10720 original gesture videos performed indoors by different subjects, as shown in Fig. 8. Compared with [24], [25], more types of gestures and larger data samples are built in this paper. For the latter, we only collected a total of 1000 positive samples of 10 kinds of gestures and 100 negative samples, as illustrated in Fig. 9. It can be observed that each video in the test set contains only one hand. In the process of collecting data, the performers perform gestures using one or two hands while observing the real-time display interface. For real-world applications, our

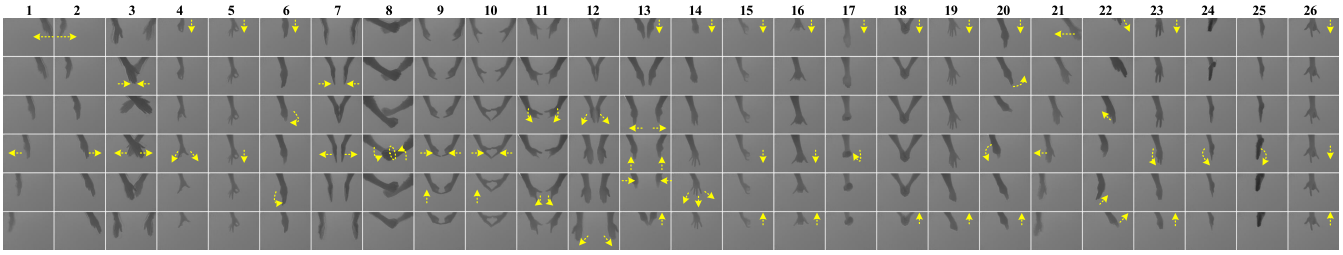


FIGURE 8. Twenty-six kinds of dynamic gestures collected by DS325 for network fine-tuning training. Each column separately represents a different gesture class and from top to bottom shows the gradual change process from the beginning to end frames of the nucleus phase. The yellow arrows denote the direction of movement.

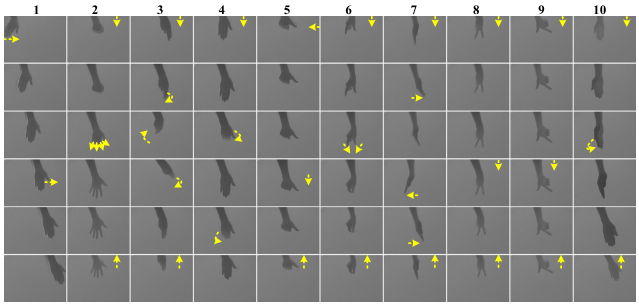


FIGURE 9. Ten kinds of dynamic gestures are used to verify the performance of OSLHGR algorithm.

TABLE 2. Variation statistics for three kinds of gesture datasets.

Datasets	Min frames	Max frames	Mean frames
IsoGD	9	405	41
Jester	12	70	35
BSG 2.0	9	119	36

gesture dataset is generated from performers with wearing full sleeves, watches or bracelets.

In order to fully understand the statistical characteristics of the collected data set for better design of system architecture and especially for the input of the proposed network, a detailed variation statistics of the three kinds of gesture datasets is conducted. In Table 2, the Min/Max number of frames and the Mean of each kind of gesture datasets are shown. It can be observed that the average number of frames is 36 in BSG 2.0 dataset. That is why we select 32 frames as a compromise between the simplicity of network parameter settings and the inclusion of more information.

To get a better visualization about the variation statistics for each category, we also plotted the above statistics for BSG 2.0 dataset which has 26 categories, as illustrated in Fig. 10. It can be seen that there is a large fluctuation in the length of video between different categories, even within the same class.

B. EXPERIMENT SCHEME AND PARAMETERS SETTING

In this section, we mainly focus on hyper-parameters settings of the proposed lightweight I3D, the way of network training and calculation method of evaluation index of OSLHGR.

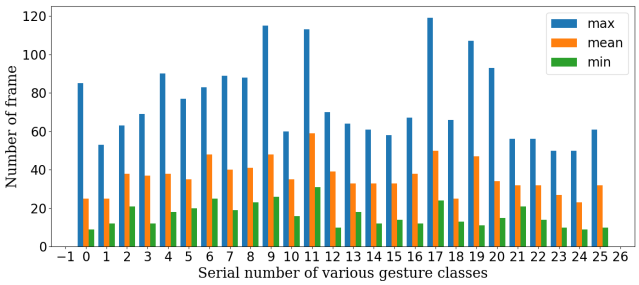


FIGURE 10. Statistical distribution of 26 gesture classes in BSG 2.0 dataset.

1) PARAMETERS SETTINGS

With the input dataset obtained, the lightweight I3D can be trained to extract the spatiotemporal features for classification. As can be seen from Figs. 1 and 2, most of $3 \times 3 \times 3$ convolution filters are replaced by two consecutive layers of spatial and temporal convolution for more efficient network structure design. Specifically, Fire module has the following parameter settings: $e_1 = e_2$ and $e_1 = 4s_1$. Similar as [7], the number of filters in each layer are set to 64, 64, 192, 256, 480, 512, 1024 and 1024 for an additional layer with a convolution kernel size of $1 \times 1 \times 1$ to project the feature into a lower dimension. Each convolution layer is followed by a BN layer and an activation layer. The first two 3D max-pooling layers (near the input layer) have kernel size $1 \times 3 \times 3$ with the intention of not merge the temporal information too early. Then a global average pooling layer with kernel size $4 \times 7 \times 7$ is performed. At last, the output of classification layer should correspond to the number of categories of gestures.

