ISSN 1330-3651 (Print), ISSN 1848-6339 (Online)
https://doi.org/10.17559/TV-20240416001467
Original scientific paper

Improved Discriminator Network Based on Residual Network for Person Image Synthesis

Zixuan CHEN, Lingyu YAN*, Chunzhi WANG, Zhiwei YE

Abstract: As the internet industry continues to advance, there has been a significant growth in the volume and variety of data, rich in features and diversity. Analyzing the distribution pattern of images quickly has become a challenge. The utilization of Convolutional Neural Networks (CNNs) has greatly enhanced the efficiency of data analysis and processing in areas like image recognition, image segmentation, and image synthesis. Despite their effectiveness, CNNs face challenges in terms of computational resources, convergence speed, and generating high-quality synthetic images. To address the described problems, this paper focuses on deep residual networks and generative adversarial networks in image synthesis algorithms. It divides human object image synthesis into three stages. The first step segments the human object image's target subject, the second step enhances the synthesized image data, and the third step fuses the segmented feature maps. To mitigate the issue of poor quality composite images, the DENSE-GAN method can be employed. To tackle the noted shortcomings, one approach is to enhance the discriminator network using a residual network. This involves substituting the CNN within the discriminator network's feature extraction module with a residual network. Additionally, to further enhance feature expansion, it is crucial to increase the network's depth by adding more layers. Different parts of the network output layer extract the overall contour information, local detail information, and identity information of the composite image. The results of these three modules are used to guide the weights of DENSE-GAN, adjusting them and experimenting on the dataset, which significantly improves the quality and clarity of DENSE-GAN composite images. Our algorithm achieved promising results on the MNIST, Market-1501, CelebA, and DeepFashion datasets. For example, after training for 100 epochs on the MNIST dataset, our algorithm exhibited an average loss of approximately 0.75.

Keywords: DENSE-GAN; generative adversarial networks; image synthesis; residual networks

1 INTRODUCTION

In recent years, image synthesis has emerged as a prominent research focus in the field of artificial intelligence, gaining significant attention and popularity in various domains including target detection, human-computer interaction, computer vision [1-3], image recognition, advertising design, and film production. The applications of image synthesis extend beyond these domains and encompass fields such as image anti-counterfeiting, biomedicine, communication engineering, and video surveillance. For example, in portrait pose state synthesis and target pose synthesis as well as specific face image recognition [4, 5]; in using AI to generate images in business environments to perceive customer value, social influence, and facilitation conditions [6]; and in researching the determinants of trust in communication between consumers and AI chatbots [7] and in evaluating self-driving cars based on user big data management and AI [8]. Therefore, in the field of image synthesis, research teams both domestically and internationally have invested significant resources into studying image synthesis algorithms and have achieved remarkable success. This includes not only interpretable image synthesis, which enables users to intuitively understand the process by which models generate images and allows intervention and adjustment in the generation process, but also cross-modal image synthesis, a technology involving the conversion of different types of data (such as text, speech, images) into images, or vice versa. Currently, due to the wide range of image sources, it is often necessary to use experienced individuals to identify some blurred images that are difficult to identify manually. With the help of image recognition and image synthesis algorithms, these challenging-to-identify images can be enhanced and repaired, thus improving the recognition rate of images and reducing the cost of manual learning. The human object image synthesis technique has a very wide range of applications but also has some problems, such as basic constraints from sequential evaluation [9], low positioning accuracy [10], poor ability to extract the original parallax map from different algorithms [11], and less robust generalization across datasets [12].

Lately, supervised learning with convolutional networks (CNNs) has been widely used in computer vision applications, while unsupervised learning has received less attention compared to CNNs [13]. Some early researchers used CNNs to implement character synthesis [14, 15]. However, CNNs lack the ability to achieve efficient spatial transformations and require a large number of convolutional operations to model long-term dependencies. If the convolutional network does not contain short enough connections between the layers close to the input and those close to the output, it will cause problems with low training efficiency and accuracy [16, 17].

To achieve effective spatial transformation, some researchers later adopted a flow-based approach [18, 19] to separate the exterior streams to obtain denser correspondences and thus achieve transformations. These methods complete the prediction of two-dimensional coordinate shifts and assign sampling locations to individual target points. While textures that are very similar to reality can be reconstructed with this method, it is accompanied by noticeable artifacts that become more prominent when severe and complex occlusion and distortion are observed [15, 20]. An attention mechanism was introduced to capture long-term dependencies, allowing direct computation of interactions between arbitrary two locations to build dependencies. However, since the target image is the result of source distortion, it can be concluded that the attention correction matrix is not dense, and irrelevant regions are excluded.

After several years of development, the prevalent research direction in image synthesis has gradually shifted towards generative adversarial networks (GANs), with many methods directly learning mappings from source images and utilizing neural networks to generate target images [21, 22]. To facilitate the transfer of feature information from source images to the target point, many

methods adopt a two-branch framework. This framework comprises both site branching and image branching.

To tackle the issue of generating low-quality synthetic images, this paper introduces a solution called Residual Generative Adversarial Networks (RGAN). This paper improves the discriminator network by introducing two key modifications. The first involves replacing the convolutional neural network (CNN) in the discriminator's feature extraction module with a residual network, and the second enhances the feature extraction capability of the discriminator by increasing the depth of the network, as modified above. The network's output layer incorporates three fully connected layers to extract distinct types of information from the composite image. These layers are responsible for capturing the overall contour information, detail information, and identity information. The outputs from these three modules are utilized to guide the weight adjustment of the residual generative adversarial network. This article mentions existing work on individual synthesis in the related works section. Sections 3, 4, and 5 of this article introduce the residual generative adversarial network, some experiments and their results, and the final conclusions.

2 RELATED WORKS

2.1 Existing Image Composition Algorithms

Much of human behavioral understanding is a complex yet unsolved problem, and traditional image synthesis often relies more on human experience to label image features, requiring accurate modeling of motion at both local and global levels. This process can demand significant human labor and often results in synthetic images of less than ideal quality, clarity, and naturalness [23]. The application of computer vision has surged with the advancing times, and generative adversarial networks (GANs) play a particularly crucial role in image synthesis [24]. GANs are frequently employed in unsupervised or semi-supervised learning and consistently outperform traditional image synthesis algorithms, exhibiting superior generalization across datasets [25].

Fig. 1 shows a flowchart of the image synthesis algorithm. Whether an image or video is entered, it is eventually converted to an image during processing. The next step involves feature extraction from the image data, resulting in the generation of a feature map. The third step is the fusion of the obtained feature map, followed by the fourth step, which is the classification of the fused image. The fifth step involves decision making and interpretation of the classified image, and finally, the sixth step is the output of the synthesized image.

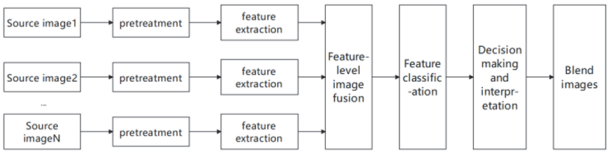


Figure 1 Image synthesis algorithm flow diagram

Fig. 2 depicts a traditional face recognition algorithm for face synthesis based on fused HOG features and deep belief networks. The method divides the feature extraction of face images into global HOG and chunked HOG. The global HOG represents global face image features such as contour and hue, while the chunked HOG represents local image features such as pose, lighting, expression, identity information, etc. [26].

