S3GRN: Structural Similar Stepwise Generative Recognizable Network for Human Action Recognition With Limited Training Data

CHAOBO LI¹, XULIN SHEN¹, AND HONGJUN LI²*¹,²
¹School of Information Science and Technology, Nantong University, Nantong 226019, China
²State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China
Corresponding author: Hongjun Li (lihongjun@ntu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61871241 and Grant 61976120, in part by the Nanjing University State Key Laboratory for Novel Software Technology under Grant KFKT2019B15, and in part by the Postgraduate Research and Practice Innovation Program of Jiangsu Province under Grant KYCX19_2056.

ABSTRACT Human action recognition is a hot topic and it has been applied to various fields. Deep learning is one of the techniques in human action recognition which has achieved good results. However, the task is still challenging due to the less collected samples. In order to address this challenge and improve the recognition accuracy, the stepwise generative recognizable network is proposed based on the generative adversarial network, which can be used to expand limited training samples and then recognize. Firstly, the stepwise generative recognizable network is designed to combine the function of images generation and recognition for human action. Secondly, the structural similar constraint is introduced to stepwise generative recognizable network, which can compare the similarity of generated images with real data to improve quality and diversity of generated images. Finally, the performance of proposed networks is verified by common databases and the self-build database which is collected in daily life. We achieved 97.14%, 94.88% and 99.69% recognition accuracy on MNIST, Weizmann and self-build dataset, respectively. The experimental results show that the combination of generation and recognition can improve the recognition accuracy without abundant training data, and the structural similar constraint not only can improve the quality and diversity of generated images but also perform better in convergence. The structural similar stepwise generative recognizable network reduces the workload of manual collection and solves the problem of lower recognition accuracy for limited training samples, which achieves the characteristics of natural expanded samples.

INDEX TERMS Generative network, recognizable network, combination, structural similar constraint.

I. INTRODUCTION
Human action recognition is a hot topic in computer vision and pattern recognition [1], which has been applied to various fields, such as medical health [2], intelligent space [3], interactive entertainment [4], robotics [5] and so on. The action information can be obtained from medical equipment to help patients with rehabilitation training. What’s more, intelligent service robots have been loved by the public. The robots can predict the person’s needs by recognizing the human action in the real scene. It also plays a vital role in many tasks such as video retrieval [6] and video surveillance [7]. In urban public security, the basis of judging or predicting the abnormal behavior is to recognize the human action accurately. The requirements for human action recognition are getting higher and higher in some applications. However, the accuracy and efficiency of human action recognition can not fully meet the needs due to the high complexity and limited data.

Human action recognition methods have achieved significant improvements in recent years, and the existing methods are mainly divided into template matching [8], [9] and machine learning [10], [11]. The template matching needs abundant training images to establish the standard templates, and the obvious difference between test images and the standard templates can reduce recognition performance. Deep learning is an important part of machine learning, which
simulates the structure of human brain and analyzes massive data. The potentiality of deep networks, especially Convolutional Neural Network (CNN), has become apparent in human action recognition [12], [13]. Zhang et al. [14] proposed a dual-stream method, where the action vector is encoded and extracted to share the inherent similar structure with the optical flow in videos, and the training speed is improved by using deeply-transferred motion vector CNN. Tu et al. [15] exploited an advanced multistream convolutional neural network to fully use semantics-derived multi-modality in both spatial and temporal domains for action recognition. These methods combine the different types of features to recognize actions, such as optical flow and saliency maps. However, the features are difficult to learn and time-consuming to obtain, and the accuracy of action recognition is affected by the quality of obtained features. Sun et al. [16] proposed the motion map 3D network, where the generative network merges the current video frames into the previous features, and discriminant network classifies the learned features. Yang et al. [17] constructed an asymmetric 3D-CNN deep model to improve ability of the feature learning for the action recognition. All these methods that using videos based on CNN have achieved significant improvements in recent years [18]. But it requires lots of videos for training networks, and the acquisition of abundant data consumes manual workload. It is very expensive on computation and costly on storage for training networks.

The Generative Adversarial Network (GAN) can make fully use of less collected samples and naturally expand images [19]. In order to improve the quality and diversity of generated images, many researchers improved GAN by adding auxiliary information or changing the network structure. Augustus et al. [20] constructed a variant of GANs employing the label to improve training of generative adversarial networks for image synthesis. Niu et al. [21] presented an effective image restoration framework, which minimizes the pixel-wise cross entropy loss and semantic-aware mean square error loss. Balakrishnan et al. [22] presented a modular generative neural network that synthesizes unseen poses using pairs of images. The generative methods based on generative adversarial networks have achieved good performance in image generation.

Meanwhile, the application of generative adversarial network to human action is drawing more and more attention [23]. Gammulle et al. [24] proposed a novel recurrent semi-supervised generative adversarial network for continuous fine-grained human action segmentation, where the generator directs the queued context information to enhance action segmentation. Barsoum et al. [25] proposed a sequence-to-sequence model for probabilistic human action prediction, which learns the probability density function of future poses based on previous poses. Wang et al. [26] introduced the generative adversarial network for action prediction, which improves the accuracy through narrowing the feature difference between partially observed videos and complete ones. The GAN is mainly applied to action segmentation and action prediction without the need to distinguish different actions in detail. Gammulle et al. [27] proposed a multilevel sequential generative adversarial network for group activity recognition, which learns internal representations to discover pertinent sub-activities. These methods mainly take advantage of the merits of generated adversarial network in feature extraction. There are many kinds of human actions in real life and the manifestation of the same action in different individuals is various. The abundant images are required for action recognition. So the few collected images contain less features for training recognizable network directly, which results in the lower accuracy. Therefore, it is necessary to expand the action images enriching the action features for action recognition with few collected images. In addition, the quality of generated images can not be ignored. To solve this problem, the generation and recognition of images are combined to obtain more images for higher recognition accuracy. So the Stepwise Generative Recognizable Network (SGRN) and improved structure namely Structural Similar Stepwise Generative Recognizable Network (S3GRN) are proposed. The proposed networks are composed of two modules: generative module and recognizable module. The generative module is mainly composed of generator and discriminator, while the recognizable module containing the classifier recognizes the human actions based on the real images and the generated images. The main contributions are summarized as follows:

1) The stepwise generative recognizable network combines the function of generation and recognition, which solves the low accuracy of action recognition under limited training data.
2) The structural similar constraint is introduced to stepwise generative recognizable network, called structural similar stepwise generative recognizable network, which can compare the similarity of generated images with real data to improve the quality of generated images.
3) The S3GRN achieves the good performance on natural sample expansion and recognition accuracy even for few training samples. It also has robust architecture, which shows the fast and stable convergence.