2) NETWORK TRAINING

For the proposed lightweight I3D, there is no pre-trained model to initialize network parameters. Therefore, the network must be trained from scratch. First, the weights of all layers are initialized from a zero-mean normal distribution with a standard deviation of $\sqrt{\frac{2}{n_i}}$, where n_i denotes the number of incoming nodes of one neuron. Then the strategy of exponentially decaying the learning rate is used. It is initialized to 0.005 and divided by 2 after every 10 epochs. We use SGD with the mini-batches of 8 samples because of

the limited memory size. After that, the length of each clip is 32 frames and the spatial size of the inputs are restricted to 224 × 224. At the same time, the number of input channels C is set to 1 to reduce the computation. Following, the L_2 regularization coefficient and the momentum term are set to 5×10^{-4} and 0.9, respectively. Finally, a total of 50 epochs and 70 epochs are required to train the IsoGD and Jester datasets, respectively.

Since no models which are pre-trained on other large-scale datasets can be utilized in the training phase, we utilize a cross-modality fine-tuning scheme [48] for IsoGD dataset. As follows, two different training schemes have been explored to evaluate the performance of the proposed lightweight I3D.

Scheme 1: RGB, optical flow, saliency and depth modalities based networks are trained from scratch on IsoGD dataset, respectively;

Scheme 2: Depth based deep CNN is well-trained from scratch and then sequentially fine-tuning the RGB, optical flow and saliency based networks based on pre-trained model of the depth modality. Then, the depth modality can be well trained based on the pre-trained model from RGB modality.

Based on the pretrained model, the proposed lightweight network can learn and extract discriminative spatiotemporal features. Therefore, 1000 positive samples and 100 negative samples in evaluation set $\boldsymbol{E}$ are used to estimate the performance of OSLHGR method. Moreover, to measure the similarity between the test samples and root samples, three measurement mechanisms are used to determine whether the test samples are rejected or not. Furthermore, to evaluate the performance of the proposed method under different rejection thresholds, these thresholds are sampled at intervals of 0.001 steps in the range of (0, 1). In addition, the maximum number of waiting samples N_w is manually adjusted based on experimental results.

To comprehensively evaluate the performance of OSLHGR, the *Precision* (P), *Recall* (R), F_1-*score* (F), and *Accuracy* (A) are computed for performance measurement of multi-class classification tasks. Detailed definitions of these variables have been systematically analyzed in [49].

C. EXPERIMENTAL RESULTS AND COMPARATIVE ANALYSIS

1) EXPERIMENTS TOWARDS LARGE-SCALE GESTURE CLASSIFICATION

In this section, a series of experiments and tests have been conducted on two large-scale benchmark datasets to evaluate the effectiveness of the proposed lightweight I3D model.

a: EFFECTIVENESS EVALUATION OF TRANSFER LEARNING ON IsoGD DATASET

The proposed lightweight I3D is designed originally, since there is no pre-trained model on large-scale dataset to initialize the network parameters. Therefore, it is difficult for the proposed network to converge to the optimal value. However,

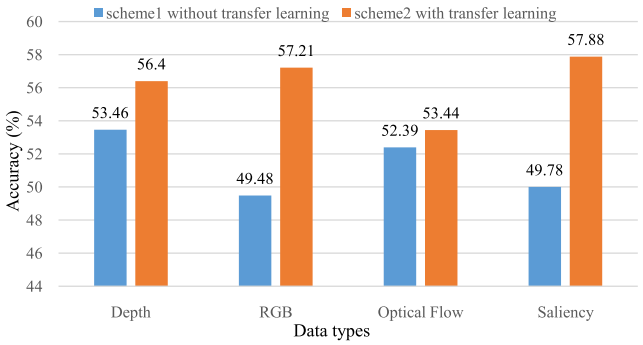


FIGURE 11. Comparison of the results of two different schemes on the validation subset of IsoGD.

transfer learning is a practical technique to prevent overfitting and has shown excellent performance in many applications. In this paper, four modalities of depth, RGB, optical flow and saliency stream based network are trained from scratch on IsoGD dataset, respectively. The test results are shown by the blue bar in Fig. 11. After that, the RGB, optical flow and saliency stream based network are fine-tune based on the pre-trained model of depth modality and the classification results are denoted by the orange bar. By comparing these two different training schemes, we observed that fine-tuning other modalities based network based on the pre-trained model of depth modality can improve the performance significantly.

b: COMPARISON WITH THE STATE-OF-THE-ART METHODS ON IsoGD DATASET

Table 3 presents a comparison of the classification results between our proposed multimodal fusion framework and other state-of-the-art methods on the validation set of IsoGD. As can be seen, the CNN-based methods outperform the handcrafted feature based method like [45] by a large margin. It can also be observed that our proposed multimodal fusion method based on lightweight I3D can achieve higher classification accuracy. Specifically, the classification results of our reproduced I3D model using the same training settings as our method (i.e. trained from scratch) are slightly lower than our lightweight I3D. Furthermore, we compared the classification results of S3D and lightweight I3D model trained from scratch on the IsoGD dataset. It can be seen that the classification accuracy of the other two modals is slightly lower than that of S3D except for the classification result on RGB video data. This shows that S3D has certain performance advantages compared with the proposed lightweight I3D in classification accuracy. However, comparing the last column in the table, our proposed lightweight I3D model is much lighter. This further indicates that our proposed network is more suitable for portable application platforms.

c: RESULTS ON JESTER DATASET

The classification results of the proposed lightweight I3D on the Jester dataset is shown in Table 4. As can be seen, higher classification results can be achieved using only single mode

TABLE 3. Comparison of the proposed method and other methods on the validation set of IsoGD.

