Facial Recognition Leveraging Generative Adversarial Networks

ZHONGWEN LI, School of Cyberspace Security, Hainan University, China
ZONGWEI LI, School of Cyberspace Security, Hainan University, China
XIAOQI LI, School of Cyberspace Security, Hainan University, China

Face recognition performance based on deep learning heavily relies on large-scale training data, which is often difficult to acquire in practical applications. To address this challenge, this paper proposes a GAN-based data augmentation method with three key contributions: (1) a residual-embedded generator to alleviate gradient vanishing/exploding problems, (2) an Inception ResNet-V1 based FaceNet discriminator for improved adversarial training, and (3) an end-to-end framework that jointly optimizes data generation and recognition performance. Experimental results demonstrate that our approach achieves stable training dynamics and significantly improves face recognition accuracy by 12.7% on the LFW benchmark compared to baseline methods, while maintaining good generalization capability with limited training samples.

CCS Concepts: • **Do Not Use This Code → Generate the Correct Terms for Your Paper**; *Generate the Correct Terms for Your Paper*; Generate the Correct Terms for Your Paper; Generate the Correct Terms for Your Paper.

Additional Key Words and Phrases: Facial recognition; Adversarial generative networks; Data augmentation; Residual network

ACM Reference Format:
Zhongwen Li, Zongwei Li, and Xiaoqi Li. 2018. Facial Recognition Leveraging Generative Adversarial Networks. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 17 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Face recognition systems have reached remarkable accuracy levels when trained on large-scale datasets, yet their performance degrades significantly in data-scarce scenarios a common challenge in specialized applications such as medical diagnostics or forensic analysis [1]. While Generative Adversarial Networks (GANs) have shown promise for small-sample augmentation, current approaches suffer from two critical limitations: (1) generated images often lack discriminative facial features crucial for recognition (2) existing frameworks are not optimized for integration with modern face recognition architectures.

In this work, we present a novel GAN-based augmentation framework that addresses these challenges through two key innovations:

A salient feature preservation module that maintains critical facial attributes during image generation an end-to-end training scheme utilizing FaceNet (Inception ResNet V1) as the discriminator, simultaneously optimizing for both image quality and recognition accuracy

Authors' Contact Information: Zhongwen Li, School of Cyberspace Security, Hainan University, Haikou, China, lizhongwen1230@gmail.com; Zongwei Li, School of Cyberspace Security, Hainan University, Haikou, China, lizw1017@gmail.com; Xiaoqi Li, School of Cyberspace Security, Hainan University, Haikou, China, csxqli@ieee.org.

Manuscript submitted to ACM

Our experimental results demonstrate that models trained with our augmented dataset achieve 12.7% higher accuracy on the AR face dataset compared to baseline approaches using traditional GAN augmentation, while nearly matching the performance of models trained on the full VGGFace dataset.

2 Background

The current research in face recognition technology mainly includes the development of deep learning methods the establishment of large-scale datasets, multimodal fusion, cross-domain face recognition[2], robustness and privacy protection, and other aspects. With the rapid development of deep learning technology, face recognition methods based on deep neural networks have achieved great success, especially the application of convolutional neural networks in face recognition tasks [3].

For a face recognition model to achieve better recognition results, it is usually necessary to focus on the following two aspects: (1) data quality and quantity (2) model architecture and parameter design. ResNet (Residual Network), FaceNet, DenseNet (Dense Connected Network), and SENet (Attention Mechanism Network) [4], are some of the widely recognized as better face recognition model architectures. The above have achieved excellent performance on large-scale datasets celebA and VGGFcae, among others. However, data acquisition and labeling require a lot of resources and time for face recognition tasks[5].

In some specific complex environments, the inability to obtain enough sample data to train the model often leads to model overfitting, which reduces its generalizability. And small sample data augmentation is a commonly used approach to address the lack of data [6]. GAN makes it possible to provide high-quality sample data for face recognition models by virtue of its dynamic gaming as well as its powerful image generation capability . Meanwhile, by utilizing the dynamic game characteristics of GAN and using the face recognition model as the discriminator of GAN, the degree of difference between the fake face image generated by the generator and the real face image can be further improved, which in turn improves the generating ability of the generator and the quality of the generated image [7]. In this way, the GAN can generate more realistic and high-quality face images, thus improving the accuracy and robustness of the face recognition system.