The remainder of the paper is organized as follows: the related work is briefly introduced in Section 2. Section 3 discusses the proposed method. In Section 4, we describe experiments and results, while Section 5 ends the paper with our conclusion.

II. RELATED WORKS

With the advance of deep learning, a large number of deep networks have been studied, such as generative adversarial network, convolutional neural network, deep belief network, auto-encoder and so on. The networks we presented are mainly related to two parts: generative adversarial network and deep convolutional generative adversarial network.
A. GENERATIVE ADVERSARIAL NETWORK

Generative adversarial network is a typical generative model proposed by Goodfellow et al. [19]. It consists of generator $G$ and discriminator $D$. The generator is mainly used to obtain the distribution of the real data and generated new samples $G(z)$. The discriminator is equivalent to a binary classifier which used to judge whether the input data is from the real data. The generator and discriminator are trained by discussing the mini-max game:

$$\min_{G} \max_{D} V(D, G) = \mathbb{E}_{x \sim p_{\text{data}}(x)} \left[ \log D(x) \right] + \mathbb{E}_{z \sim p_{z}(z)} \left[ \log(1 - D(G(z))) \right]$$

(1)

where the generator learns the distribution $p_{\text{data}}$ from the real data $x$, the prior distribution of the input noise is $p_{z}(z)$, $D(x)$ represents the probability of $x$ that is from the real data. When the number of samples is sufficient, formula (1) is deformed to:

$$\min_{G} \max_{D} V(D, G) = \mathbb{E}_{x \sim p_{\text{data}}} \left[ \log D_{G}(x) \right] + \mathbb{E}_{z \sim p_{z}} \left[ \log(1 - D_{G}(G(z))) \right]$$

$$= \mathbb{E}_{x \sim p_{\text{data}}} \left[ \log \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_{G}(x)} \right] + \mathbb{E}_{z \sim p_{z}} \left[ \log \frac{p_{G}(x)}{p_{\text{data}}(x) + p_{G}(x)} \right]$$

(2)

The global optimization of mini-max game is transformed into the difference from real data distribution and generated data distribution, and the Nash equilibrium is achieved in ideal [28]. That is, $p_{G} = p_{\text{data}}$, $D_{G}(x) = 0.5$.

B. DEEP CONVOLUTIONAL GENERATIVE ADVERSARIAL NETWORK

Deep convolutional generative adversarial network (DCGAN) [29] is an improvement on GAN. Both generator and discriminator are implemented by convolution neural network to improve the convergence and the quality of generated samples. The specific methods are as follows.

In the generator, the uniform distributed noise $z$ is taken as the input of generator, and the input is converted from a vector to 4-dimensional tensor at the beginning of convolution. The convolution is used to replace the pooling, so that the network can understand the convolution through deconvolution. The corresponding relation between convolution and deconvolution is used to further update the generator and feed back to discriminator. The output layer is followed by the Tanh activation function, while the other hidden layers use the ReLU activation function. In the discriminator, the last generated features are inputted to the Sigmoid function to obtain the probability. The neurons of the feature mapping layers are not fully connected after the convolution, which improves the convergence of the model. The Leaky ReLU is used in the discriminator instead of the ReLU activation function. The output layer of the generator and the input layer of the discriminator are not normalized to improve the stability of network. It prevents the generator to generate a single sample, and the gradient disappears in the deep model.

III. PROPOSED METHOD

The traditional recognizable network needs lots of labeled data for supervised training [30]. The generative adversarial network has been widely studied in image generation and restoration, while the researches in image recognition are beginning to draw more and more attention [31], [32].

A. MOTIVATION

It is well known that most of the networks used for recognition and classification need to learn features from abundant samples to achieve superior performance. However, it may be difficult to obtain abundant samples in some scenarios, which results in the recognition performance be worse than the ideal state. Inspired by the advantage of generative adversarial network, we consider combining the generative adversarial network and recognizable network to solve the problem of low recognition accuracy caused by the limited human action training data. Meanwhile, the differential quality of generated images may have an impact on recognition accuracy. So it is necessary to improve quality of generated images by making the generative adversarial network better and improving the overall performance for generation and recognition.

B. STEPWISE GENERATIVE RECOGNIZABLE NETWORK

Based on the deep convolutional generative adversarial network, the Stepwise Generative Recognizable Network (SGRN) is proposed, where two modules of image generation and recognition are constructed. The generative module is mainly composed of generator and discriminator, and the recognizable module contains the classifier to recognize the hybrid image set which consists of the real images and generated images. The overall framework of SGRN is shown in Figure 1. To formulate our problem, a list of primary symbols in our model is given in Table 1.

![FIGURE 1. The overall framework of SGRN.](image)

The design of SGRN includes three parts: generator, discriminator and classifier. The generator learns the conditional distribution $p_{\text{data}}(x | y)$ from real data, and then make the generated conditional distribution $p_{G}(x | y)$ gradually approach the $p_{\text{data}}(x | y)$. The $G(z | y)$ are generated samples with label.
The detailed architecture of SGRN is shown in Figure 2. It involves the operations that need to be done at each layer, including convolution, activation functions, batch normalization and so on.

