

Received April 12, 2021, accepted May 3, 2021, date of publication May 11, 2021, date of current version June 2, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3079337

SqueezeNet and Fusion Network-Based Accurate Fast Fully Convolutional Network for Hand Detection and Gesture Recognition

BAOHUA QIANG¹, YIJIE ZHAI¹, MINGLIANG ZHOU^{1,2,3}, XIANYI YANG¹, (Member, IEEE), BO PENG⁴, YUFENG WANG⁴, AND YUANCHAO PANG¹

¹Guangxi Key Laboratory of Image and Graphic Intelligent Processing, Guilin University of Electronic Technology, Guilin 541004, China

²School of Computer Science, Chongqing University, Chongqing 400044, China

³State Key Laboratory of Internet of Things for Smart City, Faculty of Science and Technology, University of Macau, Taipa, Macau

⁴The 54th Research Institute, China Electronics Technology Group Corporation, Shijiazhuang 050002, China

Corresponding authors: Mingliang Zhou (mingliangzhou@cqu.edu.cn) and Xianyi Yang (xianyiyang65@126.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61762025; in part by the Natural Science Foundation of Guangxi under Grant 2017GXNSFAA198226, Grant 2019GXNSFDA185007, and Grant 2019GXNSFDA185006; in part by the National Natural Science Foundation of Chongqing under Grant cstc2020jcyj-msxmX0790; in part by the National Natural Science Fund of Shanxi under Grant 201901D111154; in part by the Guangxi Key Research and Development Project under Grant AB18126053, Grant AB18126063, Grant AD18281002, and Grant AD19110137; in part by the Development Foundation of the 54th Research Institute of China Electronics Technology Group Corporation under Grant SXX18138 × 017; in part by the Guangxi Key Science and Technology Planning Project under Grant AA18118031 and Grant AA18242028; in part by the Innovation Project of GUET Graduate Education under Grant 2019YCXS051 and Grant 2020YCXS052; in part by the Guangxi Colleges and Universities Key Laboratory of Intelligent Processing of Computer Image and Graphics under Grant GIIP201603 and Grant GIIP1806; in part by the Guangxi Key Laboratory of Cryptography and Information Security under Grant GCIS201905; in part by the Fundamental Research Funds for the Central Universities under Grant 2020CDJ-LHZZ-052; in part by the Human Resources and Social Security Bureau project of Chongqing under Grant cx2020073; and in part by the Suzhou Institute of USTC under Grant H20201528.

ABSTRACT Accurate fast hand detection and gesture recognition for hand understanding are still challenging tasks that are influenced by the diversity of hands and the complexity of the scene in color images. To address the above problem, we propose a novel SqueezeNet and fusion network-based fully convolutional network (SF-FCNet) to accurately and quickly perform hand detection and gesture recognition in color images. First, we introduce the first 17-layer structure in the lightweight SqueezeNet as the hand feature extraction network to accelerate the detection and recognition speed by greatly compressing the network parameters. Second, a precise hand prediction fusion network is designed by adding a residual structure to the deconvolutional network to integrate high- and low-level features of hands, and hand detection and gesture recognition are performed on a single convolutional layer at multiple scales to improve the precision and reduce the computational costs. The verification results on the Oxford hand dataset show that SF-FCNet can reach a precision of 84.1% and a speed of 32 FPS. The experimental results show that SF-FCNet can substantially enhance the precision and speed of hand detection and gesture recognition on three benchmark datasets and has a strong generalization ability on a homemade test set.

INDEX TERMS Convolutional neural network, deep learning, gesture recognition, hand detection, SqueezeNet.

I. INTRODUCTION

Human hand detection and recognition are regarded as a way for computers to understand human language, enabling people to communicate with machines and interact naturally without any mechanical equipment. Human hands

The associate editor coordinating the review of this manuscript and approving it for publication was Thomas Canhao Xu.

and gestures have applications in many computer fields, such as human-computer interaction (HCI) [1], rehabilitation medicine [2], anomaly detection [3], sign language recognition [4], gesture interaction [5], virtual reality [6], etc. Hand detection is a problem that detects the locations of all hands in an image. Gesture recognition is used to detect both the location and category of the gesture in an image. Hand as a special communication tool makes us have higher

requirements for the accuracy and speed of hand detection and gesture recognition. In recent years, the development of deep learning has greatly improved hand detection and gesture recognition technology. However, how to continue to improve the accuracy and detection speed is still a major challenge due to the diversity of hands and the clutter of scenes.

Hand detection and gesture recognition are divided into conventional methods and deep learning-based methods. In some conventional methods [7]–[9], artificial features such as skin color and image shape are extracted, and then the hands and gestures are detected and recognized through modeling and a support vector machines (SVMs) classifier. However, these methods usually have great limitations due to the complexity of the hand, the challenge of modeling, and the inability to perform end-to-end training. Compared with conventional methods, deep learning-based methods have stronger feature expression capabilities due to the automatic extraction of more abstract features using a series of deep convolutional neural networks (CNNs), and the end-to-end training method of deep learning reduces the hand detection and gesture recognition costs. Therefore, the domain of hand detection and gesture recognition has recently been dominated by deep learning.

Encouraged through the use of deep learning networks [10]–[15] for classification and object detection, many methods have been applied for hand detection and gesture recognition, such as region-based CNNs (R-CNNs) [12], Faster R-CNNs [13], Mask R-CNNs [14], and the RefineDet-based method [15]. However, because the detection of hands and gestures belongs to fine-grained detection and hands are small, the accuracy of these object detection networks is not very high. Subsequently, some other methods, such as the multiple scale region-based fully convolutional network (MS-RFCN) [16], region proposal networks (RPN) [17], hand-CNN [18] and without generative adversarial network (GAN) [19], were proposed. These methods improve the hand detection accuracy by improving the RPN, region-based fully convolutional network (R-FCN) [20], Faster R-CNN [13] and Mask R-CNN [14]. There are some approaches, such as RetinaNet based [21] and ResNet50+highlight feature fusion (HFF)+Auxiliary [22], that use ResNet50 [23] as the backbone combined with other networks for detection and recognition. In [24] and [25], basic convolutional pooling layers were used to construct new detection and recognition models to improve the gesture recognition accuracy via end-to-end training on datasets. In [26], a first-person perspective dataset and a CNN-based method, which can distinguish between one's own hands and the hands of others, were proposed. However, these methods still have the following problems. First, little research has been conducted on the detection and recognition speeds of most of these methods, or the speed is slow. Second, the accuracy of most methods has yet to be improved due to the complexity and variability of hands and highocclusion.

To address the above problems, in this study, we investigated hand detection and gesture recognition on the Oxford hand dataset [7], EgoHands dataset [26], and National University of Singapore (NUS) hand posture dataset [8] and proposed a new method named the SqueezeNet and fusion network-based fully convolutional network (SF-FCNet) to accurately and quickly perform hand detection and gesture recognition on images. The main contributions of this study are as follows:

- We propose a fully convolutional network for hand detection and gesture recognition in complex and unconstrained environments and reduced the computational costs.
- We construct a SqueezeNet hand feature extraction network using a lightweight SqueezeNet to reduce the weight parameters, simplify the network structure, and improve the hand detection and gesture recognition speed.
- We design a precise hand prediction fusion network that fuses a deconvolution network and residual structure and includes multiscale feature processing to improve the hand detection and gesture recognition accuracy.
- We show the experimental data and visualization results of SF-FCNet in terms of hand detection and gesture recognition on public datasets and the in-house-built test set.

The remainder of this paper is structured as follows. Section II reviews the related work. The proposed method is presented in Section III, and the experimental results are shown in Section IV. Finally, we draw conclusions in Section V.

II. RELATED WORK

According to the feature extraction method, hand detection and gesture recognition methods are divided into conventional methods and deep learning-based methods. In this section, we review the related work on using traditional methods and deep learning-based methods to solve two problems.

A. HAND DETECTION

In the early stages of hand detection research, some conventional hand detection methods were proposed to detect hands by manually extracting features. Utsumi *et al.* [27] constructed a hand tracking system that recognizes and tracks the appearance of hands with multiple cameras using a geometrical structure-based hand statistical detection method. Xu *et al.* [28] proposed a dynamic hand detection algorithm, which used self-organizing map to realize hand detection and segmentation on the HSV color space. Some workers have proposed skin color-based methods [29]–[31], which can first directly perform hand detection based on skin color or extract the hand according to skin color and then realize noncontact human-computer interactions. Mittal *et al.* [7] used a two-stage method to generate hand bounding boxes based on skin color in the first stage, but the detection accuracy needs to be

improved. Zhao *et al.* [32] proposed a histogram of oriented gradient (HOG)-based hand detection method, which used HOG features for hand detection. Guo *et al.* [33] combined HOG features and SVM classifiers for hand detection. Conventional hand detection methods rely on manual design to extract features, and feature extraction is insufficient and is easily affected by the environment.