Methods	Accuracy (%)	Parameters
MFSK [45]	18.65	-
Pyramidal C3D (RGB + Averaging Fusion) [50]	36.58	78.07M
Pyramidal C3D (Depth + Averaging Fusion) [50]	38.00	
Pyramidal C3D (Depth + RGB + Averaging Fusion) [50]	45.02	
C3D (RGB only) [51]	37.3	78.07M
C3D (Depth only) [51]	40.5	
C3D (Depth + RGB) [51]	49.2	
3DCNN + ConvLSTM + SPP (RGB only) [48]	43.88	16.96M
3DCNN + ConvLSTM + SPP (Depth only) [48]	44.66	
3DCNN + ConvLSTM + SPP (RGB + Depth) [48]	51.02	
3DCNN + ConvLSTM + 2DCNN (RGB only) [52]	51.31	31.87M
3DCNN + ConvLSTM + 2DCNN (Depth only) [52]	49.81	
3DCNN + ConvLSTM + 2DCNN (Flow only) [52]	45.30	
3DCNN + ConvLSTM + 2DCNN (RGB + Depth + Flow) [52]	58.65	
ResC3D (RGB only) [41]	45.07	38.58M
ResC3D (Depth only) [41]	48.44	
ResC3D (Flow only) [41]	44.45	
ResC3D (Depth + RGB + Flow) + CCA + SVM [41]	64.11	
ResC3D (Depth + RGB + Flow) + CCA + TSN + SVM [41]	64.40	
I3D (RGB only, from scratch, reproduced) [7]	38.54	12.55M
I3D (Depth only, from scratch, reproduced) [7]	52.23	
I3D (Flow only, from scratch, reproduced) [7]	52.90	
S3D (RGB only, from scratch, reproduced) [38]	49.31	8.17M
S3D (Depth only, from scratch, reproduced) [38]	56.37	
S3D (Flow only, from scratch, reproduced) [38]	53.35	
Lightweight I3D (RGB only, from scratch)	49.48	3.17M
Lightweight I3D (Depth only, from scratch)	53.46	
Lightweight I3D (Flow only, from scratch)	52.39	
Lightweight I3D (Depth + RGB) + CCA + SVM	63.30	
Lightweight I3D (Depth + RGB + Flow) + CCA + SVM	64.73	
Lightweight I3D (Depth + RGB + Flow + Saliency) + CCA + SVM	**65.13**	

TABLE 4. Comparison with state-of-the-art methods on the validation of Jester dataset.

Methods	Accuracy (%)
Res3D + ConvLSTM + MobileNet [53]	95.13
MultiScale TRN [54]	93.70
MultiScale TRN (10-crop) [54]	**95.31**
Lightweight I3D (Flow only)	90.88
Lightweight I3D (RGB only)	94.62
Lightweight I3D (RGB + Flow) + CCA + SVM	94.79

RGB or flow data. Furthermore, the fusion of multimodal features can further improve the classification performance. Therefore, a series of experimental results on two benchmark datasets fully demonstrate the high performance advantages of our proposed lightweight I3D network.

2) EXPERIMENTS TOWARDS CLASSIFICATION PERFORMANCE OF OSLHGR

In this section, a series of experiments have been conducted on the collected BSG 2.0 and the CGD datasets to demonstrate the advantages of our proposed OSLHGR algorithm.

a: EXPERIMENTS ON BSG 2.0 DATASET

The experimental results of OSLHGR proposed in Algorithm 1 are shown in Table 5. we use three different similarity measurement methods to classify the test samples under two conditions of discrimination evolution and non-discrimination evolution. Thereinto, F_{max} denotes the maximum value of F_1-*score* in the process of change threshold T_{CD}. In particular, we calculated the other three evaluation indicators separately under the condition that P is equal to 100% and F_1-*score* is maximum. Furthermore, *ED* and *CC* are respectively used to replace *CD* in Algorithm 1 to measure the similarity between samples and the classification results are shown in the last two columns. As for the non-discrimination evolution, the gesture classification is performed by measuring the similarity between samples, and the information of root samples is not updated. The experimental results under three different similarity measures are shown in the first three columns. It is observed that the discrimination evolution method based on *CD* for similarity measure achieves the best classification result.

Moreover, we also explored the influence of initializing lightweight I3D network with different pre-trained model on the classification results. The comparison results are shown in Fig. 12. Thereinto, three solid lines represent the experiment results obtained by the method proposed in this paper and the three dashed lines indicate the experiment results of the pre-trained model of lightweight I3D on IsoGD dataset directly used to initialize the parameters of the network when fine-tuning on BSG 2.0 dataset. It can be seen that the solid red line is significantly higher than the dotted red line when P is equal to 100%. This means that more test samples are correctly classified under the same threshold. A practical classification system is expected to have both high P and R. Therefore, this also shows that initializing the target network

TABLE 5. Comparative results of different similarity measures under the mechanism of discrimination evolution and non-discrimination evolution.

Evaluation index			Non-discrimination evolution			Discrimination evolution		
			CD	*ED*	*CC*	*CD*	*ED*	*CC*
Results of OSLHGR	P=100 (%)	R (%)	56.40	41.90	38.60	**90.20**	85.40	89.00
		F (%)	72.12	59.06	55.70	**94.85**	92.13	94.18
		A (%)	60.36	47.18	44.18	**91.09**	86.73	90.00
	F_{max} (%)	F (%)	93.87	93.81	92.76	**99.10**	99.05	98.75
		P (%)	94.25	93.71	94.59	**99.50**	99.10	98.80
		R (%)	93.50	93.90	91.00	98.70	**99.00**	98.70
		A (%)	93.64	94.09	91.45	98.82	**99.00**	98.64

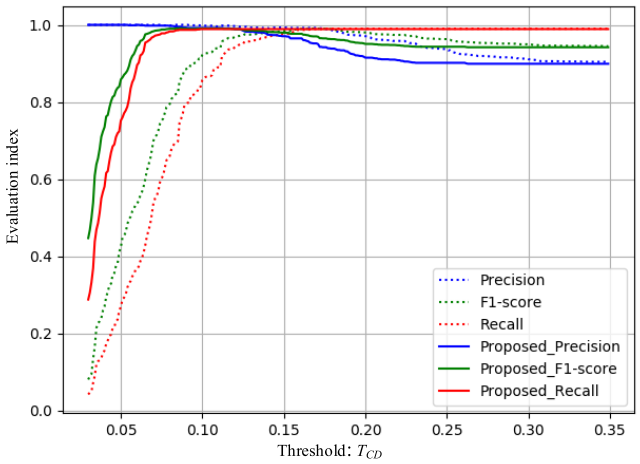


FIGURE 12. Comparison of the results of different network initialization methods.

with the pre-trained model on a larger network is beneficial to improve the performance of the model.