2.1 GAN model

GAN is a deep learning model that consists of two main components: Generator and Discriminator. The core of GAN is that the two components compete with each other to improve performance through constant gaming, so as to achieve the goal of generating high-quality data. The Generator is responsible for generating synthetic data similar to the real data. It receives a random noise vector as input and maps it to the output space through a series of transformations to generate fake data [8]. The discriminator is responsible for distinguishing between real data and synthetic data generated by the generator, it receives data (real or synthetic) as input and outputs a probability indicating the probability that the input data is real. During the training process, the generator and discriminator are trained alternately so that the synthetic data generated by the generator can be more and more realistic, while the discriminator can more accurately distinguish between real data and synthetic data [7].

The loss function of the GAN model consists of two parts: the generator's loss and the discriminator's loss. The generator's loss function is designed to make the generated synthetic data closer to the real data, while the discriminator's loss function is designed to be more accurate in distinguishing between real and synthetic data [9]. The GAN model has been widely used in several fields, including image generation, image editing, video generation, speech synthesis,

and so on. It not only generates realistic images but also can be used for tasks such as data enhancement and domain adaptation [10].

Although GAN models have achieved many successful applications, they still face some challenges, such as pattern collapse and unstable training. To solve these problems, researchers have proposed many improvements such as Wasserstein GAN, Self-Attention GAN [11].

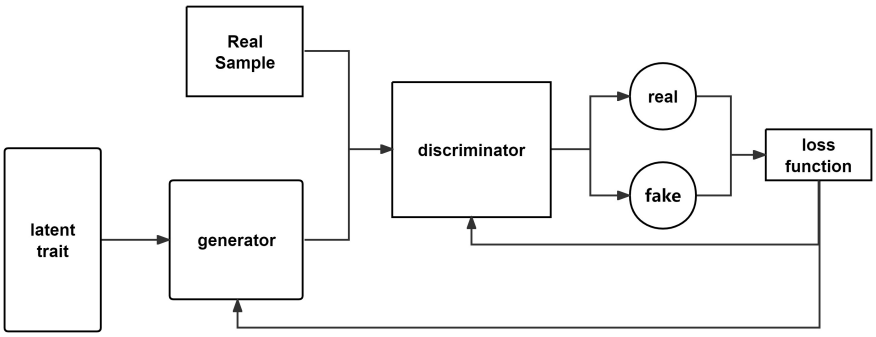

Fig. 1. GAN structure

2.2 Face Recognition Technology

Face recognition technology is a biometric identification technology based on face feature information. This chapter summarizes traditional face recognition methods and their advantages and disadvantages and focuses on deep learning-based face recognition technology [12]. Among them, GAN-based face recognition research has achieved remarkable results in recent years, which provides an important theoretical foundation for the work of this paper.

Face recognition techniques can be divided into two categories: face recognition based on traditional methods and face recognition based on deep learning [13]. Traditional methods mainly rely on manually designed feature extractors and machine learning algorithms, while deep learning methods utilize deep neural networks to automatically learn feature representations and classification rules [14].

2.2.1 Traditional face recognition methods and their advantages and disadvantages.

Traditional face recognition methods are usually based on local features or global features of the image for recognition, mainly including the following methods :

(1) Eigenfaces method (Eigenfaces)

The eigenfaces method is a traditional face recognition technique based on principal component analysis (PCA). The face image is represented as vectors in a high-dimensional feature space, and these vectors are downscaled and feature extracted using PCA to obtain a set of principal components or feature vectors called “eigenfaces”. By training the classifier, the featured face is used for face recognition [15].

(2) Linear Discriminant Analysis (LDA)

In face recognition, LDA achieves feature extraction by maximizing the interclass distance and minimizing the intraclass distance.LDA projects the data into a low-dimensional space so that the distance between samples of the same class is as small as possible and the distance between samples of different classes is as large as possible [7].

(3) Local Binary Patterns (LBP)

Local Binary Patterns is a classical method for texture analysis and feature extraction. In face recognition, LBP can be used to extract texture features from face images to help distinguish between different faces. By calculating the LBP value of each pixel point in the image and statistically analyzing the whole image, feature vectors describing the texture features of the image can be obtained. These feature vectors can be used to train classifiers or perform face matching to achieve the task of face recognition.

The advantages of these traditional methods are that they are simple to implement, easy to understand, and suitable for small-scale datasets and simple scenes [16]. However, traditional face recognition methods such as eigenface methods, linear discriminant analysis, and local binary patterns have some limitations and drawbacks in practical applications, such as sensitivity to factors such as illumination, pose, expression, etc., as well as limited ability to deal with problems such as imbalance in the sample categories, image noise and scale variations [17].