In generator, the deconvolution is mainly used to generate different kinds of human action images according to different labels. The input of the generator includes uniformly noise $z$ and corresponding labels $y$. Because the batch size is set to 64 and the database contains 10 categories to be classified, the dimension of the label is $64 \times 1 \times 1 \times 10$. The generated images are labeled automatically according to the real category. The dimension of input noise is transformed by Fully Connected layer (FC). And there are four Deconvolutional layers (Deconv) where the kernel size is $5 \times 5$ and the stride is $2 \times 2$. The batch normalization and Relu function are used behind the full connection layer and the first three deconvolutional layers, while last deconvolutional layer only uses Tanh function for activation. The output of the last layer is taken as the input of the next layer and finally output the generated images.

The discriminator evaluates the images are from the generated images or the real images, so it takes both the generated images and the real images as input. It mainly consists of four Convolutional layers (Conv) and a fully connected layer. The kernel size is $5 \times 5$ and the stride is $2 \times 2$. There is no batch normalization.

### Table 1. List of mathematical symbols.

| Mathematics Symbol | Interpretation |
|--------------------|----------------|
| $C(\cdot)$         | Contrast function between images |
| $C(x; y)$          | The probability that the classifier predicts the images $x$ belong to label $y$ |
| $D(\cdot)$         | The probability of image that is from the real data |
| $D_{\text{eq}}(\cdot)$ | Probability of determining the image from real data under Nash equilibrium |
| $G(z)$             | New generated samples |
| $L(\cdot)$         | Brightness function between images |
| $p_{\text{data}}$  | Distribution of real data |
| $p_{\gamma}$       | Distribution of generated data |
| $p_z$              | Prior distribution of input noise |
| $S(\cdot)$         | Structure function between images |
| $SSIM(\cdot)$      | Structural similarity function |
| $\mathcal{U}$      | Mean value of the image |
| $\mathcal{X}$      | Images that including real images and generated images |
| $\mathcal{Y}$      | Image labels |
| $\mathcal{Z}$      | Noise |
| $\sigma_{(\cdot)}$ | Covariance between images |

The objective function of minimax game is as follows:

$$\min_G \max_D \mathbb{E}_{x \sim p_{\text{data}}(\cdot; y)}[\log D(x; y)] + \mathbb{E}_{z \sim p_z(\cdot)}[\log(1 - D(G(z; y)))] + \mathbb{E}_{x \sim p_{\text{data}}(\cdot); y} [\log C_{y}(x)]$$

where $p_z(\cdot)$ is the prior distribution of input noise. The classifier combines the generated data distribution $p_{\gamma}(x)$ and the real data distribution $p_{\text{data}}(x)$ to classify.

In order to illustrate the overall framework of proposed method clearly, a detailed architecture of SGRN is shown in Figure 2. It involves the operations that need to be done at each layer, including convolution, activation functions, batch normalization and so on.
Following the convolutional layers, the max pooling is used. It mainly consists of two convolutional layers and the action with the highest probability is the final recognition. The size is 64. The output is a vector in which different values and the other part are the real images with labels. The batch optimization is made.

The classifier is mainly used to recognize different human actions by predicting the labels for input images and outputting the probabilities of certain types of actions. It mainly extracts the features from the generated images and the real images. Therefore, one part of the input of the classifier are the generated images with labels outputted by the generator, and the other part are the real images with labels. The batch size is 64. The output is a vector in which different values represent probabilities of belonging to different actions, and the action with the highest probability is the final recognition result. The classifier is on the basis of a six-layer neural network. It mainly consists of two convolutional layers and two pooling layers followed by two fully connected layers. Following the convolutional layers, the max pooling is used for translation invariance. The kernel sizes of the convolution and the pooling layers are 3 × 3 and 2 × 2, respectively. The strides for convolution and pooling are 1 and 2, respectively. The ReLU and softmax function were adopted for the convolutional layers and fully connected layer, respectively. The local response normalization is used after the pooling layer, which improves the generalization ability.

Loss functions are used to optimize generator, discriminator and classifier. Since the parameters of the adaptive optimizer do not need to be adjusted and the learning rate is adjusted automatically. It is suitable for the optimization of high noise, so both the generator and the discriminator adopt the adaptive optimizer. The samples generated by the updated generator can be input to the discriminator, so that the discriminator can judge the more subtle differences between the original samples and the generated samples, which is conducive to give the generator correct and effective feedback. This optimization method can promote generative network to realize the circular optimization. The learning rate of the optimizer is 0.0002 and the exponential delay rate is 0.5. The goal of the optimizer is to minimize the loss of generator, discriminator and classifier. The probability matrix $L$ is defined that discriminator determines whether the input images belongs to real data. The loss of the generator is defined as the cross-entropy between $L$ and the matrix that all elements are one. The loss of the discriminator is defined as the sum of the loss of the generated images and the loss of the real images. The loss of the generated images is the cross entropy between $L$ and the zero matrix, and the loss of the real images is the cross entropy between $L$ and the one matrix. The loss value of the classifier is defined as the cross entropy of the predicted labels and the real image labels. The recognition accuracy is predicted in training and self-optimization is made.

### C. STRUCTURAL SIMILAR STEPWISE GENERATIVE RECOGNIZABLE NETWORK

Natural images are highly structural, which reflect the strong correlations between the pixels in images, especially for the spatial similarity. These correlations carry important information about the structure of objects in visual scenes. That is to say, the correlations of structure contain important information about human action in recognition, such as the structural changes between different actions, the correlation between limbs in the same action and so on. Most of the quality assessment methods based on error sensitivity, such as Mean Square Error (MSE) and Power Signal-to-Noise Ratio (PSNR), use the sum of pixel differences to judge the image distortion, which do not involve the correlation between image structures. Image structure similarity takes the brightness and contrast that are relative to the object as the definition for comparing the image correlations.