In recent years, deep learning-based hand detection methods have begun to attract attention because they can automatically extract features. Initially, some object detection networks [12]–[14] were applied to realize hand detection, but the accuracy needs to be improved. Then, some improved networks were proposed. Dibia [34] proposed a single-shot multibox detector (SSD) [35]-based real-time hand detection method, and testing on the EgoHands dataset showed that the method can achieve real-time hand detection. Chen *et al.* [36] proposed a new deep learning framework that integrated human hand detection and pose estimation and achieved reliable human hand detection through shared convolutional layers. Le *et al.* [37] proposed a cross-resolution feature fusion method, which used two modules to obtain context and semantic information to achieve fast hand detection. Wang and Ye [38] proposed a multiscale Faster R-CNN method, which uses the Faster R-CNN [13] as the basic architecture and combines multiscale integrated features to achieve hand detection. Gao *et al.* [39] proposed a deep CNN model for hand detection, which improved the SSD [35] by combining deep and shallow networks to achieve spatial human-computer interaction. Deep learning-based hand detection methods have better robustness because they can dig deeply into image features, and the learning features are not restricted by the environment.

B. GESTURE RECOGNITION

In the early days, gesture recognition was achieved by wearing sensor gloves or making hand tags. Davis and Shah [40] used hand tags to capture the location and angle information of the hand joints of users to realize gesture recognition. Due to the poor flexibility of sensor gloves and tags, gesture recognition methods for designing artificial hand features have been studied. Van der Bergh *et al.* [41] proposed an average neighborhood margin maximization (ANMM)-based detection system, which used Haarlet coefficients to calculate the degree of matching between hand and sample datasets. Pisharady *et al.* [8] used image shape, texture, and color descriptors to recognize gestures through SVM and obtained high accuracy. Dardas and Georganas [42] proposed a real-time recognition system, which detects and tracks the hand region by subtracting the face color from the skin color and then uses a multiclass SVM to recognize gestures. Yeo *et al.* [43] proposed a method that combined skin color segmentation with Haarlike features, which can effectively remove the interference of the skin color of other parts of the body to improve the accuracy. Ikegami *et al.* [9] proposed a human-computer interaction system that extracts the user's skin color component through face detection and performs

gesture detection according to the skin color, which has good robustness.

Currently, deep learning-based gesture recognition methods are widely used. In [24] and [25], CNN-based methods were proposed, and the basic CNN architecture was used to construct deep learning networks for gesture recognition, which can achieve good recognition accuracy. Wan *et al.* [44] proposed a GAN-based model for the augmentation of hand datasets to improve the gesture recognition accuracy. Chevtchenko *et al.* [45] proposed a feature fusion-based convolutional neural network that combined a CNN with a traditional method and used depth cameras to perform gesture recognition. Si *et al.* [46] proposed a model for detecting raised hands that combines the R-FCN [20] with a feature pyramid and uses an adaptive template selection algorithm to detect raised hands in the in-house-built raised hands dataset. Rouast and Adam [47] proposed a video-based gesture recognition method that used a deep learning architecture to detect video-based gestures and collected a large amount of video data of dining occasions. Neethu *et al.* [48] proposed a CNN-based classification method that used region segmentation, finger segmentation and image normalization to process gestures and finally detected and recognized gestures using the CNN classifier.

In this paper, we mainly study deep learning-based hand detection and gesture recognition methods and propose a SqueezeNet and fusion network-based fully convolutional network, which combines a deconvolution network, a residual structure and multiscale feature processing.

III. METHODOLOGY

The architecture of the proposed network, including a SqueezeNet hand feature extraction network and a precise hand prediction fusion network, is shown in Fig. 1. The input image is first processed by a SqueezeNet hand feature extraction network to produce a map with rich hand features. Then, the feature map with gradually decreasing resolution is obtained by the precise hand prediction fusion network, and the feature map is expanded by the convolutional layer composed of the deconvolution layer and the residual structure. Finally, hand detection and gesture recognition are performed by fusing multiple feature maps on a single convolutional layer. In this section, we introduce two parts of the network: the loss function and the training algorithm of the proposed network.

A. SQUEEZENET HAND FEATURE EXTRACTION NETWORK

To achieve both precision and speed in hand detection and gesture recognition, the choice of the initial feature extraction layer is critical, and it usually involves a trade-off between speed and precision. SqueezeNet was designed to reduce the number of model parameters and the model size by Iandola *et al.* [49]. SqueezeNet can ensure recognition precision while compressing the parameters to approximately 1/50 of AlexNet [50], making the model size only 4.8 MB. SqueezeNet utilizes the strategy of convolutional separation

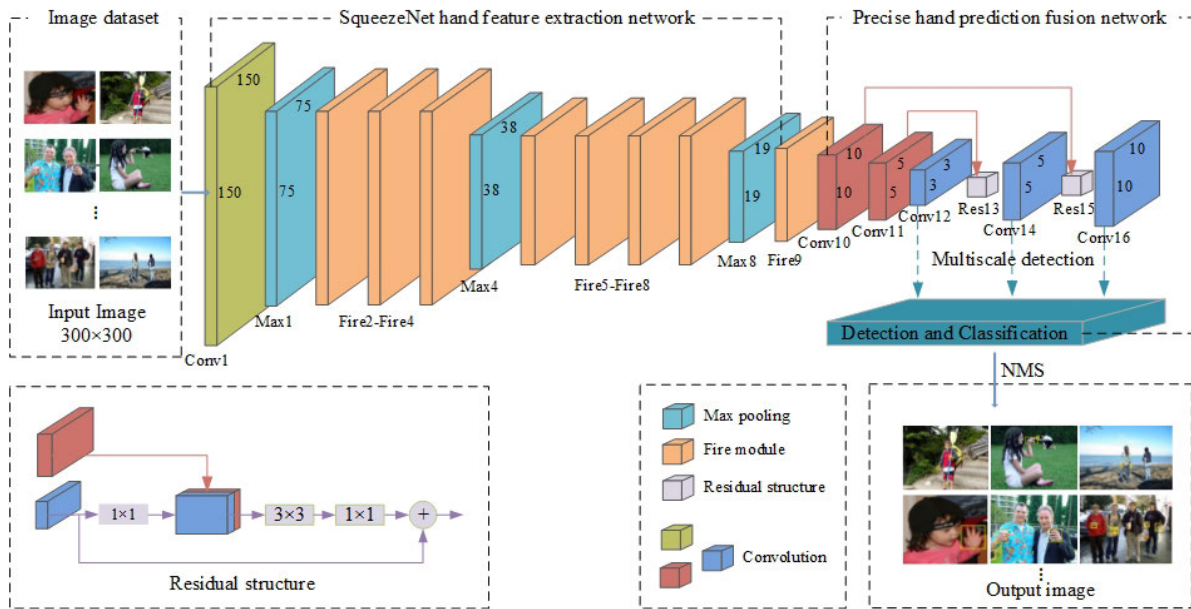


FIGURE 1. Illustration of the proposed network (SF-FCNet) architecture.

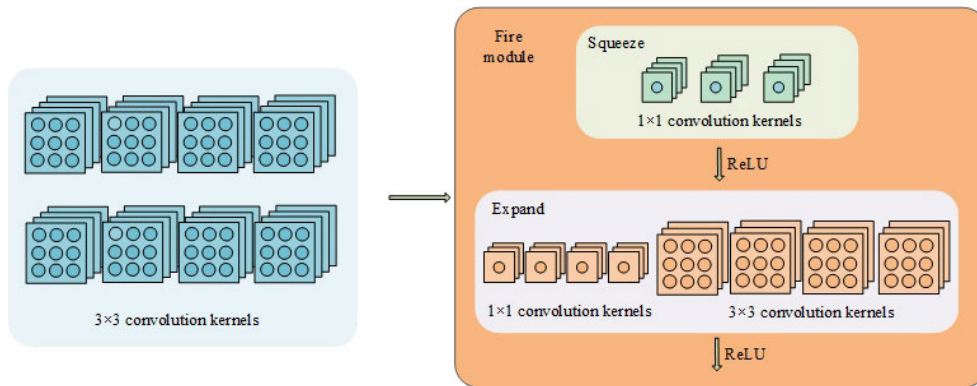


FIGURE 2. Illustration of the 3×3 convolution expanded to a fire module in SqueezeNet.

to convert the standard 3×3 convolution into a fire module by replacing part of the 3×3 convolution kernel with a 1×1 convolution kernel, as shown in Fig. 2. The fire module includes a squeeze layer and an expand layer, and each module includes a rectified linear unit (ReLU) activation function to improve the network depth. The squeeze layer contains 1×1 convolution kernels, and the expand layer contains 1×1 and 3×3 convolution kernels. The 1×1 convolution kernel can reduce the weight parameters, and the 3×3 convolution kernel can ensure the network precision. Because SqueezeNet has the advantages of being a small model and high precision, it is selected as the hand feature extraction network to shorten the feature extraction time and speed up detection and recognition.