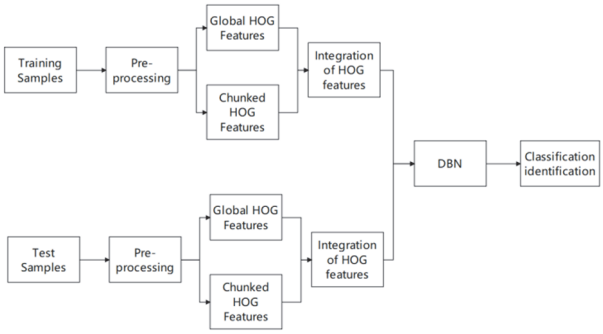


Figure 2 Flow chart of traditional face recognition algorithm based on HOG feature and deep belief network

2.2 Image Synthesis Algorithm for Generative Adversarial Networks

In recent years, the most widely used algorithm for image synthesis is GAN [27]. This algorithm is predominantly employed to tackle image synthesis challenges in scenarios including human posture transfer [28], generating synthetic images from text [4], and analyzing the heterogeneity of brain diseases from medical images [29]. Fig. 3 illustrates some of the algorithms for image synthesis in recent years, with their networks' basic structure primarily being generative adversarial networks [30, 31]. The generator G takes random noise as input and produces an image as output. Its purpose is to be trained through gradient descent to generate deceptive images that can fool the discriminator. In GAN, the role of the discriminator D is to determine whether the generated image meets the standard [32]. If the image does not meet the standard, further training will be conducted. As both generators and discriminators are iteratively trained, they gradually reach equilibrium, where the output produces images aligned with the desired standard [33, 4].

1. Algorithm flow.

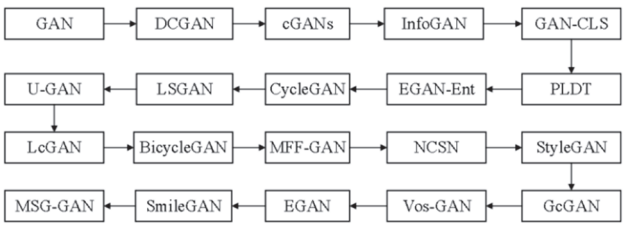


Figure 3 Map of image synthesis algorithms in recent years

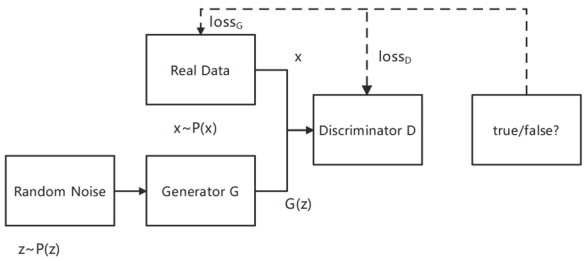


Figure 4 GAN architecture diagram

Discriminant models (D) and generative models (G) constitute the central framework of generative adversarial networks (GANs), as visually depicted in Fig. 4. GANs are typically utilized for binary classification problems [34], and this paper primarily focuses on the synthesis of face images, which falls under binary classification problems.

The generator G is a differentiable nonlinear network, and the authenticity (true/false) of the image generated by G is judged by the discriminator D. During the training process, G utilizes the feedback from D through the convolution operation of the neural network to map the low-dimensional random noise into the high-dimensional real image [35]. The primary role of the discriminant model (D) is to assess the authenticity of the generated images. Initially, it calculates the Euclidean distance between the image produced by the generative model (G) and the real image. Based on this calculation, it makes a binary judgment of true or false to indicate the authenticity of the generated image. This judgment is then provided as feedback to the generative model (G). The parameters of the GAN are adjusted according to the feedback result, and when the image generated by G has successfully fooled D, the two networks reach a balance. This equilibrium is achieved when the two networks reach stability in the process of continuous iteration, and meaningful data is generated.

Algorithm 1 is the process of GAN training

Algorithm 1 Generating adversarial networks for image synthesis algorithms

Inputs: training dataset X, number of iterations N, number of training batches at a time n
Output: the generator network produces G(z) as its output, D(x) and D(G(z)) represent the respective outputs of the discriminator network
for i=1 to T do
for j=1 to N/n do
(a) Choose a subset of samples from the initial random noise set Z{z(1), z(2), …, z(n)}.
(b) Take an equal number of samples {x(1), x(2), ..., x(n)} from the training dataset X, matching the number of samples selected in part (a).
(c) Initialize the discriminators G0 and D0, which are implemented to find D(x(1)) according to the result of the initialized G(z(1)), and calculate the distribution probability between D(x(1)) and D(G(z(1)), which will be input to G according to the result of probability judgment.
(d) Perform iterations to train D1,G1,D2,G2... .Dn,Gn.
(e) the training ends when the two networks reachequilibrium.
end
end

Eq. (1) represents the objective function of the Generative Adversarial Network (GAN):

$$V_{\mathrm{GAN}}\left(G, D\right) = E_{x\sim Pdata(x)}\left[\log\left(D\left(x\right)\right)\right] + \\ +E_{z\sim Pz(z)}\left[\log\left(1-D\left(G\left(z\right)\right)\right)\right] \tag{1}$$

Eq. (1) defines x as the actual data, P_{data} as the distribution pattern of the real data, and z as a stochastic disturbance. $D(x)$ refers to the discriminant neural network that produces 0 or 1 as its output, $G(z)$ represents the resulting data generated after introducing random noise as input, and the function $D(G(z))$ is used to assess the authenticity of the generated data. The desired outcome for model D is that $D(G(z))$ equals 0, indicating the generated data is recognized as fake. Conversely, for model G, the objective is to have $D(G(z))$ equal to 1, suggesting the generated data is perceived as real. Finding the balance between G and D is crucial; it should not be too small or too large to ensure the simulated data resembles valid data.

$$\arg\min_{G}\max_{D} V_{\mathrm{GAN}}\left(G, D\right) \tag{2}$$

The optimal objective function of the generative adversarial network (GAN) is illustrated in the figure. When the generator (G) and discriminator (D) networks reach equilibrium, the optimal state is achieved. At this point, G produces higher-quality images, and D becomes more adept at distinguishing between true and false images.

2. Principle of the algorithm.

The GAN accomplishes this by modeling the data distribution using a non-linear activation function and training a target distribution model from low to high dimensions. Structure diagrams of the network D illustrate this process.

Table 1 Generator network structure

Generator Network (G)						
Layer	X	$G1$	$G2$	$G3$	$G4$	$G(z)$
Kernel Num	64	128	256	512	1024	784
Kernel size	-	3	3	3	3	-
Stride	-	1	1	1	1	-

The following is the principle of generator network (G).

Input layer: The real data and the output data generated from the generator network G are used as inputs to the discriminator network D(748 × 748).

$G1$: The convolutional kernel size is set to 3 × 3 pixels, the learning step size, also known as the stride, is set to 1, the image size is transformed to 128 × 128 pixels.

$G2$: The convolutional kernel size is set to 3 × 3 pixels, the learning step size, also known as the stride, is set to 1, the image size is transformed to 256 × 256 pixels.

$G3$: The convolutional kernel size is set to 3 × 3 pixels, the learning step size, also known as the stride, is set to 1, the image size is transformed to 512 × 512 pixels.

$G4$: The convolutional kernel size is set to 3 × 3 pixels, the learning step size, also known as the stride, is set to 1, the image size is transformed to 1024 × 1024 pixels.

Output layer: the data is compressed to between –1 and 1 using the tanh activation function, at which point the image becomes 184 × 184.

Table 2 Discriminator network structure

Discriminator network (D)						
Layer	$G(z)$	$D1$	$D2$	$D3$	$D4$	Output
Kernel Num	748	1024	512	256	64	0/1
Kernel size	-	3	3	3	3	-
Stride	-	1	1	1	1	-

The following is the principle of discriminator network (D).

Input layer: The real data and the data generated by the generator network G(748 × 748) are used as inputs to the discriminator network D.

$D1$: The convolutional kernel size is set to 3 × 3 pixels, the learning step size, also known as the stride, is set to 1, the image size is transformed to 1024 × 1024 pixels.