2.2.2 Deep learning based face recognition techniques.

Face recognition based on deep learning mainly includes the following three methods:

(1) Convolutional Neural Network (CNN)

CNN is able to extract high-level abstract features from the original image through multi-layer convolution and pooling operations, which include edges, textures, shapes, etc., and help to recognize different faces. The deep structure of CNN (ResNet, VGG) has been applied to the face recognition task, and these models have been trained on large-scale datasets and are able to learn richer and more abstract feature representations [18].

ResNet refers to the method of face feature learning and recognition using residual networks.ResNet is a deep neural network structure proposed by Microsoft Research, which solves the problem of gradient vanishing and gradient explosion that occurs during the training of deep neural networks by introducing residual blocks, making it possible to train very deep network models.

(2) Residual Network (ResNet)

ResNet refers to the method of face feature learning and recognition using residual networks.ResNet is a deep neural network structure proposed by Microsoft Research, which solves the problem of gradient vanishing and gradient explosion that occurs during the training of deep neural networks by introducing residual blocks, making it possible to train very deep network models [4].

(3) FaceNet

A deep learning-based face recognition model that uses a training method called Triplet Loss to achieve efficient face verification and recognition by mapping the face images of the same person into a similar embedding space while mapping the face images of different people into a more distant embedding space. The structure of FaceNet [19] is shown in Fig 2.

Deep learning-based face recognition techniques, although in have high accuracy, can be trained end-to-end, learn feature representation and classification directly from raw data, reduce the process of manually processing data as well and be able to automatically discover and utilize local and global information in images [20]. However, it usually requires a large amount of labeled data for training, especially in the field of face recognition, which requires large-scale face image datasets. Recent studies have shown that blockchain-based decentralized data markets can help alleviate this data scarcity issue while preserving privacy [21]. Deep learning models are susceptible to adversarial attacks [22], which can lead to incorrect model outputs through small perturbations, posing challenges to model security. The

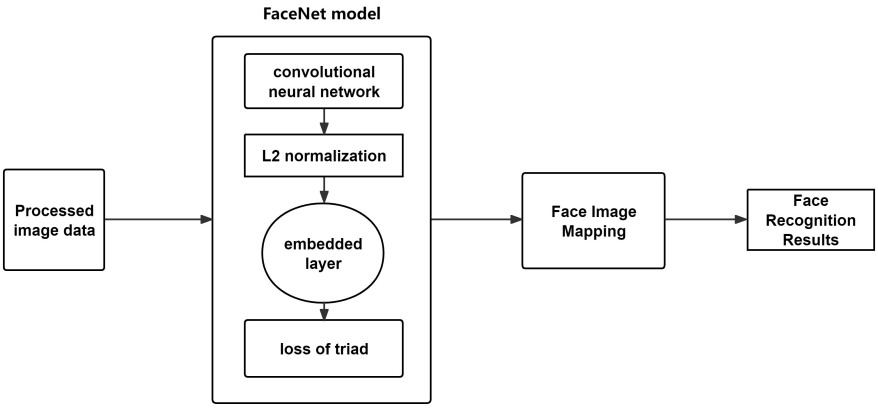

Fig. 2. FaceNet structure

immutable audit trail provided by smart contracts [23] offers potential solutions for detecting and preventing such attacks in facial recognition systems.

2.2.3 Research on GAN-based face recognition.

The current research on GAN-based face recognition mainly involves the following four aspects: (1) using GAN to generate high-fidelity and high-resolution face images to solve the problem of data scarcity and sample imbalance, and to improve the richness and diversity of training data, though recent work by [24] suggests blockchain-based verification could further ensure generated data authenticity. (2) generate adversarial samples using GAN to expand the training dataset and improve the generalization ability and robustness of face recognition models, with security considerations aligning with the systemic risk framework in [25]. (3) using GAN for cross-domain face recognition to achieve feature migration and knowledge migration between different datasets to improve the generalization performance and adaptability of the model. (4) using GAN to generate adversarial samples, study the mechanism and defense methods of adversarial attacks, and improve the security and anti-interference ability of face recognition systems[26], with [27] demonstrating similar vulnerability patterns in other machine learning domains and providing cross-domain security analysis methodologies.