To make the network have a certain depth, we deleted the last convolution and average pooling layers of SqueezeNet and retained the first 17 layers as the SqueezeNet hand feature extraction network, as shown in Fig. 1. The input image first passes through “Conv1” and “Max1”, then passes through “Fire2-Fire4” and “Max4”, then passes through

“Fire5-Fire8” and “Max8”, and finally passes through “Fire9”. The hand feature map of the image is extracted through a series of convolutions.

The structure and parameter settings of each layer of the SqueezeNet hand feature extraction network, including 1 convolutional layer, 8 fire modules and 3 max pooling layers with a stride of 2, are shown in Table 1. In the SqueezeNet hand feature extraction network, each fire module has the same structure, including a squeeze layer and an expand layer; and the network depth is 2. On the expand layer, the feature maps of the 1×1 and 3×3 convolutional outputs are spliced together in the channel as the channel of this fire module. The number of convolution kernels in the squeeze layer and expand layer satisfy the following equation:

$$X < Y_1 + Y_2 \tag{1}$$

where X is the number of 1×1 convolution kernels in the squeeze layer; and Y_1 and Y_2 are the number of 1×1 convolution kernels and the number of 3×3 convolution kernels in the expand layer, respectively.

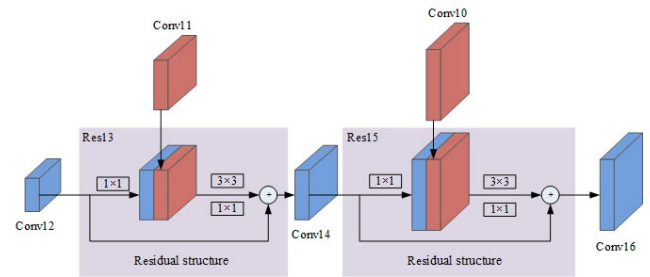
TABLE 1. The architecture parameters of the SqueezeNet hand feature extraction network.

Layer	Filter size/number	Output size	Depth
Input image		300×300×3	
Conv1	7×7/96	150×150×96	1
Maxpool1	3×3/96	75×75×96	0
Fire2	squeeze2	1×1/16	75×75×128
	expand2	1×1/64	
		3×3/64	
Fire3	squeeze3	1×1/16	75×75×128
	expand3	1×1/64	
		3×3/64	
Fire4	squeeze4	1×1/32	75×75×256
	expand4	1×1/128	
		3×3/128	
Maxpool4	squeeze5	3×3/256	38×38×256
	expand5	1×1/32	
		3×3/128	
Fire6	squeeze6	1×1/48	38×38×384
	expand6	1×1/192	
		3×3/192	
Fire7	squeeze7	1×1/48	38×38×384
	expand7	1×1/192	
		3×3/192	
Fire8	squeeze8	1×1/64	38×38×512
	expand8	1×1/256	
		3×3/256	
Maxpool8	squeeze9	3×3/512	19×19×512
	expand9	1×1/64	
		3×3/256	

The input size of the SqueezeNet hand feature extraction network is set to $300 \times 300 \times 3$, and the size of the feature maps is reduced to half of the original size by a 3×3 max pooling layer with a stride of 2. Finally, the $19 \times 19 \times 512$ hand feature map was obtained through “Fire9”. In [35], it was proven that a feature map with a large size is beneficial for the detection of small objects while a feature map with a small size is beneficial for the detection of large objects. To enhance the detection of large and small hands, we pass the $19 \times 19 \times 512$ feature map through a 3×3 convolutional layer with 1024 channels and a step size of 2 to obtain a $10 \times 10 \times 1024$ feature map as the input of the precise hand prediction fusion network to ensure that the size of the subsequent feature map is appropriate.

Table 1 shows that the number of 1×1 convolution kernels in the fire module is greater than the number of 3×3 convolution kernels. The 1×1 convolution kernel can reduce the network dimension with less information loss. Therefore, the SqueezeNet hand feature extraction network can keep more hand information and improve the hand feature extraction speed.

The superior performance of the SqueezeNet hand feature extraction network will be demonstrated in the experiment. It is this fast performance that allows our architecture to be well used for hand detection and gesture recognition that require higher real-time performance.

**FIGURE 3. The fusion fashion of the deconvolution layer and the residual structure.**

B. PRECISE HAND PREDICTION FUSION NETWORK

Inspired by [51], a precise hand prediction fusion network was constructed, as shown in Fig. 1, to supplement the lack of contextual information in the convolution process and obtain better detection performance.

The precise hand prediction fusion network is constructed using deconvolution and combines a residual structure and multiscale detection, as shown in Fig. 1. First, the output of the SqueezeNet hand feature extraction network is used as the input of the fusion network to produce a series of feature maps (“Conv10”, “Conv11”, and “Conv12”) with a gradually decreasing resolution via multiple convolutional layers. Second, the high- and low-level features of hands are fused to obtain feature maps (“Conv14” and “Conv16”) with a gradually increasing resolution via the combination of deconvolution layers and residual structures. Finally, three feature maps (“Conv12”, “Conv14”, and “Conv16”) are provided to the detection and classification layer via multi-scale detection for the detection and classification of hands.

In the precise hand prediction fusion network, “Conv11” is obtained through a set of 1×1 convolution kernels with a step size of 1 and 3×3 convolution kernels with a step size of 2, and “Conv12” is obtained through a set of 1×1 and 3×3 convolution kernels with a step size of 1. The residual structure contains a set of 1×1 , 3×3 , and 1×1 convolution kernels. The fusion method of the deconvolution layer and the residual structure is shown in Fig. 3. First, the “Conv12” input feature maps of the residual structure are upsampled to the size of “Conv11” via bilinear interpolation and then combined with “Conv11” and passed through the “Res13” residual structure. The final output feature map “Conv14” is the sum of the output of the residual structure and the upsampling of “Conv12”. Similarly, “Conv16” is obtained through the fusion of “Conv14” and “Res15”. This fusion method utilizes the feature map with rich information in the early stage to supplement the detailed information that is gradually missing due to the deep convolution, which ensures that the feature map has a larger receptive field, ensures the integrity of the context information, and effectively reduces the missed detection.

Since hand detection and classification are realized through convolutional layers, the entire network is a fully convolutional network, and most of the weights are shared. The final hand detection and classification layers are

converted from the fully connected layer passing through several convolution kernels with the same size as the feature map. The three feature maps (“Conv12”, “Conv14”, and “Conv16”) obtained by the fusion network are provided to the final classification layer, and the NMS algorithm is applied to the feature maps to determine the final detection bounding box of the hand.

The precise hand prediction fusion network uses multiscale prediction and adds residual structures to deconvolution layers to improve the hand detection and gesture recognition accuracy. Multiscale prediction performs the detection and classification of hands according to different sizes of hands to improve the detection accuracy. The contextual information is integrated by fusing residual structures to deconvolution layers, which can increase the detailed information of the hand and simplify the learning process. In addition, the location and classification of hands is performed by the convolutional layers. This fully convolutional network can not only better identify and detect both large and small hands but can also reduce repeated calculations and model complexity.

Compared with object detection, the object of hand is relatively small. Our network integrates high- and low-level features of the large and small feature maps by the residual structure, and utilizes multiscale feature maps with 10×10 , 5×5 and 3×3 to predict different sizes of hands, which improves the utilization of feature mapping with large size. The feature maps with large size are more conducive to the detection of small object [35]. Therefore, our architecture can be well worked for hand detection and gesture recognition, thus the benefits of the precise hand prediction fusion network will be justified in the experiment.

C. LOSS FUNCTION

The locating and classification of hands is achieved by searching bounding boxes. According to different scales and aspect ratios [35], a series of different-sized default bounding boxes will be produced at each pixel position of the feature map extracted by the precise hand prediction fusion network. When a $3 \times 3 \times s$ convolutional kernel is applied to the feature map with s channels, each location of the feature map produces either an output value of category score z_c or an output value of the location offset relative to the default bounding box. The location offset contains 4 offsets relative to the center coordinates, width and height of the default bounding box. For each location of the feature map, we calculate the C category scores and 4 offsets. Assuming that each location in the feature map produces f default bounding boxes, Cf category scores and $4f$ location scores can be obtained through $(C + 4)f3 \times 3 \times s$ convolutional filters. The confidence that each default bounding box matches category c of the hand is calculated as follows:

$$C(z_c) = \frac{e^{z_c}}{\sum_c e^{z_c}} \quad (2)$$

where z_c is the score of the hand for category c .