$D2$: The convolutional kernel size is set to 3 × 3 pixels, the learning step size, also known as the stride, is set to 1, the image size is transformed to 512 × 512 pixels.

*D*3: The convolutional kernel size is set to 3 × 3 pixels, the learning step size, also known as the stride, is set to 1, the image size is transformed to 256 × 256 pixels.

*D*4: The convolutional kernel size is set to 3 × 3 pixels, the learning step size, also known as the stride, is set to 1, the image size is transformed to 64 × 64 pixels.

Output layer: gives true and false results using the Sigmoid activation function.

The input to generator *G* is random noise *z*.

Suppose the generator network *G* is $G(x) = x + 1$, let *z* be $P_G(x, \theta)$, is random noise that conforms to the normal distribution law, and after a nonlinear transformation, the generated output data from the generator network *G* follows a normal distribution with a mean of 1 and a variance of 1. The generator is used to learn the underlying distribution of the input noise vector *z*. Let the random noise *z* be $P_G(x, \theta)$, which is distributed as P_{data}, and let the distribution of real data *x* be $P_{data}(x)$, construct a likelihood function for each batch in training.

$$L = \prod_{i=1}^{m} P_G\left(x^i;\, \theta\right) \tag{3}$$

Eq. (3) is the formula for finding the optimal value of the likelihood function *L*. The equation involves the accumulation of P_G, where the accumulated part P_G represents random noise conforming to the normal distribution law.

$$\theta^* = \arg\max_{\theta} \prod_{i=1}^{m} P_G\left(x^i;\, \theta\right) = \arg\min_{\theta} KL\left(P_{data}(x) \middle\| P_G\left(x^i;\, \theta\right)\right) \tag{4}$$

Eq. (4) shows that θ^* is optimal when the distributions of $P_G(x, \theta)$ and $P_{data}(x)$ are similar, so the principle of generator *G* is to find an optimal θ to achieve the effect that P_G is closer to the distribution of $P_{data}(x)$.

Compared to traditional image synthesis algorithms, GAN has the following advantages in image synthesis.

1. The generator network only uses back propagation, not Markov chains.

2. The generator network uses random noise as input, does not require manual labeling of features, and exhibits superior synthesis ability compared to other generator models. The application of GANs can extend beyond image data generation to simulate the generation of other types of data.

3. The adjustments to the generator network come from feedback provided by the discriminator network rather than from the raw data.

3 IMPROVED DISCRIMINATOR NETWORK BASED ON RESIDUAL NETWORK FOR IMAGE SYNTHESIS

3.1 Principle of Residual Networks to Improve CNNs

At present, GANs have achieved remarkable advancements in image segmentation, synthesis, and recognition research. However, to attain higher accuracy, increasing the network depth is necessary. Nevertheless, this often brings about issues such as gradient disappearance, which impedes the model's ability to accurately capture image edges and local details. Consequently, this affects the efficacy of image segmentation, synthesis, and recognition [36, 37]. In this paper, a novel image synthesis method is proposed to address this issue by combining the residual network and generative adversarial network (GAN). The proposed approach harnesses the strengths of both techniques to enhance the quality and realism of generated images. Given that this method can acquire rich semantic information, we term it DENSE-GAN, and its model structure is detailed in Tab. 3.

Table 3 Network structure of DENSE-GAN-based image synthesis algorithm

Generator Network							Discriminator network					
Layer	*X*	*G*1	*G*2	*G*3	*G*4	*G*(*z*)	*G*(*z*)	*D*1	*D*2	*D*3*3	*fc*	Output
Kernel Num	64	128	256	512	1024	784	748	3	64	512	512	n
Kernel size	-	3	3	3	3	-	-	3	3	1	-	-
Stride	-	1	1	1	1	-	-	1	1	1	-	-

Tab. 3 presents partial data from a DENSE-GAN algorithm that enhances the discriminator network by substituting CNNs with residual networks. Fig. 5 illustrates the residual structure, and the final output layer employs three fully connected layers to extract identity information. G provides the specific structure of the improved discriminator network utilizing residual networks.

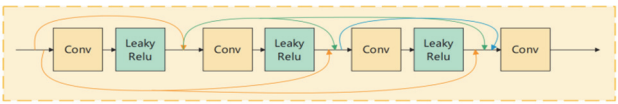


Figure 5 Structure of the dense block

Input layer: The data generated from the generator network *G* and the output data are used as inputs to the discriminator network *D* (748 × 748).

*D*1: The convolutional kernel size is set to 3 × 3 pixels, the learning step size, also known as the stride, is set to 1, the image size is transformed to 3 × 3 pixels.

*D*2: The convolutional kernel size is set to 3 × 3 pixels, the learning step size, also known as the stride, is set to 1, the image size is transformed to 64 × 64 pixels.

*D*3: The convolutional kernel size is set to 3 × 3 pixels, the learning step size, also known as the stride, is set to 1, the image size is transformed to 512 × 512 pixels.

fc: The three fully connected layers used here output three feature vectors *G*(*z*), local details of the image, and overall contour information.

Output layer: gives true and false (0/1) results using the Sigmoid activation function.

Generative adversarial networks (GANs) facilitate photorealistic image synthesis and editing (Meng et al., 2024). Following the principle of generative adversarial networks, we aim to minimize the loss of *G* and maximize

the loss of D. Only when the two networks reach equilibrium do the distribution laws of $G(z)$ resemble those of real data X.

In this chapter, the optimization of adversarial networks for end-to-end training is carried out using mean squared error (MSE), structural similarity (SSIM), and adversarial loss. The objective is to enhance the performance and convergence of the generator and discriminator networks within the generative adversarial network (GAN) framework. Furthermore, the training process involves utilizing a feedback signal to adjust the parameters of the generator network through back-propagation. The output of the generator network is then fed into the discriminator network, which continues its training. Throughout this iterative process, the two networks gradually reach equilibrium, where their outputs converge to produce images that appear realistic. Eq. (5) defines the specific loss function employed in this process.

3.1.1 Against Loss

We can utilize adversarial loss to optimize generator networks, making the images they produce more realistic. Eq. (5) represents an adversarial loss function. N represents the quantity of processed data, where $D(x)$ denotes the discriminant neural network that outputs 0 or 1, and $D(G(z))$ represents the assessment of the authenticity of the generated data.

$$L_{adv} = \frac{1}{N}\sum_{i=1}^{N}\log\left(1 - D\left(G\left(z\right)\right) + \log D\left(x\right)\right) \tag{5}$$

3.1.2 Generator/Discriminator Loss Function

Eq. (6) and Eq. (7) represent the loss functions of the discriminator network and the generator network, respectively. In Eq. (8), α represents the equilibrium coefficient of the generator network, and β represents the equilibrium coefficient of the discriminator. Eq. (8) introduces additional variables: $\lambda 1$ represents the overall information data of the image, $\lambda 2$ represents the local information of the image, and $\lambda 3$ represents the identity information of the image. L_m and L_n denote the mean and variance of the dataset (x, y), respectively.

$$L_G = \frac{1}{N}\alpha L_{adr}\left(G\left(z\right) + \beta\left(L_{MSE} + L_{SSMM} + L_{adv}\right)\right) \tag{6}$$

$$\mathrm{L_D} = \frac{1}{N}\alpha L_{adr}\left(G\left(z\right) + L_{adr}\left(x\right)\right) + \beta\left(L_{adr}\left(G\left(z\right)\right) + L_{adr}\left(G\left(x\right)\right)\right) \tag{7}$$

$$L_{adr} = \frac{1}{N}\sum_{i=1}^{N} L_{sof\max} + \lambda_1 L_m + \lambda_2 L_n + \lambda_3 \tag{8}$$

3.1.3 Improved Generative Adversarial Network Optimization Objective Function

Eq. (6) represents the loss function of the improved generative adversarial network. Eq. (6) to Eq. (9) illustrate how the generators and discriminators interact, reaching equilibrium when the generators are neither the smallest nor the largest discriminators, which is when the model performs optimally.

$$\arg\min_{G}\ \max_{D}\left(L_G + L_D\right) \tag{9}$$

3.2 Building a Network Model

Fig. 6 depicts the structure of DENSE-GAN, highlighting enhancements to the generator inputs, the discriminator network structure, and the discriminator outputs. The enhancements to the discriminator network D, utilizing the residual network, are as follows.