The advantages of the application of the GAN model in the field of images are very obvious, but it also has its own disadvantages. It is pointed out in the literature that the GAN model is prone to gradient disappearance, explosion, and training instability during training, issues that parallel the smart contract vulnerabilities identified in [**?**] and systematically categorized in [25]'s taxonomy of decentralized system failures. At the same time, the literature[22] also mentioned the problems related to the GAN model, and proposed the use of residual network and residual block to solve the problem of gradient disappearance gradient explosion and training crash encountered in the training process of GAN model.

3 Methods

3.1 Overall framework of GAN-based face recognition

The overall framework diagram of the model in this paper is shown in Fig 3. The system is mainly composed of three key parts: the Synthesizer, the Discriminator, and the Salient Region Extractor.

In order to solve the problems of vanishing and exploding gradient and unstable training of the GAN model mentioned in subsection 2.1, this paper refers to the excellent performance of the residual network in the image classification task mentioned in the literature , the application of residual network in GAN model mentioned in literature. As well as the practice of utilizing residual networks to solve the gradient vanishing and gradient explosion problems in deep neural network training in literature.

In GAN-based face recognition, the residual block technique can be applied to improve the structure of the generator and the discriminator to improve the performance and stability of the model. By introducing residual connectivity, the gradient can be better propagated and the convergence of the network accelerated, while the problem of gradient vanishing during training is reduced. This makes the GAN-based face recognition model easier to optimize and train while improving the quality and realism of the generated face images [28].

In this paper, the residual block technique is utilized to construct the GAN model of this paper as per the experimental requirements described in Ref. In this paper GAN model generator encoder and decoder add residual blocks separately and the discriminator uses the FaceNet face recognition model based on the Inception Resnet V1 structure.

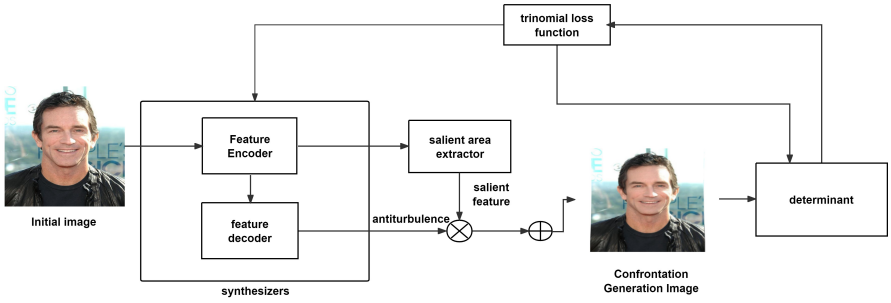

Fig. 3. General framework of face recognition method based on adversarial generative network

3.2 Generator

The generator adopts the design of the self-encoding structure, which consists of two parts: the feature encoder and the feature decoder. The encoder part adopts a lightweight structure, which includes a 9×9 convolutional layer, two 3×3 convolutional layers, three convolutional layers with Spectral Normalization of 64 channels, and six residual blocks, each of which includes a 3×3 convolutional layer and a BatchNormalization (BatchNorm) layer. three convolutional layers and a BatchNorm layer for each residual block. Implementing a lightweight encoder using a form of self-coding allows the synthesizer to have low computational complexity and a number of parameters while being able to efficiently extract features from the input data to produce high-quality images.

When the initial face image is input, the input image undergoes a convolution operation through a 9×9 convolutional layer, this convolutional layer extracts the basic features such as the edges and texture of the input face, and then the

feature map undergoes two 3×3 convolutional layers, each of which reduces the size of the feature map while increasing the depth of the features. After passing through two convolutional layers, the feature map is normalized by three spectral normalization layers. Finally, the feature map passes through six residual blocks. Higher-level information features such as contours, details, and structures of the face are learned and the encoder structure is shown in Fig 4.