For each ground truth bounding box, we select some default bounding boxes for matching and use the selected boxes for network training. The intersection over union (IoU) is a metric for evaluating whether the default bounding box and ground truth bounding box match, and its formula is as follows:

$$IoU = \frac{A_{pre} \cap A_{gt}}{A_{pre} \cup A_{gt}} \quad (3)$$

where A_{pre} is the area of the default bounding box of the hand, and A_{gt} is the area of the hand ground truth bounding box of the hand. If IoU is higher than a certain threshold, the default bounding box matches the ground truth bounding box of the hand, and the default bounding box is classified as a positive sample; otherwise, it is a negative sample. Hard-negative mining is used to solve the imbalance between the positive and negative samples by selecting the $top-n$ negative samples with the highest confidence as the negative samples for training and ensuring that the ratio of positive and negative samples is approximately 1:3.

The overall hand detection loss function is the average of the hand confidence loss ($handconf$) and the hand localization loss ($handloc$), which is shown as follows:

$$Loss_{hand} = \frac{1}{N} (Loss_{handconf} + Loss_{handloc}) \quad (4)$$

where N is the number of default bounding boxes matching ground truth bounding boxes. If $N = 0$, the loss function is 0. The hand confidence loss is calculated for positive and negative samples as follows:

$$Loss_{handconf} = - \left(\sum_{i \in ps} x_{i,j}^c \log(C(z_i^c)) + \sum_{i \in ng} \log(C(z_i^0)) \right) \quad (5)$$

where $x_{i,j}^c=1$ represents that the i th default bounding box matches the j th ground truth bounding box of the hand for category c ; otherwise, $x_{i,j}^c=0$. $c = 0$ represents that the category is background. ps is a positive sample, and ng is a negative sample. C_i^c denotes the confidence that the i th default bounding box is category c of the hand.

The hand localization loss is calculated in positive samples as follows:

$$Loss_{handloc} = \sum_{i \in ps} \sum_{k \in \{x,y,w,h\}} x_{i,j}^q \text{smooth}_{L1}(p_i^k - \hat{g}_j^k) \quad (6)$$

where p represents the predicted bounding box, and g represents the ground truth bounding box. We regress the location offset relative to the center (x, y), width w , and height h of the default bounding box b as follows:

$$\hat{g}_j^x = \frac{g_j^x - b_i^x}{b_i^w} \quad \hat{g}_j^y = \frac{g_j^y - b_i^y}{b_i^h} \quad \hat{g}_j^w = \log\left(\frac{g_j^w}{b_i^w}\right) \quad \hat{g}_j^h = \log\left(\frac{g_j^h}{b_i^h}\right) \quad (7)$$

where b_i is the i th default bounding box.

In the training process of our network, the parameters of the network model are constantly updated by minimizing the

overall loss function to achieve better hand detection and gesture recognition results.

D. PROPOSED HAND DETECTION AND GESTURE RECOGNITION ALGORITHM

The details of the training processing of the proposed network are described in Algorithm 1. First, the images are input to the network in batches. Second, the default bounding boxes are generated and divided into positive and negative samples. Finally, the Adam algorithm [52] is used to optimize the loss function in the positive and negative samples by updating the weights until the loss function converges.

Algorithm 1 The Training Process of the Proposed Network

Input: Images.

Output: Weight parameters k of network.

Global parameters:

- Ground truth bounding box g .
- Number of default bounding box f .
- Default bounding box b .
- Positive sample ps , negative sample ng .
- Weight parameters of network k .

Begin

- 1: Randomly load 32 images and corresponding g .
- 2: Produce feature maps through network.
- 3: Produce f default bounding boxes on the feature maps.
- 4: Compute whether b matches g using (3).
- 5: **If** $IoU > 0.5$, b is ps .
- 6: **Otherwise**, b is ng .
- 7: Select $top-n$ ng , keep ps : $ng = 1:3$.
- 8: Calculate confidence loss using (2) and (5).
- 9: Calculate localization loss in ps using (6) and (7).
- 10: Calculate $Loss_{hand}$ using (4).
- 11: Optimize $Loss_{hand}$ using Adam.
- 12: Update k of network.
- 13: **If** convergence, exit the loop.
- 14: **Otherwise**, jump to 1.

End

The method of this paper is trained on the hand dataset for hand detection and gesture recognition and focuses on improving the accuracy and speed performance using a lightweight SqueezeNet and fusion network-based fully convolutional network.

IV. EXPERIMENTS

We show experiments conducted on the Oxford hand dataset [7], EgoHands dataset [26], NUS hand posture dataset [8] and the in-house-built test set. The first part presents the public dataset and the in-house-built test set in detail. The second part introduces the training parameter settings and the metrics. The last two parts discuss the experimental results and the performance of the network.

A. DATASETS

The Oxford hand dataset [7] is a public comprehensive dataset that contains rich hand images from different public image datasets collected without any restrictions. The dataset has 13,050 hand instances with complex backgrounds. All hands that can be clearly seen by humans are marked with bounding rectangles. There are 4069 images for training, 813 images for testing and 444 images for validation.

The EgoHands dataset [26] includes 48 complex first-person interactive videos, which are recorded by 4 actors performing 4 activities in 3 real locations. The dataset has 4800 images with multiple hands and 15053 labeled hand instances. The EgoHands dataset contains four hand categories: “own left”, “own right”, “other left”, and “other right”. There are 3600 images for training, 795 images for testing and 405 images for validation.

The NUS hand posture dataset [8] is shot in and near the National University of Singapore, and the dataset contains 10 classes of gestures of different sizes and shapes with complex backgrounds. Forty volunteers of different races made these gestures to form 2000 different gesture images for gesture recognition. In addition, there are 750 gesture images with human skin color background in the dataset. In order to increase the number of the dataset, we also added 240 images from NUS-I [53]. There are 2990 images in total, including 15 gesture categories. There are 1575 images for training, 1184 images for testing and 231 images for validation.








In the Oxford hand dataset, only the location of the hand that appears in the image is annotated, and the categories of gestures are not distinguished. In the EgoHands dataset, one’s own hand or another’s hand are also annotated except for the location of the hand, but the gesture category is still not labeled. In the NUS hand posture dataset, the location and category of gestures are both annotated. The Oxford hand dataset and EgoHands dataset are used for hand detection, and the NUS hand posture dataset is used for gesture recognition.

Based on the requirements of the Guangxi Key Research and Development Project and testing of the generalization capabilities of the network, we produced two groups of test sets in the laboratory, and some examples are shown in Table 2. In the second group, we randomly selected 6 classes of gestures in the NUS hand posture dataset as the gesture categories. The members of the laboratory as volunteers showed hands and gestures without restriction and used a high-definition (HD) camera to shoot the images. Each group contains 72 images. The first group is used for hand detection, and the second group is used for gesture recognition. The hand and gesture in the in-house-built test set are far from and near the camera. The last column of the second group in Table 2 shows the gesture images that are far from the camera.

B. TRAINING SETUP AND METRICS

The Adam algorithm is used to optimize the loss function of the proposed network. The initial learning rate is set to

TABLE 2. The different categories and some examples from the in-house-built test set.

Group	Category	Some example
First group	hand	
Second group	A	
	M	
	P	
	E	
	O	
	G	

0.0001, the batch size is 32 images, the weight delay parameter is set to 0.0005, and the size of the input images is 300×300 . To reduce the training time and enhance precision, a fine-tuning strategy that loads the weights of SqueezeNet to train other classification tasks in the network is used.

The experiment condition is the TensorFlow framework with 32 GB memory and GTX 1080 Ti with a 3584 CUDA core GPU. The operating system is 64-bit Ubuntu 16.04. The metrics for evaluating hand detection and gesture recognition are the mean average precision (mAP) and frames per second (FPS). The mAP represents the accuracy, and the FPS is the detection speed. The threshold of the *IoU* between the ground truth bounding box and the predicted bounding box can be set from 0.5-0.95.

C. RESULTS

1) OXFORD HAND DATASET

We conducted comparative experiments of mAP and FPS on the Oxford hand dataset, and gave the detection results of hands from South Asian, Africa and far away from the camera to verify the performance of our method in hand detection.

TABLE 3. Comparison of hand detection performance in terms of the mean average precision (mAP) with state-of-the-art methods [15]–[18], [21], [22], R-CNN [12], faster R-CNN [13], mask R-CNN [14] and multiple proposals [7] On the oxford hand Dataset.

Method	mAP (%)
R-CNN [12]	42.3
Multiple proposals [7]	48.2
Faster R-CNN [13]	55.7
Mask R-CNN [14]	70.5
MS-RFCN [16]	75.1
RPN, rotate image [17]	58.1
RefineDet [15]	77.9
RetinaNet-based [21]	72.1
Hand-CNN [18]	78.8
ResNet50+HFF+Auxiliary [22]	80.6
SF-FCNet (ours)	84.1

TABLE 4. Comparison of running time (s) and FPS of state-of-the-art methods on the oxford hand Dataset and titan X GPU.