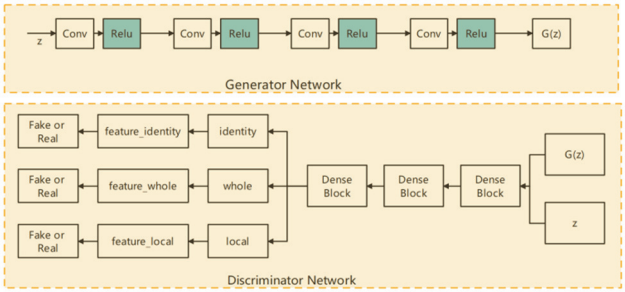


Figure 6 Structure of DENSE-GAN

3.2.1 Improving the Network Structure of Discriminators

Since the generator input is provided by the original discriminator network, the data it outputs are uncertain. Therefore, making a true or false judgment will lead to randomness in the discriminator network [38, 39]. Hence, the residual network can be utilized instead of CNN to optimize the discriminator network structure. Because images contain rich detailed information, the synthesis effect can be enhanced by evaluating the truth and falsity of three types of features through three relatively independent modules [40].

3.2.2 Input/Output Optimisation

Usually, we require a significant amount of time because training generative adversarial networks often demands abundant data [41]. We address the issue of slow convergence by enhancing the generator input, where x learns z based on real data. Consequently, the resulting synthetic data will more closely align with the distribution of x, regardless of whether it originates from aggregation or locality.

3.3 DENSE-GAN Training Steps

In this section, we will delve into strategies for enhancing discriminator networks by incorporating residual networks. We substitute the conventional CNN architecture with residual blocks and replace the original output with three fully connected layers of the generator. This alteration enables the discriminator to consider the overall contour structure and capture significant local details. This approach enhances image quality and convergence speed by analyzing the principal components of random noise.

Step 1: Dimensionality reduction of the random noise using principal component analysis to obtain the input z for the generator.

Step 2: Use z as input for the generator.

Step 3: Let the generator network perform nonlinear data transformation on data z through deconvolution operations, mapping z to a high-dimensional image $G(z)$.

Step 4: The input to the discriminator utilizes $G(z)$ and the real data x.

Step 5: If the distributions of $G(z)$ and x are similar, the loss function calculates the distance between $G(z)$ and x, as well as their respective distribution probabilities, and outputs the result directly. If there is a large difference, then step 6 is executed.

Step 6: The results of the discriminator are used as feedback information to guide the adjustment of generator network parameters, cycling through steps 1 - 5 until the two networks reach equilibrium and training is complete.

4 EXPERIMENTAL COMPARISON

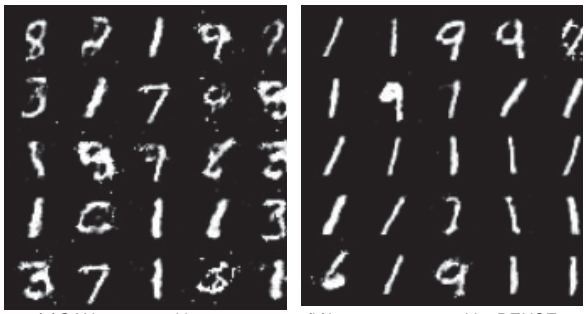
4.1 Experimental Parameters and Data

We selected the MNIST, Market-1501, CelebA, and DeepFashion datasets to validate the effectiveness of DENSE-GAN. The MNIST dataset comprises 10000 test samples and 60000 training samples [42]. Given the small number of features in images from this dataset, feature extraction becomes relatively straightforward. Consequently, the Market-1501 dataset, a publicly available dataset from 2015, is commonly utilized for this purpose. The training dataset contains over 10000 image IDs, with an average of 17 training data collected per person. The test dataset comprises nearly 20000 images, with an average of 26 test data obtained per person. These images exhibit low resolution, and pose, viewpoint, background, and lighting characteristics vary widely. CelebA is a dataset containing over 200000 face images, each labeled with multiple attributes and labels. The images are complex, featuring numerous attribute attributes, and demand higher requirements for network models in terms of feature extraction and image generation [43]. The DeepFashion dataset contains over 800000 images, 1000 attributes, and 50 categories. It also includes 300000 images with different poses and scenes. This large-scale dataset features high-resolution images (256 × 256) with clean backgrounds [44].

Table 4 GAN and DENSE-GAN training parameter settings

Training parameter settings and losses	Generating adversarial networks	DENSE-GAN
Batch size	128	128
Learning rate	0.0002	0.0002
Training epoch	100/1000/5000	100/1000/5000
Optimizer	Adam	Adam
Dropout	0.3	0/3
Activation functions	Relu	Relu
Dataset	MNIST/Market-1501	MNIST/Market-1501

Tab. 4 presents the specific parameter configurations for training the adversarial networks. The network's learning rate is determined based on the time taken to reach the global minimum [45]. A very small learning rate results in slow convergence, while a very large learning rate may impede network convergence [46]. For this scenario, a learning rate of 0.0002 is chosen, along with a batch size of 128. To enable comparison and analysis, the network is trained for three different numbers of epochs: 100, 1000, and 5000. The DENSE-GAN framework shares most of its parameters with the original generative adversarial network. However, a notable difference lies in the structure of the discriminator network [12, 47]. In DENSE-GAN, the traditional CNN in the discriminator is replaced with a residual network.

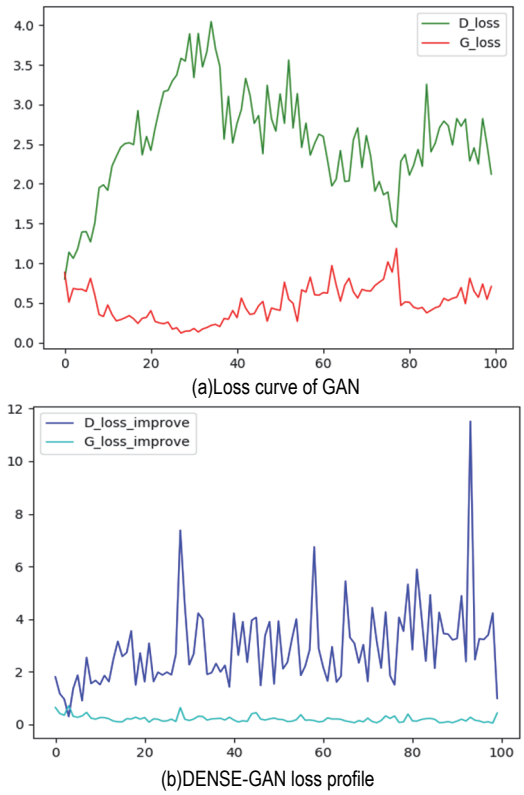


(a)GAN generated images (b)Images generated by DENSE-GAN

Figure 7 Comparison of experimental results before and after 100 iterations of improvement

4.2 Experimental Results and Analysis

The MNIST handwritten characters are initially analyzed using a pre-trained generative adversarial network (GAN). Subsequently, the network undergoes enhancements, and the analysis is repeated using the updated model. Based on the principles and loss function curves of the GAN, the empirical results of training the network for 100, 1000, and 5000 epochs are analyzed. The goal is to analyze and compare the network's performance before and after the improvement using these experimental results.