In this figure, the threshold T_{CD} is sampled at intervals of 0.001 steps in the range of (0, 0.35). During the change of T_{CD}, we also drew the confusion matrix of classification results and calculated the values of P, R, and F, respectively. Then, we will retain the confusion matrix with a maximum R at P equal to 100%, and retain the confusion matrix when F gets the maximum value. In both cases, the corresponding T_{CD} is the optimal threshold for the final selection. As we know, the smaller the distance, the greater the similarity between samples when the *ED* is used to measure the similarity of samples. However, the *CD* and *CC* are exactly the opposite. Therefore, in order to make the plotted curve having the same variation trend, the calculation results of *CD* and *CC* are subtracted by 1, respectively. Thus, we only need to replace the expression of the similarity between samples in Algorithm 1.

In addition, we conducted comparative experiments with the deep neural network-based OSLHGR methods in [24] and [25] on the BSG 2.0 dataset. The experimental results are shown in Table 6. Compared with the other two methods, the proposed method achieves better classification performance on our collected dataset. Meanwhile, the proposed network is more portable and efficient, which can greatly reduce the training time of feature extraction network.

TABLE 6. Comparison with the state-of-the-art deep neural network-based OSLHGR methods on the BSG 2.0 dataset.

Methods	P (%)	R (%)	F (%)	A (%)	Params (M)
Lu *et al.* [24]	73.98	77.90	75.89	74.18	28.69
Li *et al.* [25]	78.92	74.90	76.86	75.36	31.90
Our method	99.45	99.54	99.49	99.00	2.94

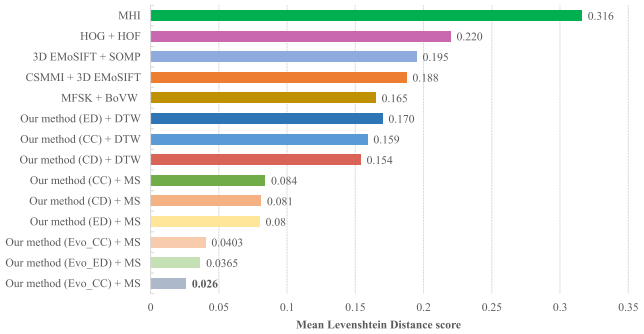


FIGURE 13. Performance comparison with state-of-the-art OSLHGR approaches on the development batches (devel01-devel20) of CGD dataset.

b: EXPERIMENTS ON CGD DATASET

Unlike the test data we collected, each test video in CGD dataset contains one or more action instances. Therefore, the first step is to segment all the different gestures accurately from the gesture sequences. At present, there are a number of research works dedicated to finding the boundaries of the temporal segmentation [55]–[57]. In this paper, automatic segmentation and manual segmentation are respectively used to obtain isolated gesture sequences. For automatic segmentation, the code of DTW [20] released by the organizers of the Chalearn gesture challenge is used to perform the segmentation of continuous gestures. For the manual segmentation, the temporal segment positions provided with the ChaLearn gesture dataset by the developers are utilized for separating test videos which contains multiple actions in devel01-20 batches.

After that, the temporal segmentation is done if multiple gestures exist. Then we will train a good feature extraction network according to the data partition method in BSG 2.0 before extracting features from samples in each development batch. To avoid the same gesture classes appearing in both the IsoGD dataset and each development batch, we first classify the training samples in each development

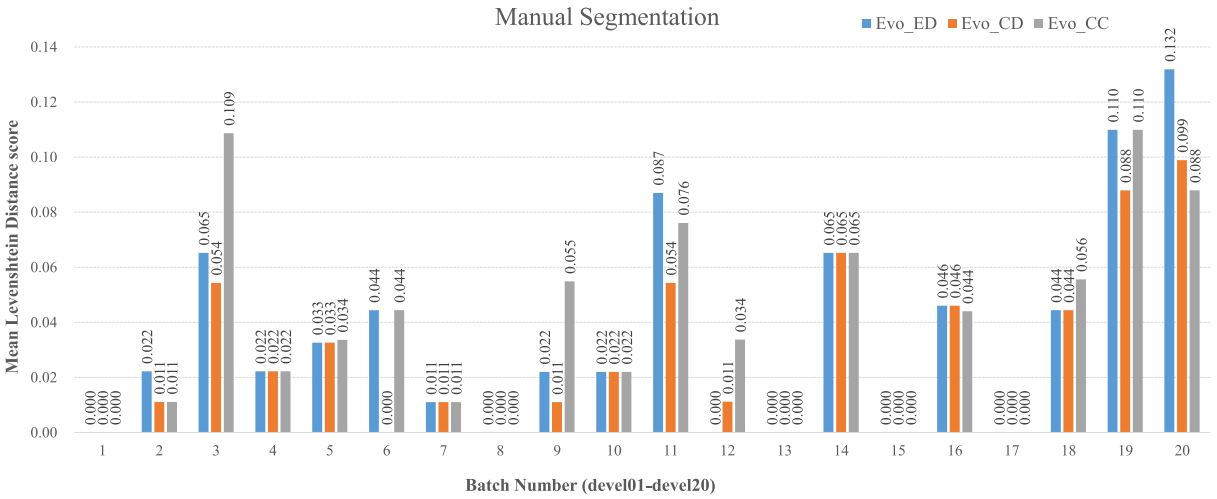


FIGURE 14. The MLD scores of our proposed method under the discrimination evolution mechanism on sub-batches devel01-devel20.

batch using a pretrained model trained on all IsoGD data samples. Then, the corresponding gesture categories in the IsoGD dataset are removed according to the predicted results. Moreover, we further manually confirm the reliability of the prediction results to ensure that there are no overlapping classes. After that, the remaining gesture classes in the IsoGD dataset are used to train lightweight network proposed in this paper. And we have a pretrained model for each development batch. Therefore, features are extracted from gesture samples in each development batch. At last, nonparametric classifier is used to classify the test sample in each development batch by calculating the distance between different samples.