The Structural Similarity (SSIM) [33] can compare the essential differences between the generated images and the real images. It compares images from luminance, contrast and structure for accelerating the fitting to the real image and improving the convergence of the network. The luminance $L$, contrast $C$, and the structure $S$ function between $x$ and $G(z)$ are defined as:

$$L(x, G(z)) = \frac{2u_x u_{G(z)} + C_1}{u_x^2 + u_{G(z)}^2 + C_1}$$

$$C(x, G(z)) = \frac{2\sigma_x \sigma_{G(z)} + C_2}{\sigma_x^2 + \sigma_{G(z)}^2 + C_2}$$

$$S(x, G(z)) = \frac{\sigma_{xG(z)} + C_3}{\sigma_x \sigma_{G(z)} + C_3}$$

where $u_x$, $u_{G(z)}$ are the mean value of $x$, $G(z)$. $\sigma_{xG(z)}$ is the co-variance between $x$ and $G(z)$. $\sigma_x$, $\sigma_{G(z)}$ are the standard deviation. $C_1$, $C_2$ and $C_3$ are constants. The structure similarity function of the images are generated by combining the three variables:

$$SSIM(x, G(z)) = L(x, G(z)) \cdot C(x, G(z)) \cdot S(x, G(z))$$

So the structural similarity is introduced to SGRN, namely Structural Similar Stepwise Generative Recognizable Network (S3GRN). The structural similarity constraint is added to the objective function for reducing the loss of the generated features and improving the quality of generated images.

In SGRN, the generator minimizes the objective function $V(D, G, C)$, while the discriminator maximizes the it. That is, $\min_{G} \max_{D, C} V(D, G, C) = \mathop{\text{E}}_{x \sim p_{data}(x|y)} \mathop{\text{log}} D(x|y) + \mathop{\text{E}}_{z \sim p_{z}(z)} [\mathop{\text{log}} (1 - D(G(z)|y))] + \mathop{\text{E}}_{x \sim p_{data}(x), y \sim p_{y}(y)} \mathop{\text{log}} C_{i}(x)$. This objective function is to learn the output probabilities $D(\cdot)$ from the discriminator by adversarial idea. It also updates the network so that the generated images are closer to the real images. The structural similarity is added on S3GRN, not only the output probability from discriminator is adversarial, but also the structural similarity $SSIM(x, G(z|y))$ between the generated images and real images is adversarial for the
The structural similarity not only focuses on the changes of the scene in the image, but also involves the structural changes among different actions and the correlation between limbs in the same action. It evaluates the quality of the generated images by comparing the degree of distortion and feeds back to S3GRN, which is helpful for the generator and discriminator to further understand the quality of the current generated images. It promotes the generated images close to the real image after updating the network.

**TABLE 2. The differences between SGRN and S3GRN.**

| Differences | Gradient |
|-------------|----------|
| SGRN        | $\nabla \sum_{i=1}^{\hat{N}}[\log(1 - D(G(z^{(i)}|y^{(i)}))]$ |
| S3GRN       | $\nabla \sum_{i=1}^{\hat{N}}[\log(1 - D(G(z^{(i)}|y^{(i)}))] - SSIM(x^{(i)}, G(z^{(i)}|y^{(i)}))]$ |
| $D$ SGRN    | $\nabla \sum_{i=1}^{\hat{N}}[\log D(x^{(i)}|y^{(i)})] + \log(1 - D(G(z^{(i)}|y^{(i)}))]$ |
| S3GRN       | $\nabla \sum_{i=1}^{\hat{N}}[\log D(x^{(i)}|y^{(i)})] + \log(1 - D(G(z^{(i)}|y^{(i)}))] - SSIM(x^{(i)}, G(z^{(i)}|y^{(i)}))]$ |

The structural similarity not only focuses on the changes of the scene in the image, but also involves the structural changes among different actions and the correlation between limbs in the same action. It evaluates the quality of the generated images by comparing the degree of distortion and feeds back to S3GRN, which is helpful for the generator and discriminator to further understand the quality of the current generated images. It promotes the generated images close to the real image after updating the network.

**D. THE ALGORITHM**

The algorithm of S3GRN is implemented as follows:

S3GRN: Adaptive Gradient Optimization, Batch Size Is 64, $i = 1, 2, \ldots, 64$

| Initial network parameters |
|---------------------------|
| For the number of generative module iterations do: |
| Generator update: |
| Input the noise $z^{(i)}$ with prior distribution $p_{\theta}(z)$ |
| Input label distribution $y^{(i)}$ of different types of images |
| Update the discriminator gradient: |
| $\nabla \sum_{i=1}^{64}[\log(1 - D(G(z^{(i)}|y^{(i)}))] - SSIM(x^{(i)}, G(z^{(i)}|y^{(i)}))]$ |
| Discriminator update: |
| Input the noise $z^{(i)}$ with prior distribution $p_{\theta}(z)$ |
| Input label distribution $y^{(i)}$ of different types of images |
| Input real images $x^{(i)}$ |
| Update the discriminator gradient: |
| $\nabla \sum_{i=1}^{64}[\log D(x^{(i)}|y^{(i)})] + \log(1 - D(G(z^{(i)}|y^{(i)}))] - SSIM(x^{(i)}, G(z^{(i)}|y^{(i)}))]$ |
| End for |
| Output generated images |
| If the classifier is selected: |
| Input real images $x^{(i)}$ |
| Input generated images $G(z^{(i)})$ |
| Update classifier gradient $\nabla \sum_{i=1}^{64}\log C_{y^{(i)}}(x^{(i)})$ |
| Output recognition result |
| End if |

**IV. EXPERIMENTS**

We conducted our experiments on a Intel Core i7-6800K 3.40 GHz CPU, NVIDIA GTX 1080Ti GPU. Our system is mainly implemented in Anaconda3, Python3.5 and TensorFlow on Windows 10. Due to the limited data, the recognition performance is extremely susceptible to sample selection. So the final results are obtained by taking the average through many experiments with the same configuration. The proposed method is validated in the MNIST [34], Weizmann [35] and our self-built database to verify the effectiveness. In order to verify the performance of the algorithm under limited samples, a small number of images are selected from MNIST and the Weizmann dataset to reconstruct datasets which can be regarded as the less collected samples.
A. VERIFICATION ON MNIST

MNIST dataset is often used to verify the effectiveness of recognition algorithms and generation algorithms [36], [37]. The reasons that we used MNIST dataset are: (1) MNIST dataset is simpler than the other two action datasets. If the interference information exists in images, such as noise, the difference in image quality will be more easily presented when verifying generation performance. (2) Moreover, the structure similarity of same numbers in the MNIST dataset is stronger, and the similarity between different numbers is weaker, which is similar to the characteristics of human action. That is to say, the similarity of the same human actions is stronger, while the similarity of different actions is weaker. Therefore, the use of MNIST dataset is conducive to the combination of action dataset to verify the generalization performance of the proposed method.