(a)Loss curve of GAN

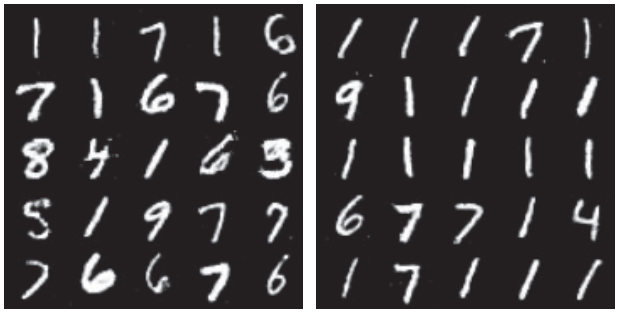
(b)DENSE-GAN loss profile

Figure 8 Plot of generator and discriminator loss before and after 100 iterations of improvement

Fig. 7 displays the results obtained from the experiments conducted after 100 training sessions. The left image in Fig. 7 corresponds to the output of the generative adversarial network (GAN) during the synthesis process, while the right image represents the synthesis by DENSE-GAN. From the results of the image synthesis, the figures synthesized on the left are blurred and contain a lot of noise, whereas the figures synthesized on the right are clearer and have less noise, although some individual figures still lack clarity.

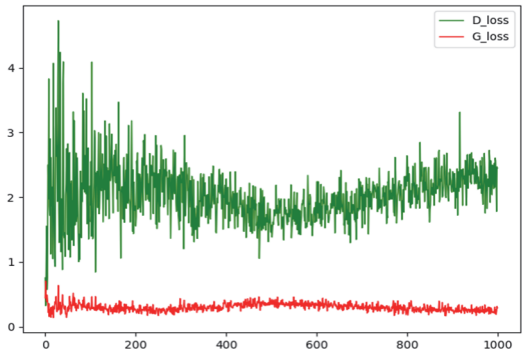
Fig. 8 shows the loss curves of the generator and discriminator, both before and after 100 iterations of the enhanced model. This demonstrates that the improved image synthesis outperforms the generative adversarial network. Due to the lack of network stabilization and the model's failure to reach a global minimum, the synthesized images suffer from low quality and lack of clarity.

The experimental results after 1000 training sessions are depicted in Fig. 9. A thorough analysis will now be conducted based on the loss function curve of the model training.

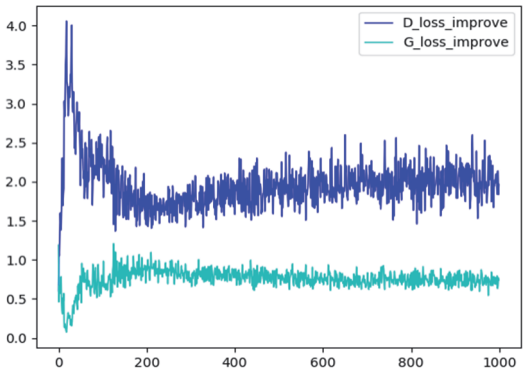


(a) GAN generated images N (b) Images generated by DENSE-GANe

Figure 9 Comparison of experimental results before and after 1000 iterations of improvement



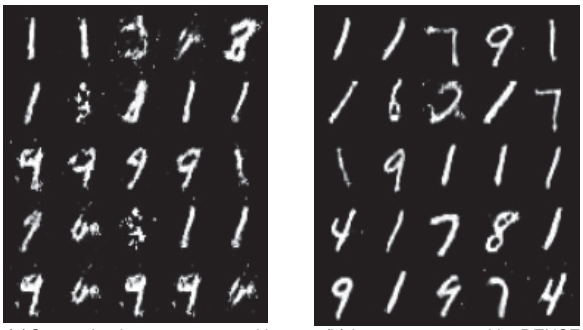
(a)GAN's loss curve



(b)DENSE-GAN loss profile

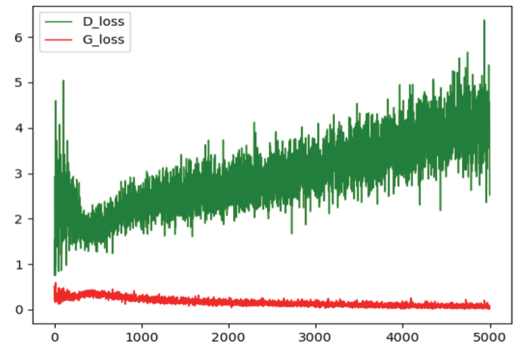
Figure 10 Plot of generator and discriminator loss before and after 1000 iterations of improvement

Fig. 10 illustrates the loss curves before and after 1000 training iterations. Upon examining the stability of the loss function, it becomes apparent that the loss curve depicted in Fig. 10b is more stable. This indicates a relatively smaller disparity between the losses of the generator network and the discriminator network. After 500 iterations of the generative adversarial network, the discriminator's loss fluctuates, while the generator's loss still stabilizes. Conversely, after 200 iterations of DENSE-GAN, both the discriminator and generator networks stabilize. The difference between the generator and discriminator losses was analyzed, and both networks were stable, with no particularly large fluctuations. Additionally, the difference between the generator and discriminator losses was only small at 1000 iterations. From the principles of the adversarial network, it can be concluded that the images synthesized by DENSE-GAN are of better quality.

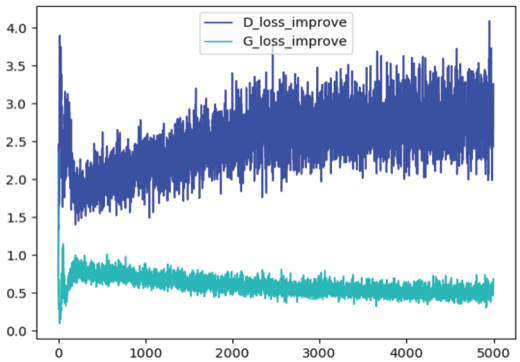


(a)Generating images generated by the adversarial network (b) Images generated by DENSE-GAN

Figure 11 Comparison of experimental results before and after 5000 iterations of improvement



(a) Generating loss curves for adversarial networks



(b) DENSE-GAN loss graph

Figure 12 Plot of generator and discriminator loss before and after 5000 iterations of improvement

Fig. 11 shows the effect of image synthesis before and after 5000 iterations of improvement. Analyzing the loss

function curve of the model training can provide further insights into the observed decrease in image quality after 5000 iterations compared to the effect observed after 1000 iterations.

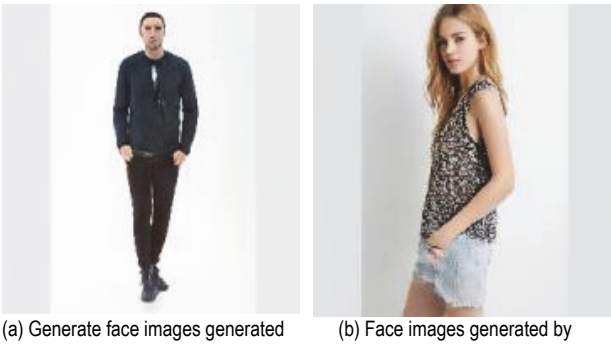
(a) Generate face images generated (b) Face images generated by DENSE-GAN by the adversarial network

Figure 13 Comparison of experimental results before and after 100 iterations of improvement

Fig. 12 shows the loss function graph pre and post-5000 iterations of enhancement. In the left graph, the loss of the generator stabilizes around 500 iterations, while the loss of the discriminator consistently increases. This contradicts the principle of generative adversarial networks, leading to a decline in the quality of the synthesized image. After 500 iterations, a noticeable reversal occurs in the loss values of the generator and discriminator networks on the right side. Additionally, the widening gap between the two losses suggests that the composite image is exhibiting exceptional performance in terms of the GAN training process.