To compare with other methods on the CGD dataset, the Levenshtein Distance (LD) measure, also known as edit distance, is employed. LD is the number of deletions, insertions, or substitutions required matching an array with another. For each unlabeled video, the distance LD(T_r, P_r) was computed, where T_r is the true vector of labels, and P_r is our predicted vector of labels. For the sake of comparison, the Mean Levenshtein Distance (MLD), which was computed as a sum of LD divided by the total number of true gestures performed in the sub-batch and multiplied by 100. It is obvious that the higher the value of MLD, the more is the number of misclassifications. Therefore, we expect the value of MLD to be as small as possible.

Initially, the performance of our proposed OSLHGR method is compared with traditional handcrafted feature-based methods proposed in recent years. The results are shown in Fig. 13. In this figure, DTW indicates automatic segmentation of continuous gesture video in the test data. The MS represents the result of manual segmentation provided by the developers. It can be seen that MLD scores of the proposed method are smaller than that of other methods either in the case of automatic segmentation or manual segmentation. Compared with current state-of-the-art MFSK [20], the MLD score decreased by 0.011. This demonstrates that the spatiotemporal features extracted from deep CNNs are more discriminative. Meanwhile, the MLD under manual segmentation is significantly lower than the result of automatic segmentation. This shows that how improving the accuracy of temporal segmentation is the key to reducing MLD. Furthermore, the classification results under discrimination evolution mechanism are significantly better than those under non-discrimination evolution mechanism. It further illustrates the superiority of the proposed method in this paper.

In Fig. 13, the MLD scores are computed at the batch level. To further demonstrate the differences between development batches, the MLD scores of each sub-batch of the development batches are calculated separately using our proposed method. Meanwhile, we also compared the performance obtained in batches 01-20 for development data when using manually segmented gestures and the automatic segmentation approach. The classification results obtained in terms of percentage of MLD are displayed in Figs. 14 and 15 on each of the first 20 development batches. In Fig. 14, gesture classification under discrimination evolution is performed based on *ED*, *CD*, and *CC* based classifier and the results are compared respectively. It can be seen from the figure that the results vary widely among different batches. Meanwhile, we can also observe that most of the sub-batches achieve superior recognition results. Especially on devel01, devel08, devel13, devel15 and devel17, all the test videos are correctly recognized under specific classification criteria. However, compared to the manual segmentation method, the performance of DTW-based automatic segmentation method is worse, as shown in Fig. 15. We can also observe that most of the sub-batches have lower MLD scores except devel11 and devel14. By looking at video sequence after DTW segmentation, we find that most gestures are incomplete or single video contains multiple gestures. Therefore, the values of these two groups of MLD are relatively large.

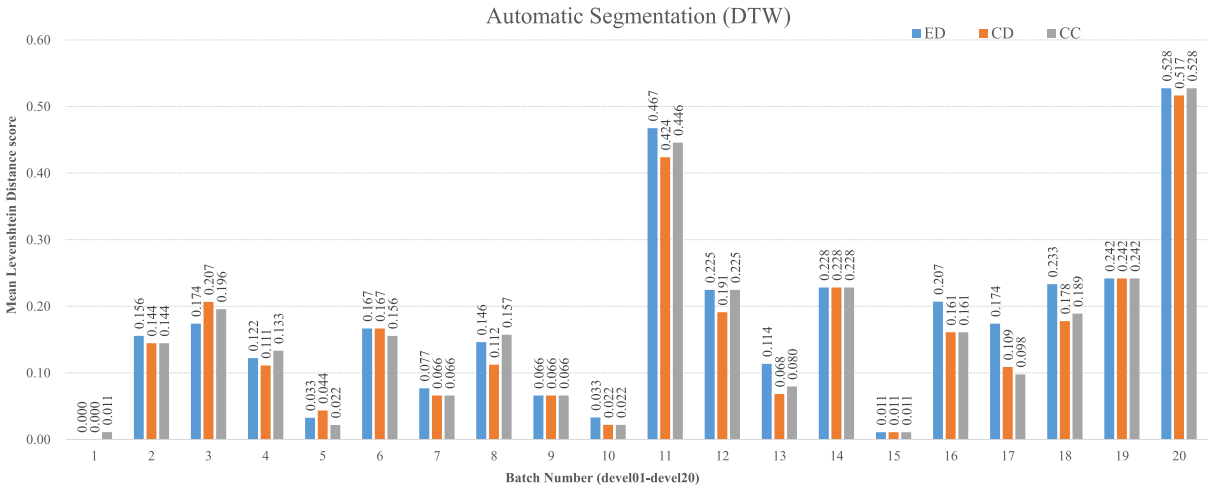


FIGURE 15. The MLD scores of our proposed method under the non-discrimination evolution mechanism on sub-batches devel01-devel20.

TABLE 7. Comparison of our proposed model to the original I3D and S3D with different evaluation criteria.

Model	P (%)	R (%)	F_{max} (%)	A (%)	Model Size (MB)	Parameters (M)	FLOPs (G)	Computation time (ms/f)	
								Tesla K40c	GTX 1080
I3D	98.40	98.20	98.30	98.09	49.60	12.28	46.76	15.43	5.04
S3D	98.90	98.80	98.84	98.27	32.5	7.94	33.44	19.50	7.64
Lightweight I3D	**99.45**	**99.54**	**99.49**	**99.00**	12.12	2.94	17.63	9.82	4.23

Furthermore, we also conducted experiments comparing with the deep neural network-based one-shot learning hand gesture recognition methods [24], [25] on the CGD dataset to verify the effectiveness of our proposed method. Due to the limitation of computing resources, we only conducted comparison experiments on devel01-05 sub-batches of CGD dataset respectively in accordance with the implementation method in the original paper. The classification results after segmenting the continuous test sequences using DTW algorithm are shown in Fig. 16. It can be observed that the proposed method in this paper is significantly superior to other deep neural network-based methods on the CGD dataset. More importantly, the parameters of the lightweight I3D network are much smaller than the other two methods. Therefore, less time is spent training the feature network on each development batch. Thus it can be seen that the proposed method in this paper is more effective and efficient.