Method	Running time (s)	FPS	Environment
Multiple proposals [7]	120	0.008	2.5 GHz CPU
R-CNN [12]	9.0	0.111	Titan X GPU
Faster R-CNN [13]	0.08	13	Titan X GPU
MS-RFCN [16]	0.215	5	Titan X GPU
RPN, joint [17]	8.0	0.125	Titan X GPU
RPN, rotate image [17]	1.0	1	Titan X GPU
RPN [17]	0.1	10	Titan X GPU

The performance of the proposed network (SF-FCNet) is verified on the Oxford hand dataset. The comparison of the hand detection performance in terms of the mAP and FPS for the state-of-the-art methods [15]–[18], [21], [22], R-CNN [12], Faster R-CNN [13], Mask R-CNN [14] and multiple proposals [7] is shown in Table 3. Table 3 shows that SF-FCNet trained on the Oxford hand dataset can reach an mAP of 84.1%, which outperforms the state-of-the-art methods [15]–[18], [21], [22]. The superior performance of the method is due to the fusion of a residual structure and a deconvolution network, which can combine the high- and low-level features of hands and detect hand in a multiscale fashion to improve the detection accuracy. The results in Table 3 show the effectiveness of the precise hand prediction fusion network in SF-FCNet in terms of its detection precision.

Tables 4 and 5 show the comparison of SF-FCNet and the state-of-the-art methods [16], [17], [19], R-CNN [12], Faster R-CNN [13] and multiple proposals [7] on running time and FPS on the Oxford hand dataset. The running time is the detection time of each image in seconds. Due to the limitations of the current laboratory hardware environment, we only have GTX 1080 Ti GPU devices. To increase the credibility of our experiment, we use Faster-RCNN as the evaluation medium for the detection speed between two different GPUs.

Table 4 shows the comparison of other methods on the Titan X GPU. It can be seen from Table 4 that Faster R-CNN has the fastest detection speed compared with the other state-of-the-art methods on a Titan X GPU, indicating that Faster-RCNN has the best performance. Table 5 shows the comparison of our methods with other methods on the GTX



FIGURE 4. Hand detection results of the proposed network (SF-FCNet) for multiple hands and hands far away from the camera on the Oxford hand dataset.



FIGURE 5. Hand detection results of the proposed network (SF-FCNet) on South Asian, African and darker skinned hands on the Oxford hand dataset.

TABLE 5. Comparison of the running times (s) and FPS with state-of-the-art methods [19], R-CNN [12] and faster R-CNN [13] on the oxford hand dataset and GTX 1080 Ti GPU.

Method	Running time (s)	FPS	Environment
R-CNN [12]	7.752	0.129	GTX 1080 Ti GPU
Faster R-CNN [13]	0.069	15	GTX 1080 Ti GPU
without GAN [19]	0.111	9	GTX 1080 Ti GPU
SF-FCNet (ours)	0.031	32	GTX 1080 Ti GPU

1080 Ti GPU. It can be seen from Table 5 that SF-FCNet can achieve a detection speed of 32 FPS, which is almost 2.1 times faster than the Faster-RCNN on a GTX 1080 Ti GPU, which indicates that our method has a fast. Combining Tables 4 and 5 shows that SF-FCNet has the fastest detection speed compared with the other methods. This method mainly benefits from the reduction of the weight parameters in the SqueezeNet hand feature extraction network, and the results show the effectiveness of the method in improving the detection speed.

Hand detection in SF-FCNet on the Oxford hand dataset is shown in Figs. 4 and Fig. 5. Fig. 4 shows the hand detection results of SF-FCNet on multiple hand images and hands far away from the camera. Fig. 5 shows the results of hand

detection on images from South Asian, Africa, and areas with dark-skinned people. In the figures, green is the ground truth bounding box, and yellow is the bounding box of the hand predicted by SF-FCNet. The hand detection results in Figs. 4 and 5 show that SF-FCNet can accurately detect the locations of multiple hands, including hands far from the camera and hands from South Asian, African and darker skinned people, which shows the effectiveness of SF-FCNet. The hand far from the camera corresponds to a smaller size in the image, which suggest that our method has better detection results in terms of small-sized hands.

2) EGOHANDS DATASET

We conducted comparison experiment of mAP, and drew a test accuracy curve on the EgoHands dataset, and gave the results of hand detection under the first-person perspective to verify the performance of our method in hand detection.

The comparison of performance in terms of the mAP and method [26] on the EgoHands dataset is shown in Table 6. Table 6 shows that SF-FCNet has higher detection precision on 4 categories of hands, including “own left”, “own right”, “other left”, and “other right”; and it can achieve 89.4% precision on all hands, which outperforms the other methods.

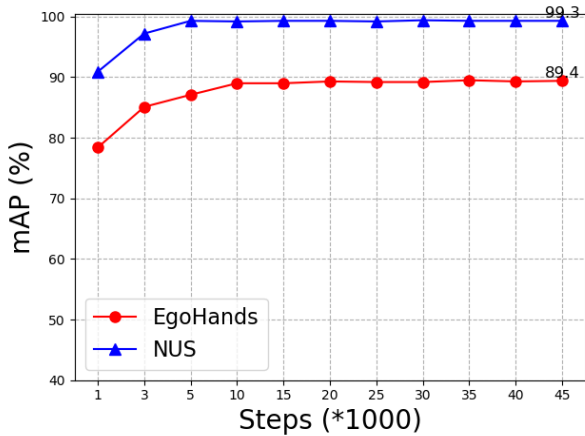


FIGURE 6. The result of hand detection of SF-FCNet under the first-person perspective on the EgoHands dataset.

TABLE 6. Comparison with the state-of-the-art method [26] in terms of the mAP on the EgoHands Dataset.

Category		EgoHands [26]	SF-FCNet (ours)
own hands	left	64.0	88.0
	right	72.7	89.2
other hands	left	81.3	90.1
	right	78.1	90.2
all hands	-	80.7	89.4

The SF-FCNet has a 10.8% higher mAP than [26]. The results show that SF-FCNet has a greater advantage than other state-of-the-art methods in terms of detection precision on the EgoHands dataset.

The red curve in Fig. 6 shows the relationship between the test accuracy of SF-FCNet after training on the EgoHands dataset and steps. The total number of iterations is 45k. The red curve in Fig. 6 shows that the test accuracy of the network gradually converges after the number of steps reaches 10k, which shows that SF-FCNet has a faster convergence rate.

Fig. 7 shows the result of hand detection of SF-FCNet for hands with darker skin and hands far from the camera under the first-person perspective on the EgoHands dataset, where yellow represents the ground truth bounding box, orange represents the predicted bounding box of “other right”, cyan represents the predicted bounding box of “other left”, red represents the predicted bounding box of “own left”, and green represents the predicted bounding box of “own right”. Fig. 7 contains some hands far from the camera, and the second picture in the second row has hands with darker skin. The detection results in Fig. 7 show that these hands have better detection results, which indicates that SF-FCNet can achieve accurate hand detection with darker skin and far from the camera under the first-person perspective.

3) NUS HAND POSTURE DATASET

We conducted comparative experiments and ablation experiments on the NUS hand posture dataset, and drew a test accuracy curve, and gave the results of gesture recognition

TABLE 7. Comparison of gesture recognition performance with the state-of-the-art methods of [24], [25] and [8] in terms of the mAP on the NUS hand posture Dataset.

Method	The number of training/testing	mAP (%)
Pisharady et al. [8]	1800/200	94.4
Mohanty et al. [24]	1200/800	89.1
CNN [25]	1600/400	95.5
SF-FCNet (ours)	1575/1184	99.3

TABLE 8. mAP and FPS of SF-FCNet on the EgoHands Dataset and NUS hand posture Dataset when the threshold of IoU is 0.5 and 0.75.

	EgoHands dataset		NUS hand posture dataset	
	AP _{0.5}	AP _{0.75}	AP _{0.5}	AP _{0.75}
mAP (%)	89.4	82.2	99.3	98.1
FPS	12	11	38	35

TABLE 9. The effect of multiscale features and residual structure on performance measured on the NUS hand posture Dataset.

Method	Multiscale feature	Residual structure	mAP (%)
SF-FCNet	√	×	97.4
SF-FCNet	×	√	98.6
SF-FCNet	×	×	95.2
SF-FCNet	√	√	99.3

in a complex background to verify the performance of our method in gesture recognition.