The MNIST contains 70000 binary images of handwritten numbers, 28 × 28 in size, 60000 training images and 10000 test images. The 300 images per class are randomly selected from the training set to verify SGRN and S3GRN respectively. The specific settings are as follows: The input and output images with size of 28 × 28. The real images, generated images by SGRN and generated images by S3GRN are shown in Figure 3. The red marks in Figure 3 refer to the larger interference information existing in the generated images by SGRN and S3GRN. Through comparing the images generated by SGRN and S3GRN, it is obvious that the generated images by S3GRN have lower noise and higher sharpness which shows that S3GRN with structural similar constraint can improve the quality of the generated images.

In addition, in order to objectively evaluate the image quality, we use the no-referenced Information Entropy (IE) to intuitively reflect the uncertain interference, such as noise, which affects the image quality. The information entropy not only reflects the amount of information carried by the images, but also reflects the uncertainty of images. In general, the larger the information entropy is, the higher the uncertainty is, which indicates that there will be more interference in the image. Meanwhile, the full referenced image quality assessment methods, Peak Signal to Noise Ratio (PSNR) and the Mean Squared Error (MSE) are adopted to further verify the image quality difference between the two generative methods. The results is shown in the last line of Figure 3.

The information entropy is used to evaluate the real images and the generated images by SGRN and S3GRN. It can be seen that the information entropy of the real images is the smallest, which indicates that the uncertainty of the real images is the smallest, and the interference in these images is the least. The information entropy of generated images by SGRN is larger than S3GRN, indicating the uncertainty of generated images by S3GRN is smaller than that of SGRN, and there are less interference in S3GRN. Furthermore, generated images by S3GRN obtain larger PNSR value and smaller MSE value. These results demonstrate that the noise interference introduced by S3GRN to generate images is less than SGRN, which is also consistent with the visualization of generated images.

The real samples are expanded to satisfy 50 images for different classes. During the training, the recognition accuracy is gradually improved and the classifier converges, so the recognition accuracy is used to judge the convergence of the classifier, as shown in Figure 4. As we can see, the classifier is trained with several training sets with different proportion
between the generated images and real images, and the number of iterations is 1000. Whether SGRN or S3GRN, the training set including 25 real images and 25 generated images in each class converges about 400 iterations, and the recognition accuracy tend to be 1. Two training sets of 10 real images and 40 generated images, single real image and 49 generated images are stable after 300 iterations. The recognition accuracy of S3GRN is higher than SGRN. The results show that the more generated images, the better convergence for the network.

| TABLE 3. Recognition accuracy of different sets. |
|-----------------------------------------------|
| Recognition accuracy (Real images + Generated images) | Class |
| BP | 71.74% | 70.84% | 69.69% |
| CNN | 95.36% | 91.40% | 73.42% |
| SGRN | 96.71% | 94.11% | 90.69% |
| S3GRN | 97.14% | 96.70% | 93.85% |

The recognition accuracy is compared in Table 3. These data are obtained by taking the average value through 10 experiments. Compared with other networks, BP network has the lowest recognition rate. The structure of CNN is similar to the classifier of proposed method. The CNN is trained by using the real images. The recognition accuracy of using a mixture of generated images and real images for training network is higher than using only real images, which due to generated images have a good effect on recognition. It shows that the combination of generation and recognition is helpful to improve the recognition accuracy. Moreover, the recognition accuracy of S3GRN is higher than the SGRN, which is much higher than BP neural network. Especially the recognition accuracy of S3GRN can reach 93.85% when using single real image and 49 generated images, which is 3.16% higher than SGRN. The diverse images generated by S3GRN can express the characteristics of the real dataset more effectively. It shows that the structural similar constraint can improve the quality of the generated images and further improve the recognition efficiency.

B. WEIZMANN
The S3GRN has shown a good performance on MNIST database. Furthermore, the experiment will be done on human action database to verify the robust of S3GRN. Common action libraries include Weizmann, KTH, YouTube, UCF101, HMDB51 and so on. The advantage of Weizmann is smaller labeled data than YouTube, UCF101, HMDB51 and higher quality than KTH. Weizmann database is composed of 90 video sequences which is showed by nine different people, 180 x 144 in size, and each person performs 10 natural actions including walk, bend, run, skip, jump, pjump, side, wave1, wave2 and jack. Each video contains different number of frames and all the sequences can be convert into frames more than 5600. Because of the height, amplitude of action, background and other factors from each person, it is easier to extract the detailed features making fully use of the S3GRN.

We evaluate generative module from the S3GRN using the real images. At training time, 400 images per class are put into network, and output 6400 images through 1000 iteration. The real images, generative images by S3GRN are showed in Figure 5. It is difficult to distinguish if the generated images and the real images are mixed together.

In addition, the differences of loss values for generative module are shown in Figure 6. The structural similar constraint has influenced the loss values of generator and discriminator in S3GRN. The trend of generator loss is almost gentle which show the generator has strong robust. Whether discriminating the generated images or the real images, the loss of discriminator decreases quickly, accelerating network convergence. By comparing the difference between the real images and the generated images, the discriminator can judge the images more accurately and promotes the generator to generate higher quality images.
On recognizable module, there are diverse proportions between real images and generated images for training sets. When testing on 800 real images, the results are shown in Table 4. The speed of convergence for S3GRN is faster than SGRN in the same set. The S3GRN has improved accuracy over the SGRN and CNN. When using few real data, the accuracy of S3GRN is 2.75% higher than SGRN. It is obvious that the mixed set of real images and generated images have a good performance on recognition efficiency.