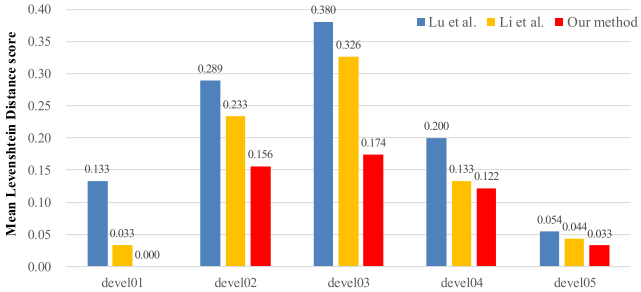
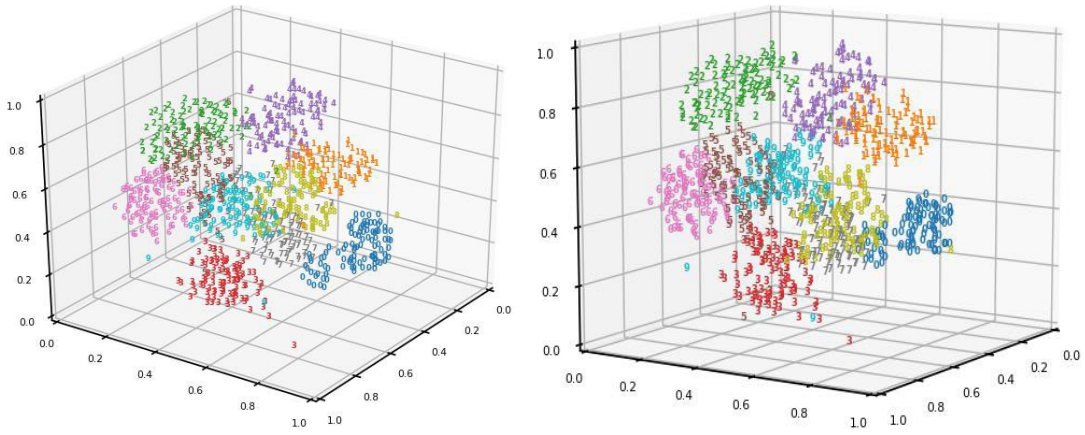


FIGURE 16. Comparison with the state-of-the-art deep neural network-based OSLHGR methods on the development batches (devel01-devel05) of CGD dataset.

D. PERFORMANCE ASSESSMENT

In this section, we used the voting integration mechanism based on multi-classifiers to comprehensively integrate the classification results of different classifiers under the optimal threshold, as shown in Fig. 5. We compared the results of OSLHGR based on the proposed lightweight I3D, the original I3D, and S3D with different evaluation criteria, as illustrated in Table 7. All models are trained on BSG 2.0 dataset and accept images with an input resolution of 224 × 224. It can be seen from the comparison results that the proposed network is more efficient and effective. Furthermore, we also compared the actual inference time of the two models on different experiment platform (Tesla K40c and GTX 1080), as shown in the last two columns of the table. The results show that the proposed model can be implemented in real time on the experiment platform with limited computing resources. Therefore, we believe that such a lightweight model can be used for mobile applications.

In order to understand the spatial distribution of features of different gesture classes more intuitively, we qualitatively evaluate whether these features learned by lightweight I3D are discriminative for gesture classification. It was achieved by visualizing the features extracted from the evaluation subset (containing 10 predefined categories). Then, these high-dimensional features were projected to 3D space using t-distributed stochastic neighbor embedding (t-SNE) [58]. As shown in Fig. 17(a), these categories have a good separability in 3D space. However, there are overlapping areas between different gesture classes in 3D space, such as



(a) Result of feature visualization from the main perspective

(b) Result of feature visualization after rotating at a certain angle

FIGURE 17. Visualization of the extracted features on evaluation subset *E* using t-SNE. Each video is visualized as a digit and videos belonging to the same class have the same color.

'*class*2' and '*class*5'. So to illustrate that these two classes are separable in space, the results of rotation at a certain angle are shown in Fig. 17(b). Therefore, it can be seen that the superior spatial and temporal feature extraction ability of lightweight I3D is an important prerequisite for achieving higher classification performance.

V. CONCLUSION

In this paper, a new approach based on the efficient spatial-temporal feature extraction of lightweight I3D and the discrimination evolution of root sample with cosine similarity measure is proposed for OSLHGR. This method is inspired by the spatiotemporal separability of 3D convolution and the lightweight structural design of Fire module and our previous research work. First, we use the I3D network with the best performance on the UCF-101 dataset as the basic model. On this basis, the three model compression strategies presented in this paper are used for lightweight design of the original I3D network. Then a series of experiments and test results on the IsoGD and Jester datasets verify the effectiveness of the proposed lightweight I3D network structure and CCA-based feature fusion. After that, the nonparametric classification method based on distance similarity measure is used to classify the test samples. Meanwhile, the root sample updating strategy based on discrimination evolution is used to enhance the classification performance of OSLHGR. At the end, a series of tests based on BSG 2.0 and CGD datasets are used to evaluate the performance of the proposed OSLHGR method. The results verify the robustness and timeliness of our proposed method. In the future studies, we will focus on solving the OSLHGR issues in a more complex context so that it can be applied to a broader application scenario and the temporal segmentation of continuous gesture.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Annu. Conf. Adv. Neural Inf. Process. Syst.* Stateline, NV, USA: Harrahs and Harveys, Dec. 2012, pp. 1106–1114.

[2] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 640–651.