The comparison of SF-FCNet in terms of the mAP performance with the state-of-the-art methods [24], [25] and [8] on the NUS hand posture dataset is shown Table 7. Table 7 shows that SF-FCNet can reach a mAP of 99.3%, which is higher than those of the state-of-the-art methods [24], [25]. The SF-FCNet attains better gesture recognition precision, which shows the effectiveness of SF-FCNet in gesture recognition.

The blue curve in Fig. 6 shows the relationship between the test accuracy of SF-FCNet after training on the NUS hand posture dataset and steps. The blue curve in Fig. 6 shows that the test accuracy of the network gradually converges after the number of steps reaches 5k, which shows that SF-FCNet has a faster convergence rate.

Table 8 shows the mAP and FPS of SF-FCNet on the EgoHands dataset and NUS hand posture dataset when the threshold of the IoU is 0.5 and 0.75. Table 8 shows that an increase in the threshold will reduce the average precision, while the impact on FPS is not great.

To demonstrate the effectiveness of the multiscale features and residual structure of SF-FCNet, we establish the following experiment on the NUS hand posture dataset: while keeping the original network structure unchanged, we only use one or two of the multiscale features and residual structure or use neither. The experimental results Table 9 show that the accuracy of SF-FCNet with multiscale features and the residual structure is the highest, which indicates that the multiscale and residual structure of SF-FCNet can promote performance improvement to a certain extent.

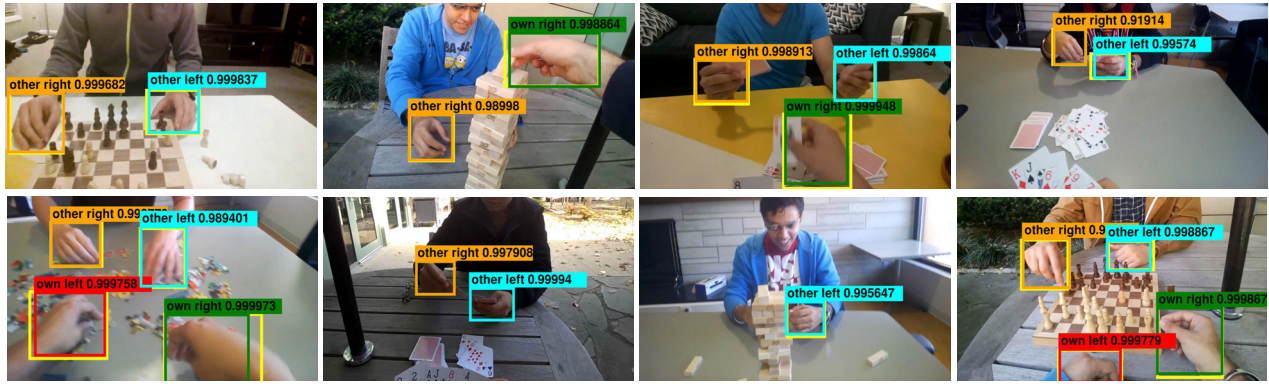


FIGURE 7. The test accuracy curve on the EgoHands dataset and NUS hand posture dataset.

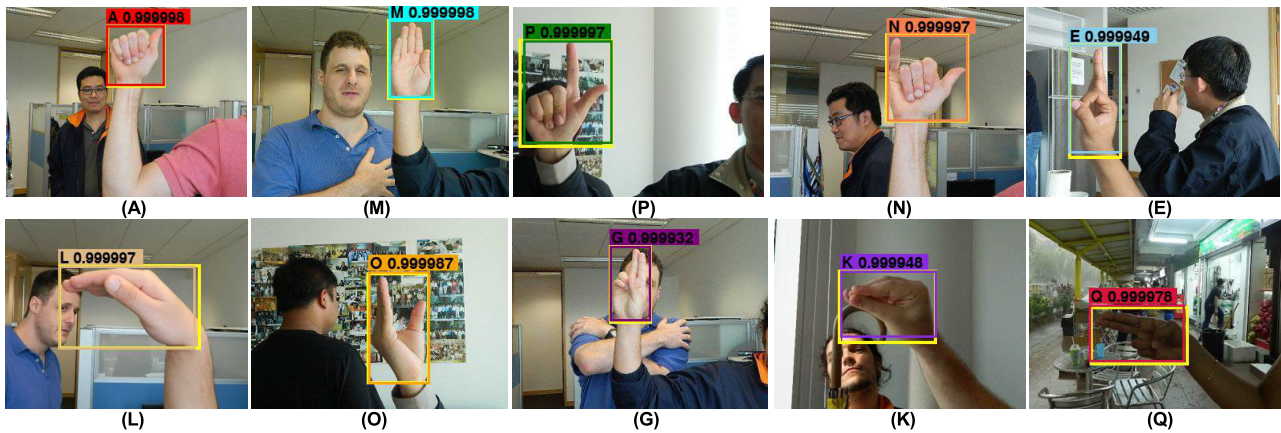


FIGURE 8. The gesture recognition results for 10 categories of gestures with complex backgrounds on the NUS hand posture dataset.

To verify the effectiveness of the SqueezeNet hand feature extraction network, we conduct an experiment on the NUS hand posture dataset: we replace SqueezeNet with ResNet50 as the feature extraction network of SF-FCNet.

TABLE 10. The effect of the SqueezeNet module on performance measured using the NUS hand posture Dataset.

Method	mAP (%)	FPS
SF-FCNet (Resnet50)	99.2	27
SF-FCNet (SqueezeNet)	99.3	38

The experimental results are shown in Table 10. Table 10 shows that the accuracy of SF-FCNet with SqueezeNet is roughly the same as that with ResNet50, but the FPS is greatly improved compared with ResNet50. This is mainly due to the design of the fire module in the SqueezeNet hand feature extraction network, which retains the depth of the network and reduces the weight parameters. The experimental results show that the SqueezeNet hand feature extraction network can improve the efficiency and speed without scarifying accuracy.

Fig. 8 shows recognition results of SF-FCNet for 10 categories of gestures on the NUS hand posture dataset. In Fig. 8, yellow is the ground truth bounding box, other colors are the bounding boxes predicted by SF-FCNet, and a color represents a category. The images in Fig. 8 contain complex

backgrounds such as human faces and cluttered objects. The results show that SF-FCNet can accurately detect the location and category of gestures for images containing a complex background, which shows that SF-FCNet can achieve better gesture recognition when there is interference from other skin colors or cluttered objects.

4) IN-HOUSE-BUILT TEST SET

To evaluate the effectiveness and generalization ability of SF-FCNet, we conducted hand detection and gesture recognition tests on an in-house-built test set.

Fig. 9 shows the detection results of SF-FCNet on our in-house-built test set. In our in-house-built test set, the camera-hand distance range for shooting hands and gestures is 0.5m-1.5m, so the range of the selected hand and gesture is 0.5m-1.5m. The first row shows the hand detection using SF-FCNet trained on the Oxford hand dataset. The second row shows the gesture recognition using SF-FCNet trained on the NUS hand posture dataset. Fig. 9 shows that the detection results of SF-FCNet for hands and gestures with a camera-hand distance between 0.5m-1.5m are basically above 95%, indicating that SF-FCNet has better effectiveness and generalization. These experiments also indicate that our method has better detection results for small hands far away from the camera in terms of the in-house-built test set.



FIGURE 9. The hand detection and gesture recognition results of SF-FCNet on the homemade test set. The first row is hand detection, and the second row is gesture recognition.



FIGURE 10. Some frames captured during real-time gesture recognition through SF-FCNet on the video.

In the Guangxi Key Research and Development Project, SF-FCNet is used for real-time gesture recognition. Fig. 10 shows some of the frames captured during real-time gesture recognition by SF-FCNet on the video. The recognition result on the video demonstrated the real-time performance of SF-FCNet. All the work proves that SF-FCNet has excellent practicability.

D. DISCUSSION

On three benchmark datasets, SF-FCNet achieves a higher mAP than the other state-of-the-art methods. The main reason is that we combine the deconvolution network and the residual structure to increase the detailed information of hand in the precise hand prediction fusion network of SF-FCNet and use multiscale features to improve the accuracy on small hands. In addition, the speed of SF-FCNet is better than those of other state-of-the-art methods on the Oxford hand dataset. This is mainly due to the SqueezeNet hand feature extraction network greatly reducing the weight parameters of the entire network via model compression.

In general, the mAP and FPS of SF-FCNet are superior to those of other state-of-the-art methods, which show that SF-FCNet has state-of-the-art hand detection and gesture recognition performance. The results on the in-house-built test set show the strong generalization ability of SF-FCNet. The detection results of hands and gestures far away from the camera on the four datasets reflects that our method has a

better detection effect on small object hands. In addition, from the analysis of the experimental results of the Oxford hand dataset and EgoHands dataset, our method is more suitable for hand detection with a simple background.