By employing a modified version of the generative adversarial network, utilizing the same network model as previously mentioned, we proceeded to analyze and train the Market-1501 dataset. The experimental results were as follows.

In Fig. 13, the impact of the enhanced model can be observed by comparing the synthetic face images before and after 100 iterations. It is evident that at this stage, the model has not fully converged, resulting in blurred synthetic images. To enhance image synthesis quality, training iterations can be increased, and network parameters can be adjusted accordingly. Notably, the DENSE-GAN synthesis method outperforms the original GAN in image synthesis.



(a) Generate face images generated by the adversarial network (b) Face images generated by DENSE-GAN

Figure14 Comparison of experimental results before and after 5000 iterations of improvement

Fig. 14 shows the effect of the synthetic face images before and after 1000 iterations of improvement, at which point the model has not yet found the global minimum. After 1000 iterations, there is a noticeable improvement in the quality and sharpness of the composite image. It becomes evident that DENSE-GAN synthesis surpasses the performance of the original generative adversarial network. The enhancements made in the training process and network architecture have yielded better results in terms of image synthesis.

Table 5 Comparison of different network models in the Celeb A data set into image similarity

Network Architecture	100	200	300	400	500	1000	5000
GAN	1.910	2.924	2.765	2.991	1.442	1.849	2.088
MSG-GAN	0.251	0.224	0.234	0.237	0.227	0.233	0.241
DENSE-GAN	1.372	1.442	1.241	1.438	1.355	1.375	1.394

Upon analyzing the experimental results, a conclusion can be drawn that optimal synthesized image quality tends to occur when the generator and discriminator networks reach a stable state, with the generator's loss not excessively small and the discriminator's loss not excessively large.

To evaluate the effectiveness of the algorithm, image synthesis operations were performed on the CelebA dataset and the DeepFashion dataset. The changes in the loss function values of the GAN model, MSG-GAN model, and DENSE-GAN model on the CelebA dataset were recorded for iterations at 100, 200, 300, 500, 1000, and 5000. Additionally, the contrast values of the composite images were calculated before and after the synthesis process. A value of 1 indicates identical images, while a value closer to 1 implies a higher similarity between the composite and original images.

The experiment results conducted on the CelebA dataset are summarized in Tab. 5. From the data provided, the mean similarity value for GAN is 2.281, with the smallest similarity value being 1.442 and the largest similarity value being 2.991. For MSG-GAN, in the range of 100 to 5000 iterations, the average similarity value is 0.235, with the minimum similarity value being 0.224 and the maximum similarity value being 0.251. The synthesis of images on the CelebA dataset is improved with the DENSE-GAN model, as evidenced by its minimum similarity value of 1.241, maximum value of 1.442, and mean value of 1.374.

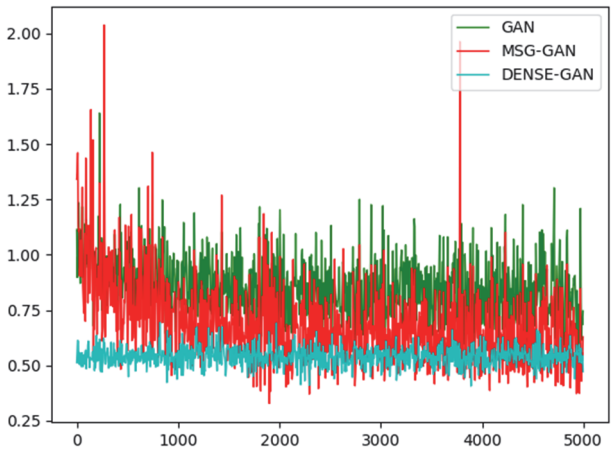


Figure15 Loss function curves for different network models at CelebA

Table 6 Comparison of different network models into image similarity in the DeepFashion dataset

Network Architecture	100	200	300	400	500	1000	5000
GAN	0.255	0.358	0.312	0.312	0.232	0.190	0.321
MSG-GAN	0.565	0.810	0.677	0.640	0.616	0.560	0.496
DENSE-GAN	0.696	0.970	0.809	0.960	0.712	1.375	0.870

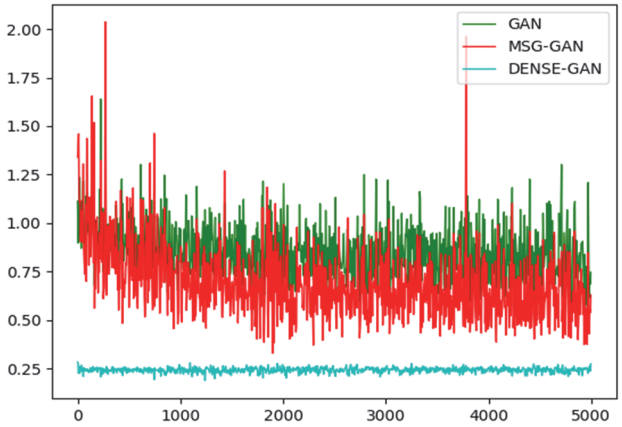


Figure 16 Loss function curves of different network models in DeepFashion

Fig. 15 shows the descent curves of loss for GAN, MSG-GAN, and DENSE-GAN. The loss curves of GAN and MSG-GAN networks exhibit considerable fluctuations around 100, 1000, and 5000 iterations, whereas DENSE-GAN stabilizes around 500 iterations with minimal loss fluctuation. The DENSE-GAN model demonstrates greater stability, leading to faster convergence on the CelebA dataset.

Tab. 6 presents the results of experiments conducted on the DeepFashion dataset. Upon examining the data, it is evident that the GAN model achieves a minimum similarity value of 0.190. For iterations ranging from 100 to 5000, the average similarity value is 0.283. Regarding MSG-GAN, the minimum similarity value is 0.496, with an average similarity value of 0.623 for iterations from 100 to 5000. The maximum similarity value recorded is 0.810. Based on the data in the table, it can be deduced that the DENSE-GAN model demonstrates superior image synthesis capabilities on the DeepFashion dataset, with a mean value of 0.818, a minimum similarity value of 0.696, and a maximum value of 0.970.

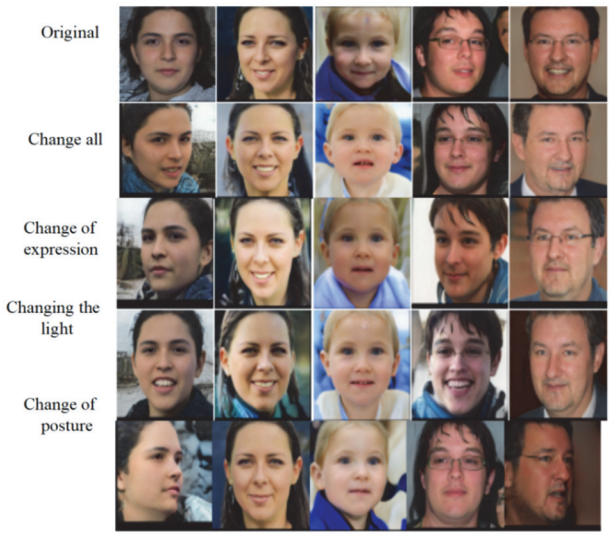


Figure17 Images and effects

Fig. 16 illustrates the decreasing loss curves of GAN, MSG-GAN, and DENSE-GAN. The loss curves of GAN and MSG-GAN networks exhibit significant fluctuations in model stability around 100, 1000, and 5000 iterations, while the loss of the DENSE-GAN network is minimal and tends to stabilize. Regarding convergence speed, the GAN and MSG-GAN networks stabilize only after approximately 1000 iterations, with relatively large fluctuations afterward, whereas the DENSE-GAN network stabilizes after about 200 iterations without significant fluctuations.