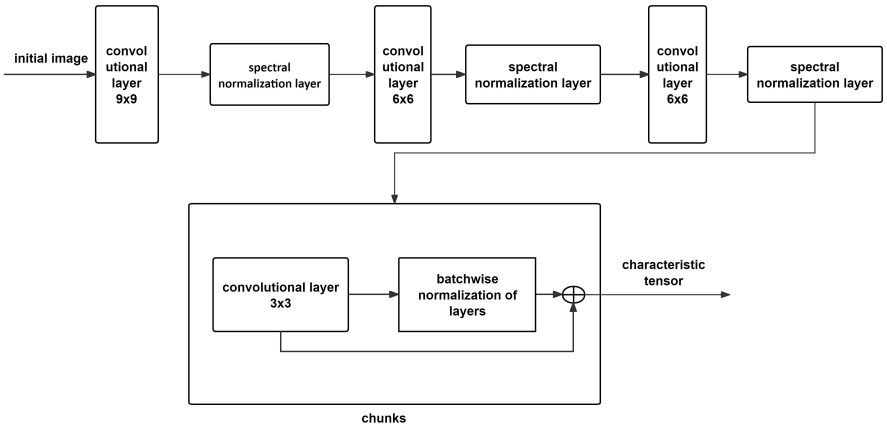

Fig. 4. Encoder Structure

The decoder part adopts the structure of the transpose convolutional layer and batch normalization layer, which is used to decode the feature mapping extracted by the encoder part into the final image output. It consists of two 3×3 transposed convolutional layers, one 3×3 ordinary convolutional layer, and three batch normalization layers, and such a structural design ensures that the generated images have better quality and realism. The network structure is shown in Fig 5.

After the encoder converts the input face images into low-dimensional feature representations, the decoder's task is to reconstruct the antagonistic perturbations from these feature representations so that the perturbations can be added to the original image to generate antagonistic samples. The decoder employs a transposed convolutional layer to incrementally increase the feature map and eventually generate an antagonistic perturbation of the same size as the original image. The procedure is as follows:

(1) 3×3 transposed convolutional layer: this layer doubles the size of the feature map so that a higher resolution image can be recovered from the low dimensional features obtained from the encoding.

(2) 3×3 transposed convolutional layer: this layer again doubles the size of the feature map, which is closer to the size of the original image.

(3) 3×3 convolutional layer: the feature map is optimized through convolutional operations to extract and adjust features, making the final antiperturbation more suitable for combining with the original image.

(4) Batch normalization layer: A batch normalization layer is used in the decoder for normalization.

Through the above decoding process, the decoder gradually generates the antiperturbation with the same size as the original image, thus realizing the generation of perturbation on the input face image.

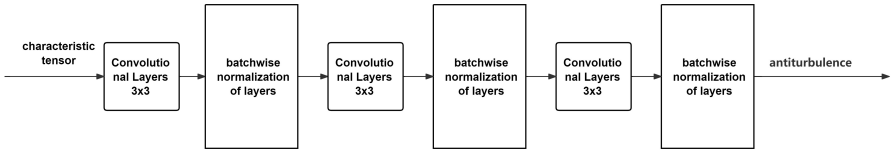

Fig. 5. Decoder Structure

3.3 Significant region extractor

The structure of the salient region extractor also three convolutional layers and three batch normalization layers, the difference is that the dimension of the output of the last one convolutional layer is changed because the output of the salient mapper is usually of the same dimensions as the original image, while the output of the decoder in the generator is of the same dimensions as the input image. The salient region extractor receives the output of the encoder in the generator as input. The main process steps are:

(1) Input one feature map containing the feature information extracted after encoding the input image. The size of the feature map is increased by a factor of one through one 3×3 transposed convolutional layer.

(2) The size of the feature map is increased by using 3×3 transposed convolutional layer again.

(3) one regular 3×3 convolutional layer is used. It keeps the size of the feature map the same, but reduces the dimensionality by decreasing the number of channels of the feature map to one, while using the BatchNorm layer for normalization.

(4) A significant map with values between zero and one is obtained through processing. In this saliency map, larger values represent higher importance in the input image, while smaller values represent lower importance. This saliency map indicates the saliency of different regions in the input image and provides useful information for further image analysis or processing. The structure of the salient region raiser is shown in Fig 6.

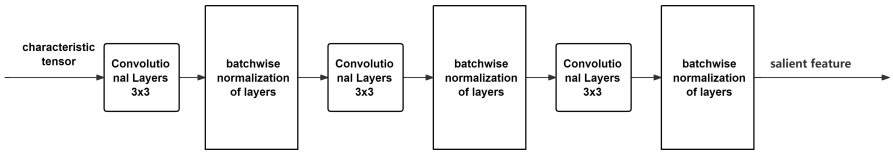

Fig. 6. Structure of the salient region extractor

3.4 Discriminator

The discriminator uses the FaceNet model based on the Inception Resnet V1 architecture, which combines the advantages of the Inception module and the Residual Connection. The Inception module processes the input data in parallel with multiple convolutional kernels of different sizes to capture features at different scales, employing a multi-scale analysis approach similar to the opcode vectorization strategy used in smart contract security analysis [29]. The Residual Connection, on the other hand, helps to solve the problem of gradient vanishing and representation bottleneck in deep networks, allowing the network to stack deeper without performance degradation [30]. When this structure is applied to face recognition, FaceNet is able to extract richer and more robust face features. These features are more robust to changes in illumination, expression, and posture, thus improving the accuracy of face recognition.