V. CONCLUSION

In this work, we propose a new efficient network (SF-FCNet) for hand detection and gesture recognition in images. The SqueezeNet hand feature extraction network is built to improve the detection speed. A deconvolution network, a residual structure and multiscale processing are introduced to the precise hand prediction fusion network to improve the precision and share weights. The experimental results show that SF-FCNet is competitive and generalizable, and it outperforms other state-of-the-art methods on the three benchmark datasets, which shows that SF-FCNet can achieve accurate and fast hand detection and gesture recognition.

The successful application of SF-FCNET in actual engineering shows that the method has certain validity and practicability. In addition, our work in this paper on different datasets not only provides a new method in the field of hand detection and gesture recognition, but also provides new experimental data for the research in the field of detection speed. The research on the EgoHands dataset also provides a new method for research work under the first-person perspective.

REFERENCES

- [1] M. A. Kassab, M. Ahmed, A. Maher, and B. Zhang, "Real-time human-UAV interaction: New dataset and two novel gesture-based interacting systems," *IEEE Access*, vol. 8, pp. 195030–195045, Oct. 2020, doi: [10.1109/ACCESS.2020.3033157](https://doi.org/10.1109/ACCESS.2020.3033157).
- [2] S. Jacob, V. G. Menon, F. Al-Turjman, P. G. Vinoj, and L. Mostarda, "Artificial muscle intelligence system with deep learning for post-stroke assistance and rehabilitation," *IEEE Access*, vol. 7, pp. 133463–133473, Sep. 2019, doi: [10.1109/ACCESS.2019.2941491](https://doi.org/10.1109/ACCESS.2019.2941491).
- [3] G. Wang, Q. Li, L. Wang, Y. Zhang, and Z. Liu, "Elderly fall detection with an accelerometer using lightweight neural networks," *Electronics*, vol. 8, no. 11, p. 1354, Nov. 2019, doi: [10.3390/electronics8111354](https://doi.org/10.3390/electronics8111354).
- [4] M. Al-Hammadi, G. Muhammad, W. Abdul, M. Alsulaiman, M. A. Bencherif, T. S. Alrayes, H. Mathkour, and M. A. Mekhtiche, "Deep learning-based approach for sign language gesture recognition with efficient hand gesture representation," *IEEE Access*, vol. 8, pp. 192527–192542, Oct. 2020, doi: [10.1109/ACCESS.2020.3032140](https://doi.org/10.1109/ACCESS.2020.3032140).
- [5] C. Lee, J. Kim, S. Cho, J. Kim, J. Yoo, and S. Kwon, "Development of real-time hand gesture recognition for tabletop holographic display interaction using azure kinect," *Sensors*, vol. 20, no. 16, p. 4566, Aug. 2020, doi: [10.3390/s20164566](https://doi.org/10.3390/s20164566).
- [6] X. Yu, L. Jiang, and L. Wang, "Virtual reality gesture recognition based on depth information," in *SID Int. Symp. Dig. Tech. Paper*, Copenhagen, Denmark, 2020, pp. 196–200.
- [7] A. Mittal, A. Zisserman, and P. Torr, "Hand detection using multiple proposals," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, Dundee, U.K., 2011, pp. 75.71–75.11.
- [8] P. K. Pisharady, P. Vadakkepat, and A. P. Loh, "Attention based detection and recognition of hand postures against complex backgrounds," *Int. J. Comput. Vis.*, vol. 101, no. 3, pp. 403–419, Feb. 2013, doi: [10.1007/s11263-012-0560-5](https://doi.org/10.1007/s11263-012-0560-5).
- [9] S. Ikegami, C. Premachandra, B. H. Sudantha, and S. Sumathipala, "A study on mobile robot control by hand gesture detection," in *Proc. 3rd Int. Conf. Inf. Technol. Res. (ICITR)*, Moratuwa, Sri Lanka, Dec. 2018, pp. 1–5.
- [10] A. S. Winoto, M. Kristianus, and C. Premachandra, "Small and slim deep convolutional neural network for mobile device," *IEEE Access*, vol. 8, pp. 125210–125222, Jun. 2020, doi: [10.1109/ACCESS.2020.3005161](https://doi.org/10.1109/ACCESS.2020.3005161).
- [11] Z. Baozhou, Z. Al-Ars, and H. P. Hofstee, "REAF: Reducing approximation of channels by reducing feature reuse within convolution," *IEEE Access*, vol. 8, pp. 169957–169965, Sep. 2020, doi: [10.1109/ACCESS.2020.3024252](https://doi.org/10.1109/ACCESS.2020.3024252).
- [12] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, Jun. 2014, pp. 580–587.
- [13] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: [10.1109/TPAMI.2016.2577031](https://doi.org/10.1109/TPAMI.2016.2577031).
- [14] K. M. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 2980–2988.
- [15] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li, "Single-shot refinement neural network for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake, UT, USA, Jun. 2018, pp. 4203–4212.
- [16] T. H. N. Le, K. G. Quach, C. Zhu, C. N. Duong, K. Luu, and M. Savvides, "Robust hand detection and classification in vehicles and in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Honolulu, HI, USA, Jul. 2017, pp. 1203–1210.
- [17] X. Deng, Y. Zhang, S. Yang, P. Tan, L. Chang, Y. Yuan, and H. Wang, "Joint hand detection and rotation estimation using CNN," *IEEE Trans. Image Process.*, vol. 27, no. 4, pp. 1888–1900, Apr. 2018, doi: [10.1109/TIP.2017.2779600](https://doi.org/10.1109/TIP.2017.2779600).
- [18] S. Narasimhaswamy, Z. Wei, Y. Wang, J. Zhang, and M. H. Nguyen, "Contextual attention for hand detection in the wild," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Piscataway, NJ, USA, Oct. 2019, pp. 9566–9575.
- [19] C. Xu, W. Cai, Y. Li, J. Zhou, and L. Wei, "Accurate hand detection from single-color images by reconstructing hand appearances," *Sensors*, vol. 20, no. 1, p. 192, Dec. 2019, doi: [10.3390/s20010192](https://doi.org/10.3390/s20010192).
- [20] J. F. Dai, Y. Li, K. M. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Conf. Neural Inf. Process. Syst. (NIPS)*, Barcelona, Spain, 2016, pp. 379–387.
- [21] A. A. Q. Mohammed, J. Lv, and M. S. Islam, "A deep learning-based end-to-end composite system for hand detection and gesture recognition," *Sensors*, vol. 19, no. 23, p. 5282, Nov. 2019, doi: [10.3390/s19235282](https://doi.org/10.3390/s19235282).
- [22] D. Liu, L. Zhang, T. Luo, L. Tao, and Y. Wu, "Towards interpretable and robust hand detection via pixel-wise prediction," *Pattern Recognit.*, vol. 105, Sep. 2020, Art. no. 107202, doi: [10.1016/j.patcog.2020.107202](https://doi.org/10.1016/j.patcog.2020.107202).
- [23] 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.
- [24] A. Mohanty, S. S. Rambhatla, and R. R. Sahay, "Deep gesture: Static hand gesture recognition using CNN," in *Proc. Int. Conf. Comput. Vis. Image Process.*, Singapore, 2017, pp. 449–461.
- [25] V. Adithya and R. Rajesh, "A deep convolutional neural network approach for static hand gesture recognition," in *Proc. Conf. Comput. Netw. Commun.*, Trivandrum, India, 2020, pp. 2353–2361.
- [26] S. Bambach, S. Lee, D. J. Crandall, and C. Yu, "Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 1949–1957.
- [27] A. Utsumi, N. Tetsutani, and S. Igi, "Hand detection and tracking using pixel value distribution model for multiple-camera-based gesture interactions," in *Proc. IEEE Workshop Knowl. Media Netw.*, Kyoto, Japan, Jul. 2002, pp. 31–36.
- [28] X. J. Wu, L. Q. Xu, B. Y. Zhang, and Q. G. Ge, "Hand detection based on self-organizing map and motion information," in *Proc. Int. Conf. Neural Netw. Signal Process. (ICNNSP)*, Nanjing, China, 2003, pp. 253–256.
- [29] S. K. Kang, M. Y. Nam, and P. K. Rhee, "Color based hand and finger detection technology for user interaction," in *Proc. Int. Conf. Conver. Hybrid Inf. Technol.*, Daejeon, South Korea, 2008, pp. 229–236.
- [30] Y. Wang, W. Lin, and L. Yang, "A novel real time hand detection based on skin-color," in *Proc. IEEE Int. Symp. Consum. Electron. (ISCE)*, Hsinchu, Taiwan, Jun. 2013, pp. 141–142.
- [31] P. Chinthaka, C. Premachandra, and S. Amarakeerthi, "Effective natural communication between human hand and mobile robot using raspberry-pi," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Las Vegas, NV, USA, Jan. 2018, pp. 1–3.
- [32] Y. Zhao, Z. Song, and X. Wu, "Hand detection using multi-resolution HOG features," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Guangzhou, China, Dec. 2012, pp. 1715–1720.
- [33] J. Guo, J. Cheng, J. Pang, and Y. Guo, "Real-time hand detection based on multi-stage HOG-SVM classifier," in *Proc. IEEE Int. Conf. Image Process.*, Melbourne, VIC, Australia, Sep. 2013, pp. 4108–4111.
- [34] V. Dibia. (2017). *Real-Time Hand Tracking Using SSD on Tensorflow*. Accessed: Jan. 13, 2021. [Online]. Available: <https://github.com/victordibia/handtracking>
- [35] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Computer Vision—ECCV*, Cham, Switzerland: Springer, 2016, pp. 21–37.
- [36] T.-Y. Chen, M.-Y. Wu, Y.-H. Hsieh, and L.-C. Fu, "Deep learning for integrated hand detection and pose estimation," in *Proc. 23rd Int. Conf. Pattern Recognit. (ICPR)*, Cancun, Mexico, Dec. 2016, pp. 615–620.
- [37] T.-H. Le, S.-C. Huang, and D.-W. Jaw, "Cross-resolution feature fusion for fast hand detection in intelligent homecare systems," *IEEE Sensors J.*, vol. 19, no. 12, pp. 4696–4704, Jun. 2019, doi: [10.1109/JSEN.2019.2901259](https://doi.org/10.1109/JSEN.2019.2901259).
- [38] J. Wang and Z. Ye, "An improved faster R-CNN approach for robust hand detection and classification in sign language," in *Proc. 10th Int. Conf. Digit. Image Process. (ICDIP)*, Shanghai, China, Aug. 2018, p. 210.
- [39] Q. Gao, J. Liu, and Z. Ju, "Robust real-time hand detection and localization for space human-robot interaction based on deep learning," *Neurocomputing*, vol. 390, pp. 198–206, May 2020, doi: [10.1016/j.neucom.2019.02.066](https://doi.org/10.1016/j.neucom.2019.02.066).
- [40] J. Davis and M. Shah, "Visual gesture recognition," *IEE Proc.-Vis., Image Signal Process.*, vol. 141, no. 2, pp. 101–106, Apr. 1994, doi: [10.1049/ip-vis:19941058](https://doi.org/10.1049/ip-vis:19941058).
- [41] M. Van den Bergh, E. Koller-Meier, F. Bosche, and L. Van Gool, "Haarlet-based hand gesture recognition for 3D interaction," in *Proc. Workshop Appl. Comput. Vis. (WACV)*, Snowbird, UT, USA, Dec. 2009, pp. 1–8.
- [42] N. H. Dardas and N. D. Georganas, "Real-time hand gesture detection and recognition using bag-of-features and support vector machine techniques," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 11, pp. 3592–3607, Nov. 2011, doi: [10.1109/TIM.2011.2161140](https://doi.org/10.1109/TIM.2011.2161140).