The DeepFashion dataset was utilized to generate controlled face images with varying lighting, poses, expressions, and environments. These variations were meticulously designed to create synthetic images while preserving the subject's identity.

Analyzing the experimental results depicted in Fig. 17, it is evident that the DENSE-GAN network proficiently preserves the image's identity. Moreover, it enhances the quality of image synthesis by managing factors such as expression, lighting, and posture.

4.3 Summary of this Chapter

In our approach, random noise serves as the input to the adversarial network, which then generates an output image. The abundance of features in an image poses difficulty in their extraction, leading to diminished clarity and quality in synthesized images. This paper suggests mitigating this issue by employing the residual generative adversarial network technique, capitalizing on the benefits offered by residual networks. Specifically, the proposed method utilizes three fully connected layers within the output layer to extract image features, while also extracting features from the discriminator network. By leveraging these extracted features, it aims to improve the synthesis process, resulting in higher quality and clearer synthesized images. To evaluate the authenticity of synthetic images, the paper employs the calculation of Euclidean distance. The obtained distance serves as a measure for determining the fidelity of the synthesized image. By analyzing this result, adjustments are made to the parameters of the generative adversarial network, specifically by feeding it back to the generator network. Multiple experiments have been conducted, yielding promising findings regarding the effectiveness of this method. The results indicate a significant enhancement in the quality and clarity of the synthesized images following the implementation of this approach. We will continue to refine the algorithm, aiming to achieve higher precision, more realistic and faster image generation, and improved processing speed.

Acknowledgment

This work is funded by the National Natural Science Foundation of China(62472149) and Hubei Provincial Education Science Planning Project(2022GB030).

5 REFERENCES

[1] Samah, S. B., Trung, N. L., & Tam, V. N. (2023). Image synthesis: a review of methods, datasets, evaluation metrics,

and future outlook. *Artificial Intelligence Review*, *56*(10), 10813-10865. https://doi.org/10.1007/s10462-023-10434-2

[2] Yan, L., Fu, J., Wang, C., Zhiwei, Y., Hongwei, C., & Hefei, L. (2021). Enhanced network optimized generative adversarial network for image enhancement. *Multimedia Tools and Applications, 80*(9), 14363-14381, https://doi.org/10.1007/s11042-020-10310-z

[3] Alam, F., Sang Ko, H., Lee, H. F., & Yuan, C. (2023). Deep Learning Approach for Volume Estimation in Earthmoving Operation. *International Journal of Industrial Engineering and Management*, *14*(1), 41-50. https://doi.org/10.24867/IJIEM-2023-1-323

[4] Lu, H. T. & Zhang, Q. C. (2016). A Review of the Application of Deep Convolutional Neural Networks in Computer Vision. *Data Acquisition and Processing, 31*(1), 1-17.

[5] Zhang, X. W., Xuan, C. Z., Ma, Y. H., Liu, H. Y., & Xue, J. (2024). Lightweight model-based sheep face recognition via face image recording channel. *Journal of animal science*, *102*. https://doi.org202410.1093/JAS/SKAE066

[6] Maican, C. I., Sumedrea, S., Tecau, A., Nichifor, E., Chitu, I. B., Lixandroiu, R., & Bratucu, G. (2023). Factors Influencing the Behavioural Intention to Use AI-Generated Images in Business: A UTAUT2 Perspective with Moderators. *Journal of Organizational and End User Computing (JOEUC)*, *35*(1), 1-32. https://doi.org/10.4018/JOEUC.330019

[7] Li, J., Wu, L., Qi, J., Zhang, Y., Wu, Z., & Hu, S. (2023). Determinants Affecting Consumer Trust in Communication with AI Chatbots: The Moderating Effect of Privacy Concerns. *Journal of Organizational End User Computing (JOEUC)*, *35*(1), 1-4. https://doi.org/10.4018/JOEUC.328089

[8] Shanshan, P., Chao, M., Haitao, Z., & Kun, L. (2022). An Evaluation System Based on User Big Data Management and Artificial Intelligence for Automatic Vehicles. *Journal of Organizational End User Computing (JOEUC)*, *34*(10), 1-21. https://doi.org/10.4018/JOEUC.309135

[9] Zhu, Z. L., Rao, Y., Wu, Y., et al. (2019). Research Progress on Attention Mechanism in Deep Learning. *Journal of Chinese Information*, *33*(06), 1-11.

[10] Chen, L. C., Papandreou, G., Kokkinos, I., et al. (2017). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transac-tions on pattern analysis and machine intelligence*, *40*(4), 834-848.

[11] Pu, C., Song, R., Tylecek, R., Li, N., & Fisher, R. B. (2019) Sdf-man: Semi-supervised disparity fusion with multi-scale adversarial networks, *Remote Sens., 11*(5), 487. https://doi.org/10.3390/rs11050487

[12] Dieste, A. G., Arguello, F., & Heras, D. B. (2023). ResBaGAN: A Residual Balancing GAN with Data Augmentation for Forest Mapping. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *16*, 6428-6447. https://doi.org/10.1109/JSTARS.2023.3281892

[13] Sadeghi, Z. & Testolin, A. (2017). Learning representation hierarchies by sharing visual features: a computational investigation of Persian character recognition with unsupervised deep learning. *Cognitive processing*, *18*(3), 273-284. https://doi.org/10.1007/s10339-017-0796-7

[14] Zhang, C. P. & Su, G. D. (2000). A review of face recognition technology. *Chinese Journal of Graphics*, *5*(11), 885-894. https://doi.org/10.11834/jig.20001101

[15] Zhang, C. Q. (2016). Research on Classification Algorithm for Natural Scene Text and Non-Text Images. *Hubei: Huazhong University of Science and Technology*, 617-626. https://doi.org/10.1109/COMPSAC57700.2023.00087

[16] He, K. M., Zhang, X. Y., Ren, S. Q., & Sun, J. (2015) Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *37*(9), 1904-1916. https://doi.org/10.1109/TPAMI.2015.2389824

[17] Ren, X., Zhang, F., Sun, Y., & Liu, Y. (2024). A Novel Dual-Channel Temporal Convolutional Network for Photovoltaic Power Forecasting. *Energies, 17*(3), 698. https://doi.org/10.3390/en17030698

[18] Kang, H., Lee, S. Y., & Chui, C. K. (2009). Flow-Based Image Abstraction. *IEEE transactions on visualization and computer graphics, 15*(1), 62-76. https://doi.org/10.1109/TVCG.2008.81

[19] Zhao, J. L., Liu, Z. B., Sun, Q. X., Li, Q., Jia, X. Y., & Zhang, R. M. (2022). Attention-based dynamic spatial-temporal graph convolutional networks for traffic speed forecasting. *Expert Systems with Applications, 204*, 117511. https://doi.org/10.1016/j.eswa.2022.117511

[20] Kim, Y., Soh, J. W., & Cho, N. I. (2020). AGARNet: Adaptively Gated JPEG Compression Artifacts Removal Network for a Wide Range Quality Factor. *IEEE Access, 8*, 20160-20170. https://doi.org/10.1109/ACCESS.2020.2968944

[21] Khamaiseh, S. Y., Bagagem, D., Al-Alaj, A., Mancino, M., Alomari, H., & Aleroud, A. (2023). Target-X: An Efficient Algorithm for Generating Targeted Adversarial Images to Fool Neural Networks. *Proceedings-International Computer Software and Applications Conference*, 617-626. https://doi.org/10.1109/COMPSAC57700.2023.00087

[22] Zhang, P., Tang, J., Zhong, H., Ning, M., & Fan, Y. (2022). Rotated Target Recognition of Sonar Images via Convolutional Neural Networks with Rotated Inputs. *Proceedings of SPIE - The International Society for Optical Engineering,* 12342. https://doi.org/10.1117/12.2644531

[23] Yang, J., Shao, H., Qin, H., Xie, Y., & Huang, L. (2023). Instance-level image synthesis method based on multi-scale style transformation. *Proceedings of SPIE - The International Society for Optical Engineering,* 12705. https://doi.org/10.1117/12.2680108

[24] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Bing, X., David, W. F., Sherjil, O., Aaron, C., & Yoshua, B. (2020). Generative Adversarial Networks. *Communications of the ACM*, *63*(11), 139-144. https://doi.org/10.1145/3422622

[25] Wang, K. F., Gou, C., Duan, Y. J., et al. (2017). Research Progress and Prospect of Generative Adversarial Networks (GANs). *Acta Automatica Sinica*, *43*(3), 321-332.