The module works as follows:

(1) Input the confrontation sample synthesized by the synthesizer.

(2) The sample image is passed through a series of convolution, pooling, and residual modules for feature extraction and transformation of the image. The model outputs embedded features.

(3) Compare the feature vectors of the target face with the template face to determine the identity of the face object.

The FaceNet model works by converting face images into feature vectors and mapping these feature vectors into a multidimensional space. In this multidimensional space, the feature vectors corresponding to the face images of the same person are close together, while the feature vectors corresponding to the face images of different people are farther apart. By calculating the distance between these feature vectors in Euclidean space, the similarity between faces can be evaluated, with stability verification principles adapted from the pattern consistency detection in [29]. If the distance between the feature vectors of two face images is less than a certain threshold, they are considered to be from the same person; otherwise, they are considered to be from different people. This method can effectively distinguish faces and achieve face recognition. The Euclidean spatial distance formula is shown below, where d represents the distance, and x_{1m},x_{2m} represent the feature vectors of the two images.

$$d = \sqrt{\sum_{m=1}^{n} (x_{1m} - x_{2m})^2}$$

4 Experiment

4.1 Experimental platform

The experiments in this paper are realized by renting the GPU cloud-sharing platform machine of Moment Pool Cloud, and the specific experimental environment is as follows. Processor: Intel Xeon Platinum 8260C Graphics. card: NVIDIA GeForce RTX 3090. Memory: 86G . Video Memory: 24G. Operating. system: Ubantu 20.04. Software platform: Anaconda-Python 3.10. Deep Learning Framework: PyTorch 2.1.1. Main dependent libraries: CUDA 11.8 cuDNN 8

4.2 Data presentation

The AR Face Database (also known as AR Face Database) contains image data of 50 males and 50 females, 26 images per person, about more than 5,000 images. The face part of these images has different expression variations, such as smiling or not smiling, eyes open or not open, and wearing glasses or not. It is used as the initial training set of the model in this experiment.

The LFW (Labeled Faces in the Wild) dataset contains a set of face images from the Internet with different poses, lighting conditions, expressions, and ages. There are 13,233 face images in total. It is used for validation experiments in this experiment. Yale Face Database: contains 165 face images from 15 different people, each containing 11 different expressions and lighting conditions. It is used for comparison experiments in this experiment.

CelebA (Celebrities Attributes) is a large-scale face attribute dataset containing about 200,000 celebrity images from the Internet. It is used for comparison experiments in this experiment.

CALFW (Caltech Faces Dataset) is a dataset for face verification tasks. It consists of about 3,000 face images from about 500 identities. It is used for comparison experiments in this experiment.

4.3 Data processing

This paper's uses AR face library as the training set. Each image is aligned and cropped to 128×128 using the face detector MTCNN. Fig 7 shows some of the data after MTCNN alignment and cropping. For other datasets used to perform validation and comparison experiments again each image is aligned and cropped to 128×128 using face detector MTCNN.

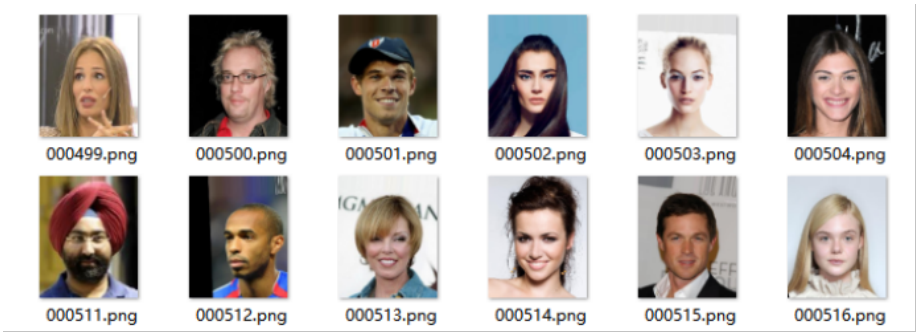

Fig. 7. Sample dataset after processing

The AR face library dataset for GAN model training and the sample images generated by the generator are whitened before the training data enters the discriminator. The data is linearly transformed to have zero mean and unit variance. This preprocessing method can help improve the comparability of the data and the performance of the model. Fig 8 illustrates the results of the image data whitening process.