We further analyze the quality of the generated images by evaluating recognition accuracy with varying amounts of generated data. Increasing the quantity of data generated images does not result in better accuracy when there are lower quality generated samples. In contrast, when the quality of samples is better, whether the generated samples or real samples, the recognition accuracy will be improved with the increase of the number of samples in a certain range.

The Figure 7 shows the influence of combination of real images from the training set and generated images on S3GRN and SGRN respectively. Using 3200 real images from the training set and different number of generated images (Figure 7a), we observe that adding different number of generated images to the real images improves the accuracy on S3GRN and the achieves an accuracy of 94.88%. The SGRN is weaker than S3GRN, adding generated images for SGRN does not provide any noticeable improvement, which may due to poor quality of generated images. When only 3200 generated images is used (Figure 7b), the recognition accuracy of S3GRN is higher than SGRN. With the increase of real images, the recognition accuracy is gradually improved for S3GRN and SGRN.

To prove the performance, the recognition accuracy of six typical and different methods are compared in Table 5. The different feature selection methods are used for human action recognition in references [38]–[40]. Among them, reference [38] mainly used the binary genetic swarm optimization algorithm to select features. The reference [39] combined wrapper-filter feature selection, Ant colony optimization method to train the feature extractor and the automatic optimizer. Reference [40] proposed the cooperative genetic algorithm to select important and discriminating features from the entire feature set to improve the recognition accuracy. Reference [41] used a deep learning network with features optimized using particle swarm optimization for human action recognition. Reference [42] compared the nearest neighbor classifier and Gaussian mixture model classifier for human action recognition. Most of these methods achieve good performance. As we can see in Table 5, the action recognition methods based on optimization algorithms are potential and the recognition accuracy gradually increases. The Gaussian mixture model classifier obtained the best accuracy when using different classifiers to recognize human actions. In addition, the combination of optimization algorithm and deep network achieves an accuracy of 94.00%. By combining recognition with generation, our proposed method achieves an accuracy of 94.88% using both the generated images and real images. The comparison shows that the advantage of proposed S3GRN in human action recognition. This is because the combination of generated images and real images can make more effective use of generated and real human action features.

### C. SELF-BUILT DATABASE

The proposed network has achieved good experimental results in the MNIST and Weizmann database. In order to show the robustness of network, experiments are carried out in the self-built human action image dataset. In order to conform to the life scene, a small amount of sample data is collected and labeled. The dataset consists of 10 categories: bend, grovel, side sit, wave, sit, rightlegs, walk, squat, chest expansion and stretch. Each category includes 321 images and the same action is collected from the same angle but with different action amplitude. The image size is 1920 × 1080. Figure 8 shows the real and the generated images respectively.

| Training data per class | Test data | Convergence time (second) | Recognition accuracy (%) |
|-------------------------|-----------|---------------------------|--------------------------|
| Real images             | CNN       | SGRN                      | CNN                      | SGRN    | S3GRN  |
| 240 80                  | 1235.4    | 1584.0                    | 1569.6                   | 70.75    | 81.00  | 85.13  |
| 160 160                 | 982.6     | 1137.6                    | 964.1                    | 64.38    | 79.38  | 80.00  |
| 40 280                  | 369.3     | 691.2                     | 662.4                    | 53.50    | 75.63  | 78.38  |
TABLE 6. Recognition accuracy and convergence of different training sets.

| Training data per class | Test data (Real images) | Convergence time/second | Recognition accuracy |
|-------------------------|-------------------------|-------------------------|----------------------|
| Real        | Generated | | CNN   | SGRN   | S3GRN   | CNN   | SGRN   | S3GRN   |
| 128         | 128       | 640                     | 44.23 | 83.91  | 82.89   | 97.50% | 98.75% | 99.69% |
| 64          | 192       |                        | 28.26 | 80.64  | 78.32   | 96.41% | 98.12% | 99.38% |
| 32          | 224       |                        | 15.22 | 77.25  | 73.84   | 90.00% | 97.81% | 99.25% |
| 10          | 10        | 3000                    | 12.56 | 15.08  | 15.03   | 79.30% | 90.13% | 90.76% |
| 5           | 15        |                        | 8.26  | 14.41  | 14.46   | 78.97% | 90.03% | 90.03% |
| 1           | 19        |                        | 4.03  | 13.90  | 13.93   | 64.00% | 86.47% | 88.40% |

The quality of the image is evaluated by the generative loss and discriminant loss of outputs per 20 iterations. Figure 9 shows the case where the discriminant loss and the generative loss change as the number of iterations increases. As the number of iterations increases, the generative loss and discriminant loss change from large to small. The adversarial network reaches equilibrium and the image quality is the best.

The generated images are selected randomly and compared with all the real images to calculate the SSIM values. Figure 10 shows the SSIM values between the generated images and all the real images under different actions. For the same action, all the real images are divided into 40 batches and each batch contains 8 images. There is a maximum value for each curve, which means the generated image has the highest similarity to the real images. These generated images can be used as an extended sample of the real image. The difference between maximum value and minimum value shows that the diversity of generated images.

The following experiment is divided into two parts. The ratio of the real images to the generated images in the training data is different. First, the total number of training set and testing set is 3200. When 2560 images and 640 images are used for training and testing respectively, the recognition effect is shown in the first three lines of Table 6. These final results are obtained by taking the average value through 10 experiments. The recognition accuracy of S3GRN is 1.44% higher than SGRN when using 32 real images and 224 generated images per class. Second, the number of training samples is reduced and the number of test samples is increased while the total number of training samples and test samples is 20. The experimental results of the training samples less than the test samples are shown in the last three lines of Table 6. For single real image, S3GRN can achieve 88.40% which is higher than
SGRN and CNN for 1.93% and 24.4% respectively. When the total number of training samples is fixed, both the convergence time and recognition accuracy decreases as the number of generated samples increases. However, the recognition accuracy of the three networks decreased by 7.5%, 0.94% and 0.44% respectively in the first part. In the second part, the recognition accuracy of the three networks decreased by 15.3%, 3.66% and 2.36% respectively. Although the convergence speed of CNN is significantly fastest, the recognition accuracy is lowest. Compared with SGRN, S3GRN has a great performance on recognition accuracy and convergence speed for all the training sets. It shows that the combination of generation and recognition can improve accuracy. In addition, introduced structural similar constraint can improve the quality of the generated images and further improve convergence speed and the recognition efficiency.