[26] Ponraj, A. & Canessane, R. A. (2023). Deep Learning with Histogram of Oriented Gradients- based Computer-Aided Diagnosis for Breast Cancer Detection and Classification. *Proceedings-2023 3rd International Conference on Smart Data Intelligence*, *ICSMDI 2023*, 527-532. https://doi.org/10.1109/ICSMDI57622.2023.00099

[27] Vinicius, L. T. de S., Bruno, A. D. M., Harlen, C. B., Joãot, P. G. (2023). A review on Generative Adversarial Networks for image generation. *Computers & Graphics*, *114,* 13-25. https://doi.org/10.1016/j.cag.2023.05.010

[28] Ogundokun, R. O., Maskeliūnas, R., Misra, S., & Damasevicius, R. (2022). A Novel Deep Transfer Learning Approach Based on Depth-Wise Separable CNN for Human Posture Detection. *Information (Switzerland)*, *13*(11). https://doi.org/10.3390/info13110520

[29] Wen, J., Varol, E., Sotiras, A., Zhijian, Y., Ganesh, B. C., Guray, E., Haochang, S., Ahmed, A., Gyujoon, H., Dominic, B. D., Alessandro, P., Paola, D., Rene, S. K., Hugo. G. S., Marcus, V. Z., Eva, M., Geraldo, F. B., Benedicto, C. F., Romero-Garcia, R., Christos, P., & Christos, D. (2022). Multi-scale semi-supervised clustering of brain images: Deriving disease subtypes. *Medical Image Analysis, 75,* 102304. https://doi.org/10.1016/j.media.2021.102304

[30] Hamghalam, M. & Simpson, A. L. (2024). Medical image synthesis via conditional GANs: Application to segmenting brain tumours. *Computers in Biology and Medicine, 170*. https://doi.org/10.1016/j.compbiomed.2024.107982

[31] Tang, H., Sun, G., Sebe, N., & Van Gool, L. (2023). Edge Guided GANs With Multi-Scale Contrastive Learning for

Semantic Image Synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 45*(12), 14435-14452. https://doi.org/10.1109/TPAMI.2023.3298721

[32] Jung, E., Luna, M., & Park, S. H. (2021). Conditional GAN with an Attention-Based Generator and a 3D Discriminator for 3D Medical Image Generation. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12906 LNCS, 318-328. https://doi.org/10.1007/978-3-030-87231-1_31

[33] Durall, R., Frolov, S., Hees, J., Raue, F., Pfreundt, F. J., Dengel, A., & Keuper, J. (2021). Combining Transformer Generators with Convolutional Discriminators. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12873 LNAI, 67-79. https://doi.org/10.1007/978-3-030-87626-5_6

[34] Guo, M. W., Zhao, Y. Z., Xiang, J. P., et al. (2014). Overview of target detection algorithms based on support vector machines. *Control and Decision, 2*, 193-200.

[35] Mohana, P. P. (2022). A Survey of Modern Deep Learning based Generative Adversarial Networks (GANs). *Proceedings - 6th International Conference on Computing Methodologies and Communication, ICCMC 2022*, 1146-1152. https://doi.org/10.1109/ICCMC53470.2022.9753782

[36] Li, X. X., Xie, X., Li, B., et al. (2021). Application of Generative Adversarial Networks in Medical Image Processing. *Computer Engineering and Applications, 57*(18), 24-37.

[37] Vijay, B., Alex, K., & Roberto, C. (2017)SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*(12), 2481-2495. https://doi.org/10.1109/TPAMI.2016.2644615

[38] Jiang, T., Li, Y. S., Xie, W. Y., & Du, Q. (2020). Discriminative Reconstruction Constrained Generative Adversarial Network for Hyperspectral Anomaly Detection. *IEEE Transactions on Geoscience and Remote Sensing, 58*(7), 4666-4679. https://doi.org/10.1109/TGRS.2020.2965961

[39] Wang, G. G., Guo, T., Yu, Y., et al. (2019). Multilayer Perceptron Generative Adversarial Network Based on Semi-Supervised Learning. *Journal of Small & Miniature Computers, 40*(11), 2297-2303.

[40] Baswaraj, D. & Srinivas, K. (2024). Usage of Generative Adversarial Network to Improve Text to Image Synthesis. *Lecture Notes in Electrical Engineering, 1106*, 185-196. https://doi.org/10.1007/978-981-99-7954-7_17

[41] Bukowski, M., Antoniuk, I., & Kurek, J. (2023). Improved efficient capsule network for Kuzushiji-MNIST benchmark dataset classification. *Bulletin of the Polish Academy of Sciences: Technical Sciences, 71*(6). https://doi.org/10.24425/bpasts.2023.147338

[42] Saadna, Y., Behloul, A., & Mezzoudj, S. (2019). Speed limit sign detection and recognition system using SVM and MNIST datasets. *Neural Computing and Applications, 31*(9), 5005-5015. https://doi.org/10.1007/s00521-018-03994-w

[43] Wu, H., Bezold, G., Günther, M., Boult, T., King, M. C., & Bowyer, K. W. (2023). Consistency and Accuracy of CelebA Attribute Values. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 3258-3266. https://doi.org/10.1109/CVPRW59228.2023.00328

[44] Li, Z., Xia, P., Tao, R., Niu, H., & Li, B. (2023). A New Perspective on Stabilizing GANs Training: Direct Adversarial Training. *IEEE Transactions on Emerging Topics in Computational Intelligence*, *7*(1), 178-189. https://doi.org/10.1109/TETCI.2022.3193373

[45] Madhavan, R., Garg, R., Wadhawan, K., & Mehta, S. (2023). CFL: Causally Fair Language Models Through Token-level Attribute Controlled Generation. *Proceedings of the Annual Meeting of the Association for Computational Linguistic*s. Association for Computational Linguistics (ACL), 11344-11358. https://doi.org/10.18653/v1/2023.findings-acl.720

[46] Daphal, S. D. & Koli, S. M. (2024). Enhanced Classification of Sugarcane Diseases Through a Modified Learning Rate Policy in Deep Learning. *Traitement Du Signal, 41*(1), 441-449. https://doi.org/10.18280/ts.410138

[47] Yan, L., Sheng, M., Wang, C., Gao, R., & Yu, H. (2021). Hybrid neural networks based facial expression recognition for smart city. *Multimedia Tools and Applications, 81*, 319-342. https://doi.org/10.1007/ s11042-021-11530-7

Contact information:

Zixuan CHEN
School of Computer Science,
Hu Bei University of Technology,
Wuhan, 430086, China

Lingyu YAN
(Corresponding author)
School of Computer Science,
Hu Bei University of Technology,
Wuhan, 430086, China
E-mail: raslut281@21cn.com

Chunzhi WANG
School of Computer Science,
Hu Bei University of Technology,
Wuhan, 430086, China

Zhiwei YE
School of Computer Science,
Hu Bei University of Technology,
Wuhan, 430086, China