[3] T. Wang, M. Qiao, Z. Lin, C. Li, H. Snoussi, Z. Liu, and C. Choi, "Generative neural networks for anomaly detection in crowded scenes," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 5, pp. 1390–1399, May 2019.

[4] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back, "Face recognition: A convolutional neural-network approach," *IEEE Trans. Neural Netw.*, vol. 8, no. 1, pp. 98–113, Jan. 1997.

[5] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 221–231, Jan. 2013.

[6] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, "The kinetics human action video dataset," 2017, *arXiv:1705.06950*. [Online]. Available: https://arxiv.org/abs/1705.06950

[7] J. Carreira and A. Zisserman, "Quo vadis, action recognition? A new model and the kinetics dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 4724–4733.

[8] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: https://arxiv.org/abs/1409.1556

[9] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1–9.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.

[11] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 2261–2269.

[12] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," 2017, *arXiv:1709.01507*. [Online]. Available: https://arxiv.org/abs/1709.01507

[13] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 594–611, Apr. 2006.

[14] I. Guyon, V. Athitsos, P. Jangyodsuk, and H. J. Escalante, "The chalearn gesture dataset (CGD 2011)," *Mach. Vis. Appl.*, vol. 25, no. 8, pp. 1929–1951, 2014.

[15] D. Wu, F. Zhu, and L. Shao, "One shot learning gesture recognition from RGBD images," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops (CVPR)*, Providence, RI, USA, Jun. 2012, pp. 7–12.

[16] J. Wan, Q. Ruan, W. Li, and S. Deng, "One-shot learning gesture recognition from RGB-D data using bag of features," *J. Mach. Learn. Res.*, vol. 14, no. 1, pp. 2549–2582, Sep. 2013.

[17] N. A. Goussies, S. Ubalde, and M. Mejail, "Transfer learning decision forests for gesture recognition," *J. Mach. Learn. Res.*, vol. 15, pp. 3847–3870, Nov. 2014.

[18] J. Konečný and M. Hagara, "One-shot-learning gesture recognition using HOG-HOF features," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 2513–2532, 2014.
[19] J. Wan, V. Athitsos, P. Jangyodsuk, H. J. Escalante, Q. Ruan, and I. Guyon, "CSMMI: Class-specific maximization of mutual information for action and gesture recognition," *IEEE Trans. Image Process.*, vol. 23, no. 7, pp. 3152–3165, Jul. 2014.
[20] J. Wan, G. Guo, and S. Li, "Explore efficient local features from RGB-D data for one-shot learning gesture recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 8, pp. 1626–1639, Aug. 2016.
[21] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *Proc. Int. Conf. Mach. Learn.*, Lille, France, Jul. 2015, pp. 1–30.
[22] Z. Xu, L. Zhu, and Y. Yang, "Few-shot object recognition from machine-labeled Web images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 5358–5366.
[23] L. Kaiser, O. Nachum, A. Roy, and S. Bengio, "Learning to remember rare events," in *Proc. ICLR*, Toulon, France, Apr. 2017, pp. 24–26.
[24] Z. Lu, S. Qin, X. Li, L. Li, and D. Zhang, "One-shot learning hand gesture recognition based on modified 3D convolutional neural networks," *Mach. Vis. Appl.*, pp. 1–24, Aug. 2019. [Online]. Available: https://link.springer.com/article/10.1007/s00138-019-01043-7
[25] X. Li, S. Qin, K. Xu, and Z. Hu, "One-shot learning gesture recognition based on evolution of discrimination with successive memory," in *Proc. IEEE Int. Conf. Intell. Robot. Control Eng. (IRCE)*, Gansu, China, Aug. 2018, pp. 263–269.
[26] M. Denil, B. Shakibi, L. Dinh, M. Ranzato, and D. Nando, "Predicting parameters in deep learning," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2013, pp. 2148–2156.
[27] H. Hu, R. Peng, Y.-W. Tai, and C.-K. Tang, "Network trimming: A data-driven neuron pruning approach towards efficient deep architectures," 2016, *arXiv:1607.03250*. [Online]. Available: https://arxiv.org/abs/1607.03250
[28] H. Mao, S. Han, J. Pool, W. Li, X. Liu, Y. Wang, and W. J. Dally, "Exploring the regularity of sparse structure in convolutional neural networks," 2017, *arXiv:1705.08922*. [Online]. Available: https://arxiv.org/abs/1705.08922
[29] S. Anwar, K. Hwang, and W. Sung, "Structured pruning of deep convolutional neural networks," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 13, no. 3, Dec. 2015, Art. no. 32.
[30] V. Lebedev and V. Lempitsky, "Fast convnets using group-wise brain damage," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 2554–2564.
[31] Z. Liu, J. Li, Z. Shen, G. Huang, S. Yan, and C. Zhang, "Learning efficient convolutional networks through network slimming," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Honolulu, HI, USA, Jul. 2017, pp. 2755–2763.
[32] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. P. Graf, "Pruning filters for efficient convnets," 2016, *arXiv:1608.08710*. [Online]. Available: https://arxiv.org/abs/1608.08710
[33] J.-H. Luo, J. Wu, and W. Lin, "ThiNet: A filter level pruning method for deep neural network compression," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 5068–5076.
[34] M. Lin, Q. Chen, and S. Yan, "Network in network," 2014, *arXiv:1312.4400*. [Online]. Available: https://arxiv.org/abs/1312.4400
[35] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50X fewer parameters and <0.5 MB model size," 2016, *arXiv:1602.07360*. [Online]. Available: https://arxiv.org/abs/1602.07360
[36] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: https://arxiv.org/abs/1704.04861
[37] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," 2017, *arXiv:1707.01083*. [Online]. Available: https://arxiv.org/abs/1707.01083
[38] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy, "Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Munich, Germany, Sep. 2018, pp. 318–335.
[39] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 2818–2826.
[40] J. Duan, J. Wan, S. Zhou, X. Guo, and S. Z. Li, "A unified framework for multi-modal isolated gesture recognition," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 14, no. 1s, 2018, Art. no. 21.
[41] Q. Miao, Y. Li, W. Ouyang, Z. Ma, X. Xu, W. Shi, and X. Cao, "Multimodal gesture recognition based on the ResC3D network," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 3047–3055.
[42] P. Molchanov, S. Gupta, K. Kim, and J. Kautz, "Hand gesture recognition with 3D convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1–7.
[43] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Montreal, QC, Canada, Dec. 2014, pp. 3320–3328.
[44] T. Wang, Y. Chen, M. Zhang, J. Chen, and H. Snoussi, "Internal transfer learning for improving performance in human action recognition for small datasets," *IEEE Access*, vol. 5, pp. 17627–17633, 2017.
[45] J. Wan, Y. Zhao, S. Zhou, I. Guyon, S. Escalera, and S. Z. Li, "ChaLearn looking at people RGB-D isolated and continuous datasets for gesture recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 761–769.
[46] *Twentybn Jester Dataset: A Hand Gesture Dataset*. Accessed: 2017. [Online]. Available: https://www.twentybn.com/datasets/jester
[47] I. Guyon, V. Athitsos, P. Jangyodsuk, B. Hamner, and H. J. Escalante, "ChaLearn gesture challenge: Design and first results," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2012, pp. 1–6.
[48] G. Zhu, L. Zhang, P. Shen, and J. Song, "Multimodal gesture recognition using 3-D convolution and convolutional LSTM," *IEEE Access*, vol. 5, pp. 4517–4524, 2017.
[49] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Inf. Process. Manag.*, vol. 45, no. 4, pp. 427–437, Jul. 2009.
[50] G. Zhu, L. Zhang, L. Mei, J. Shao, J. Song, and P. Shen, "Large-scale isolated gesture recognition using pyramidal 3D convolutional networks," in *Proc. IEEE Conf. Pattern Recognit. (ICPR)*, Cancún, México, Dec. 2016, pp. 19–24.
[51] Y. Li, Q. Miao, K. Tian, Y. Fan, X. Xu, R. Li, and J. F. Song, "Large-scale gesture recognition with a fusion of RGB-D data based on the C3D model," in *Proc. IEEE Conf. Pattern Recognit. (ICPR)*, Cancún, México, Dec. 2016, pp. 25–30.
[52] L. Zhang, G. Zhu, P. Shen, J. Song, S. A. Shah, and M. Bennamoun, "Learning spatiotemporal features using 3DCNN and convolutional LSTM for gesture recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 3120–3128.
[53] L. Zhang, G. Zhu, L. Mei, P. Shen, S. A. A. Shah, and M. Bennamoun, "Attention in convolutional LSTM for gesture recognition," in *Proc. Neural Inf. Process. Syst. (NeurIPS)*, Montreal, QC, Canada, Dec. 2018, pp. 1953–1962.
[54] B. Zhou, A. Andonian, A. Oliva, and A. Torralba, "Temporal relational reasoning in videos," 2017, *arXiv:1711.08496*. [Online]. Available: https://arxiv.org/abs/1711.08496
[55] G. Zhu, L. Zhang, P. Shen, J. Song, S. A. A. Shah, and M. Bennamoun, "Continuous gesture segmentation and recognition using 3DCNN and convolutional LSTM," *IEEE Trans. Multimedia*, vol. 21, no. 4, pp. 1011–1021, Apr. 2019.
[56] X. Chai, Z. Liu, F. Yin, Z. Liu, and X. Chen, "Two streams recurrent neural networks for large-scale continuous gesture recognition," in *Proc. 23rd Int. Conf. Pattern Recognit. (ICPR)*, Cancún, México, Dec. 2016, pp. 31–36.
[57] Z. Liu, X. Chai, Z. Liu, and X. Chen, "Continuous gesture recognition with hand-oriented spatiotemporal feature," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 3056–3064.
[58] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 1, pp. 2579–2605, Nov. 2008.