V. CONCLUSION

A novel structural similar stepwise generative recognizable network is designed, which solves the problem of lacking abundant training data and avoids the laborious process of manually collecting a large number of images. It pays more attention to the extension of features and the effective use of existing features. The generative module is combined with the recognizable module, in which the recognizable module distinguishes the human action based on the generated samples and the real samples during updating the generative module. The loss functions of the recognizable module and the generative module are jointed, which can promote the generator to pay more attention to the key features used in human action recognition and obtaining more effective features.

In addition, the structural similarity is introduced to the generative module for reducing the loss of the generated features and improving the quality of generated images. Especially, the generated features are not exactly the same as the original features. They may not be as abundant as the original features, or the redundancy may exist in the generated features. It is easier to extract effective features from different aspects when the original features and generated features are input into the recognizable module. It is proved that the proposed method is effective for the expanding samples and improving the recognition accuracy. It has a good performance even in few labeled samples. In the future, we would like to improve the network to accelerate the rate of generating images, reduce the complexity of the algorithm and the wastage of computer resources.

REFERENCES

[1] Q. Wang and K. Chen, “Multi-label zero-shot human action recognition via joint latent ranking embedding,” Neural Netw., vol. 122, pp. 1–23, Feb. 2020.

[2] K. Fan, P. Wang, and S. Zhang, “Human fall detection using slow feature analysis,” Multimedia Tools Appl., vol. 78, no. 7, pp. 9101–9128, Apr. 2019.

[3] Y. Zhang, G. Tian, S. Zhang, and C. Li, “A knowledge-based approach for multiagent collaboration in smart home: From activity recognition to guidance service,” IEEE Trans. Instrum. Meas., vol. 69, no. 2, pp. 317–329, Feb. 2020.

[4] G. Tatiya and J. Sinapov, “Deep multi-sensory object category recognition using interactive behavioral exploration,” in Proc. Int. Conf. Robot. Autom. (ICRA), May 2019, pp. 7872–7878.

[5] C.-C. Wong, H.-C. Chen, C.-T. Lee, C.-C. Wang, and H.-M. Feng, “High interactive sensory robot system design in the indoor autonomous services applications,” J. Intell. Fuzzy Syst., vol. 36, no. 2, pp. 1259–1271, Mar. 2019.

[6] C. Veinidis, I. Pratikakis, and T. Theoharis, “Unsupervised human action retrieval using salient points in 3D mesh sequences,” Multimedia Tools Appl., vol. 78, no. 3, pp. 2789–2814, Feb. 2019.

[7] R. Singh, A. K. S. Kushwaha, and R. Srivastava, “Multi-view recognition system for human activity based on multiple features for video surveillance system,” Multimedia Tools Appl., vol. 78, no. 12, pp. 17165–17196, Jun. 2019.

[8] A. Ulaq, X. Yin, J. He, and Y. Zhang, “On space-time filtering framework for matching human actions across different viewpoints,” IEEE Trans. Image Process., vol. 27, no. 3, pp. 1230–1242, Mar. 2018.

[9] L. Liu, S. Ma, and Q. Fu, “Human action recognition based on locality constrained linear coding and two-dimensional spatial-temporal templates,” in Proc. Chin. Autom. Congr. (CAC), Oct. 2017, pp. 1879–1883.

[10] B. Liu, H. Cai, Z. Ju, and H. Liu, “RGB-D sensing based human action and interaction analysis: A survey,” Pattern Recognit., vol. 94, pp. 1–12, Oct. 2019.

[11] M. Ehatisham-Ul-Haq, A. Javed, M. A. Azam, H. M. A. Malik, A. Itzaza, I. H. Lee, and M. T. Mahmood, “Robust human activity recognition using multimodal feature-level fusion,” IEEE Access, vol. 7, pp. 60736–60751, 2019.

[12] C. N. Phyo, T. T. Zin, and P. Tin, “Deep learning for recognizing human activities using motions of skeletal joints,” IEEE Trans. Consum. Electron., vol. 65, no. 2, pp. 243–252, May 2019.

[13] J. Li, X. Liu, M. Zhang, and D. Wang, “Spatio-temporal deformable 3D ConvNets with attention for action recognition,” Pattern Recognit., vol. 98, Feb. 2020, Art. no. 107037.

[14] B. Zhang, L. Wang, Z. Wang, Y. Qiao, and H. Wang, “Real-time action recognition with deeply transferred motion vector CNNs,” IEEE Trans. Image Process., vol. 27, no. 5, pp. 2326–2339, May 2018.

[15] Z. Tu, W. Xie, J. Daewels, B. Li, and J. Yuan, “Semantic cues enhanced multimodality multistream CNN for action recognition,” IEEE Trans. Circuits Syst. Video Technol., vol. 29, no. 5, pp. 1423–1437, May 2019.

[16] Y. Sun, X. Wu, W. Yu, and F. Yu, “Action recognition with motion map 3D network,” Neurocomputing, vol. 297, pp. 33–39, Jul. 2018.

[17] H. Yang, C. Yuan, B. Li, Y. Du, J. Xing, W. Hu, and S. J. Maybank, “Asymmetric 3D convolutional neural networks for action recognition,” Pattern Recognit., vol. 85, pp. 1–12, Jan. 2019.

[18] A. Voulodimos, N. Douiamis, A. Douiamis, and E. Protopapadakis, “Deep learning for computer vision: A brief review,” Comput. Intel. Neurosci., vol. 2018, pp. 1–13, Feb. 2018.