- [43] H.-S. Yeo, B.-G. Lee, and H. Lim, "Hand tracking and gesture recognition system for human-computer interaction using low-cost hardware," *Multimedia Tools Appl.*, vol. 74, no. 8, pp. 2687–2715, Apr. 2015, doi: [10.1007/s11042-013-1501-1](https://doi.org/10.1007/s11042-013-1501-1).
- [44] C. Wan, T. Probst, L. Van Gool, and A. Yao, "Crossing nets: Combining GANs and VAEs with a shared latent space for hand pose estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 1196–1205.
- [45] S. F. Chevtchenko, R. F. Vale, V. Macario, and F. R. Cordeiro, "A convolutional neural network with feature fusion for real-time hand posture recognition," *Appl. Soft Comput.*, vol. 73, pp. 748–766, Dec. 2018, doi: [10.1016/j.asoc.2018.09.010](https://doi.org/10.1016/j.asoc.2018.09.010).
- [46] J. Si, J. Lin, F. Jiang, and R. Shen, "Hand-raising gesture detection in real classrooms using improved R-FCN," *Neurocomputing*, vol. 359, pp. 69–76, Sep. 2019, doi: [10.1016/j.neucom.2019.05.031](https://doi.org/10.1016/j.neucom.2019.05.031).
- [47] P. V. Rouast and M. T. P. Adam, "Learning deep representations for video-based intake gesture detection," *IEEE J. Biomed. Health Informat.*, vol. 24, no. 6, pp. 1727–1737, Jun. 2020, doi: [10.1109/JBHI.2019.2942845](https://doi.org/10.1109/JBHI.2019.2942845).
- [48] P. S. Neethu, R. Suguna, and D. Sathish, "An efficient method for human hand gesture detection and recognition using deep learning convolutional neural networks," *Soft Comput.*, vol. 24, no. 20, pp. 15239–15248, Oct. 2020, doi: [10.1007/s00500-020-04860-5](https://doi.org/10.1007/s00500-020-04860-5).
- [49] 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," 2017, *arXiv:1602.07360*. [Online]. Available: <https://arxiv.org/abs/1602.07360>
- [50] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. 26th Conf. Neural Inf. Process. Syst. (NIPS)*, Lake Tahoe, NV, USA, 2012, pp. 1097–1105.
- [51] N. Dvornik, K. Shmelkov, J. Mairal, and C. Schmid, "BlitzNet: A real-time deep network for scene understanding," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 4174–4182.
- [52] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, 2015, pp. 1–15.
- [53] P. P. Kumar, P. Vadakkepat, and A. P. Loh, "Hand posture and face recognition using a fuzzy-rough approach," *Int. J. Humanoid Robot.*, vol. 7, no. 3, pp. 331–356, Sep. 2010, doi: [10.1142/S0219843610002180](https://doi.org/10.1142/S0219843610002180).



BAOHUA QIANG was born in Nanyang, China, in 1972. He received the B.S. and M.S. degrees from Southwest University, China, in 1996 and 2002, respectively, and the Ph.D. degree from Chongqing University, in 2005. In 2007, he joined the University of Illinois as a Visiting Scholar. From 2007 to 2009, he was a Postdoctoral Researcher with the South China University of Technology. He is currently a Professor with the Guilin University of Electronic Technology.

He has authored one book, more than 60 articles, and more than ten inventions. His major research interests include web information processing, intelligent search, massive data processing, image processing, and network information integration.



YIJIE ZHAI was born in Zhoukou, China, in 1995. She received the B.E. degree from the Henan Institute of Science and Technology, China, in 2018. She is currently pursuing the M.S. degree in computer science and technology with the Guilin University of Electronic Technology, China. Her research interests include gesture recognition, image processing, and deep learning.



MINGLIANG ZHOU received the Ph.D. degree in computer science from Beihang University, Beijing, China, in 2017. He held a postdoctoral position at the Department of Computer Science, City University of Hong Kong, Hong Kong, from September 2017 to September 2019. He is currently a Lecturer with the School of Computer Science, Chongqing University, Chongqing, China. His research interests include image and video coding, perceptual image processing, multimedia signal processing, rate control, multimedia communication, as well as machine learning and optimization.



XIANYI YANG (Member, IEEE) received the B.S. degree in technical physics from Peking University, China, in 1987, the M.S. degree in biophysics from the Chinese Academy of Sciences, China, in 1990, the M.S. degree in electronic and computer engineering from the University of Houston, USA, in 1996, and the Ph.D. degree in electronic and computer engineering from the University of Alberta, Canada, in 1999. He was hired as an Assistant Professor and a Tenured Professor with the University of Guelph, Canada, in 1999 and 2002, respectively. He is currently an Adjunct Professor with the Guilin University of Electronic Technology. He is the author of more than 200 articles. His major research interests include robotics and automation, artificial intelligence, and control science. He won the 2004–2006 President's Outstanding Professor Award of the University of Guelph. He is an Editor of the Special Issue of *International Journal of Robotics and Automation of Biologically Inspired Robotics*, and the Technical Editor of *Dynamics of Continuous, Discrete and Impulsive Systems*, an International Journal, and *International Journal of Information Acquisition*.



BO PENG was born in Shahe, China, in 1990. He received the B.S. degree from North China Electric Power University, China, in 2012. He is currently an Engineer with The 54th Research Institute, China Electronics Technology Group Corporation. His research interests include open source intelligence, big data applications, and deep learning.



YUFENG WANG received the B.E. degree from Xi'an Jiaotong University. He is currently a Researcher with The 54th Research Institute, China Electronics Technology Group Corporation. His research interests include cloud computing, big data, and system integration.



YUANCHAO PANG was born in Beihai, China, in 1993. He received the M.S. degree from the Guilin University of Electronic Technology, China, in 2019. He is currently an Engineer and engaged in research of computer vision. His research interests include object detection, face recognition, and deep learning.

...