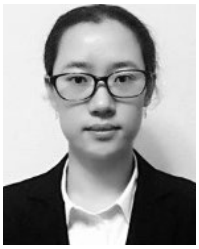
ZHI LU received the bachelor's degree in electronic and information engineering from Shenyang Aerospace University, Shenyang, China, in 2013, and the master's degree in computer science and technology from the Beijing Institute of Information Control, Beijing, China, in 2016. He is currently pursuing the Ph.D. degree with the School of Automation Science and Electrical Engineering, Beihang University, Beijing. His research interests include machine vision and pattern recognition.

SHIYIN QIN received the master's degree in automatic controls and industrial systems engineering from Lanzhou Jiaotong University, Lanzhou, China, in 1984, and the Ph.D. degree in industrial control engineering and intelligent automation from Zhejiang University, Zhejiang, China, in 1990. He is currently a Professor with the School of Automation Science and Electrical Engineering, Beihang University, Beijing, China. His research interests include pattern recognition and machine learning, image processing and computer vision, artificial intelligence, and knowledge engineering.

LIANWEI LI received the bachelor's degree from the School of Control Science and Engineering, Shandong University, Jinan, China, in 2017. He is currently pursuing the Ph.D. degree with the School of Automation Science and Electrical Engineering, Beihang University, Beijing, China. His key research interests include deep learning and computer vision.

DINGHAO ZHANG received the bachelor's degree in automation from Beihang University, in 2018, where he is currently pursuing the master's degree. His current research interests include computer vision and pattern recognition.

KUANHONG XU received the Ph.D. degree from Nankai University, Tianjin, China, in 2011. He is currently an R&D Leader with the Sony China Research Laboratory, with a focus on computer vision and machine learning.



ZHONGYING HU received the Ph.D. degree from Kyushu University, Fukuoka, Japan, in 2015. Since 2015, she has been with the Artificial Intelligence Research Department, Sony China Research Laboratory. Her current research interests include computer vision and machine learning.