[19] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in Proc. NIPS, 2014, pp. 2672–2680.

[20] O. Augustus, C. Olah, and S. Jonathon, “Conditional image synthesis with auxiliary classifier GANs,” in Proc. MLR, 2017, pp. 2642–2651.

[21] X. Niu, B. Yan, W. Tan, and J. Wang, “Effective image restoration for semantic segmentation,” Neurocomputing, vol. 374, pp. 100–108, Jan. 2020.

[22] G. Balakrishnan, A. Zhao, A. V. Dalca, F. Durand, and J. Guttag, “Synthesizing images of humans in unseen poses,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 8340–8348.

[23] Z. Pan, W. Yu, X. Yi, A. Khan, F. Yuan, and Y. Zheng, “Recent progress on generative adversarial networks (GANs): A survey,” IEEE Access, vol. 7, pp. 36322–36333, 2019.

[24] H. Gammulle, S. Denman, S. Sridharan, and C. Fookes, “Fine-grained action segmentation using the semi-supervised action GAN,” Pattern Recognit., vol. 98, Feb. 2020, Art. no. 107039.

[25] E. Barsoum, J. Kender, and Z. Liu, “HP-GAN: Probabilistic 3D human motion prediction via GAN,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), Jun. 2018, pp. 1499–1508.

[26] D. Wang, Y. Yuan, and Q. Wang, “Early action prediction with generative adversarial networks,” IEEE Access, vol. 7, pp. 35795–35804, 2019.

[27] H. Gammulle, S. Denman, S. Sridharan, and C. Fookes, “Multi-Level Sequence GAN for Group Activity Recognition,” in Proc. ACCV, 2019, pp. 331–336.
[28] L. J. Ratliff, S. A. Burden, and S. S. Sastry, “Characterization and computation of local Nash equilibria in continuous games,” in Proc. 51st IEEE Annu. Allerton Conf. Commun., Control, Comput., Oct. 2013, pp. 917–924.

[29] A. Radford, L. Metz, and S. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” 2015, arXiv:1511.06434. [Online]. Available: http://arxiv.org/abs/1511.06434

[30] L. Su, X. Xu, Q. Lu, and W. Zhang, “General image classification method based on semi-supervised generative adversarial networks,” High Technol. Lett., vol. 25, no. 1, pp. 35–41, 2019.

[31] L. Wang, C. Gao, L. Yang, Y. Zhao, W. Zuo, and D. Meng “PM-GANs: Discriminative representation learning for action recognition using partial-modalities,” in Proc. ECCV, 2018, pp. 389–406.

[32] L. Zhu, Y. Chen, P. Ghamisi, and J. A. Benediktsson, “Generative adversarial networks for hyperspectral image classification,” IEEE Trans. Geosci. Remote Sens., vol. 56, no. 9, pp. 5046–5063, Sep. 2018.

[33] Z. Wang, E. P. Simoncelli, and A. C. Bovik, “Multiscale structural similarity for image quality assessment,” in Proc. 37th Asilomar Conf. Signals, Syst. Comput., 2003, pp. 1398–1402.

[34] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proc. IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.

[35] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri, “Actions as space-time shapes,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 29, no. 12, pp. 2247–2253, Dec. 2007.

[37] R. Torkzadehmahani, P. Kairouz, and B. Paten, “DP-CGAN: Differentially private synthetic data and label generation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), Jun. 2019, pp. 98–104.

[38] M. Ghosh, R. Guha, I. Alam, P. Loharwal, D. Jalan, and R. Sarkar, “Binary genetic swarm optimization: A combination of GA and PSO for feature selection,” J. Intell. Syst., vol. 29, no. 1, pp. 1598–1610, Sep. 2019.

[39] M. Ghosh, R. Guha, R. Sarkar, and A. Abraham, “A wrapper-filter feature selection technique based on ant colony optimization,” Neural Comput. Appl., vol. 32, no. 12, pp. 7839–7857, Jun. 2020, doi: 10.1007/s00521-019-04171-3.

[40] R. Guha, A. H. Khan, P. K. Singh, R. Sarkar, and D. Bhattacharjee, “CGA: A new feature selection model for visual human action recognition,” Neural Comput. Appl., vol. 32, no. 18, pp. 1–20, Sep. 2020, doi: 10.1007/s00521-020-05297-5.

[41] S. J. Berlin and M. John, “Particle swarm optimization with deep learning for human action recognition,” Multimedia Tools Appl., vol. 79, nos. 25–26, pp. 17349–17371, Jul. 2020.

[42] K.-P. Cho, M. Prasad, D. Wu, N. Sharma, D.-L. Li, Y.-F. Lin, M. Blumenstein, W.-C. Lin, and C.-T. Lin, “Robust feature-based automated multi-view human action recognition system,” IEEE Access, vol. 6, pp. 15283–15296, 2018.

CHAOBO LI received the B.E. degree from the School of Information Science and Technology, Nantong University, Nantong, China, where she is currently pursuing the M.S. degree. Her research interests mainly include computer vision and deep learning.

XULIN SHEN received the B.E. degree from the School of Information Science and Technology, Nantong University, Nantong, China, where she is currently pursuing the M.S. degree. Her research interests include pattern recognition and fuzzy systems.

HONGJUN LI received the Ph.D. degree from the Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing, China, in 2011. He was a Visiting Scientist with the Centre for Pattern Recognition and Machine Intelligence (CENPARMI), Concordia University, Montreal, QC, Canada. He is currently an Associate Professor with Nantong University, China. He has published over 40 papers in flagship journals and conference proceedings as the first author. His research interests include image processing, pattern recognition, and artificial intelligence. He is a Senior Member of CCA, CCF, and CSIG. He regularly serves on the program committee for premier conferences, such as ICPRAI, IJAC, ICIAL, and CISP-BMEI. He is a Reviewer of the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, INFORMATION SCIENCE, and Neurocomputing.