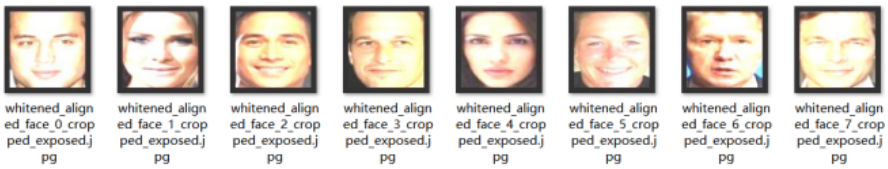

Fig. 8. Whitening Processing Data

4.4 Model Training

4.4.1 *Experimental parameters.*

The size of the input image in this experiment is 121×121 with pixel values between 0 and 255. The Learning Rate is set to 0.0002 and the Batch Size is set to 16. The Batch Size is set to 16 and the Epochs are set to 100.

The following loss functions are used in the generator and salient region extraction modules respectively:

(1) Mean Squared Error (MSE) loss function: used to measure the difference between the generated image and the target image. The difference between the generated image and the target image is used as part of the target attack loss, which is used to guide the generator to produce adversarial samples that are closer to the target image.

(2) Frobenius Loss: Used to constrain the magnitude of the generated adversarial perturbation to avoid the perturbation being too large. the Frobenius Loss is multiplied by a small coefficient added to the total loss as a regularization term to help control the magnitude of the perturbation.

For the judgmental FaceNet model a ternary loss function is used. An introduction to this loss function has been given in section 2.

4.4.2 *Experimental method and procedure.*

The experimental steps in this paper are:

(1) The initial image is fed into the synthesizer, and the high-level features in the initial image are decoded by the encoder in the synthesizer, and decoded by the decoder, and the antiperturbation factors are finally synthesized.

(2) Secondly, the significant region extraction module decodes the perturbation input from the decoder in the synthesizer and generates a feature map reflecting the importance of each part of the initial image, and combines the antagonistic perturbation factor with the feature map and combines it with the initial image to obtain the antagonistic sample.

(3) The antagonistic sample is fed into the discriminator for recognition judgment to improve the recognition ability of the discriminator. At the same time, the discriminator feeds the recognition results back to the generator part, guides the generator to generate the sample image through the mean square error, and at the same time, controls the size of the antagonistic perturbation added to the significant feature region by the significant region extraction module through the Frobenius Loss. Through the dynamic game process between the generator and the discriminator, the recognition ability of the discriminator is improved.

(4) Validation The FaceNet face recognition model trained by this paper's method is compared with the FaceNet model with the same structure obtained by training on large-scale datasets VGGface2 and CASIA-Webface on the LFW dataset in the comparison experiments, respectively, to judge the effectiveness and feasibility of this paper's experimental method, as well as the proposed viewpoints of this paper, through the introduction of the accuracy rate.

Fig 9 —11 demonstrates the process of synthesizing adversarial samples by the synthesizer. Where Fig 9 shows the adversarial perturbation, Fig 10 shows the salient feature extraction, and Fig 11 shows the generated adversarial sample. The salient map reflects the importance of each part of the face image, the higher the importance of the region in the face image corresponds to a larger value of the salient map, and the lower the importance of the region corresponds to a smaller value of the salient map. From the figure, it can be seen that the saliency map generated by this method can well determine different perturbation regions according to different inputs, and generate different weights for the antiperturbation according to the relative importance of each part of the input image, thus improving the image quality of the antiperturbation samples. With the input of about 5000 face images from the initial AR face library dataset, after the random feature perturbations are added by the GAN model generator and the salient region extraction module in this paper, finally about 500000 (0.5M) sample images are generated for the training of the discriminator.

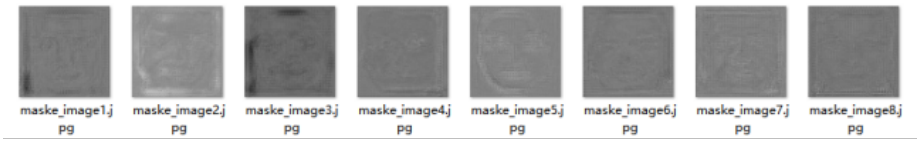

Fig. 9. Counteracting disturbance factors

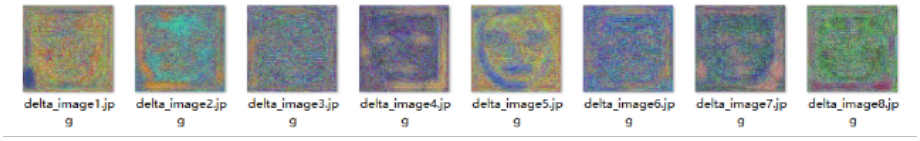

Fig. 10. Salient feature extraction

Fig. 11. Salient feature extraction

4.5 Model Implementation

The implementation is based on Python with the PyTorch framework. The code structure consists of four core modules:

Table 1. Model Code Composition

Code File	Function
`dataloader`	Define the methodology for all data processing in this paper
`FaceNet`	Constructing the discriminator for the GAN model in this paper
`generator`	Constructing the generators for the GAN models in this paper, **with security checks inspired by [**?**]'s interaction-aware validation**
`train_GAN`	Define generator and discriminator calls, and GAN model training methods

Where in generator, the salient feature extraction module mentioned in this paper is implemented by defining a class named SAE, in which two decoders are included: a perturbation decoder and a mask decoder. The perturbation decoder is used to learn the transformations that perturb the input image

```
self.perturb_decoder_1 = nn.Sequential
```

the mask decoder is used to learn the generation of a mask for the reconstruction of the image

```
self.mask_decoder_3 = nn.Sequential
```

The train-GAN file implements the training process with security monitoring adapted from [31]'s vulnerability detection framework, including: (1) Loading the data with integrity checks (2) Model parameter validation (3) Training process recording with anomaly detection

4.6 Experimental results and analysis

4.6.1 GAN model training results.

Fig 12 shows the change process of the generator loss function and discriminator loss function during the training of the GAN model in this paper.

The generator loss function first rises and then falls, at the beginning of training, the generator may generate some low-quality samples, which are more different from the real samples, resulting in the discriminator being more likely to recognize them as false samples, so the generator loss function may rise. As training proceeds, the generator gradually learns a better generation strategy that produces generated samples that are more realistic and closer to the real samples. This makes it more challenging for the discriminator to distinguish between the generated samples and the real samples, as the differences between them gradually become blurred. As a result, the loss function of the generator decreases gradually.

The discriminator loss function decreases as a 1/x function. In the initial stage of training, the discriminator has a more limited ability to distinguish between real and generated samples, and therefore the loss function is higher. As training proceeds, the discriminator gradually learns more accurate feature representations, which improves its ability to distinguish between real samples and generated samples, so the loss function gradually decreases. This process can be understood as the discriminator continuously improving its ability to better distinguish between different types of samples. When the performance of the discriminator is good enough, its loss function may stabilize and no longer change significantly.

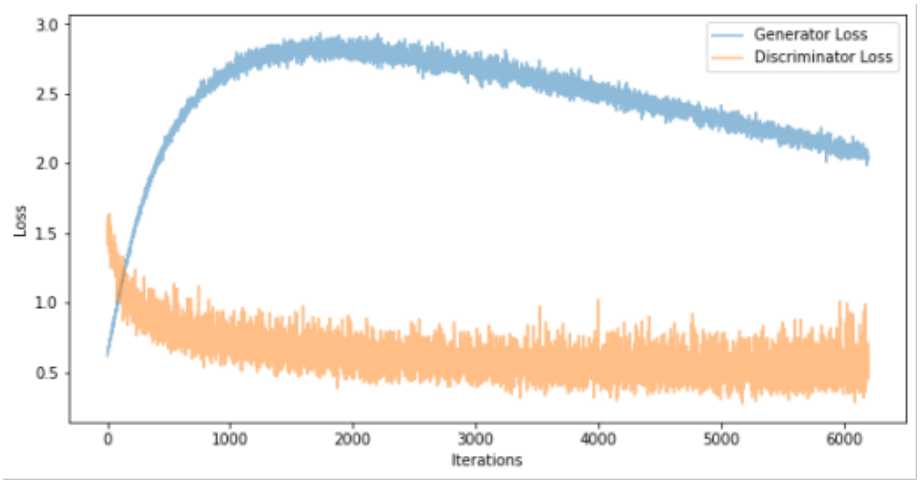

Fig. 12. Loss function change during model training process

4.6.2 Experimental results.

To verify the real face recognition effect of the discriminator in section 4.4.1. The LFW dataset is utilized for verification. Choose the face recognition accuracy on the LFW dataset as the accuracy of face recognition as the validation index.

The validation is performed using the LFW dataset, and the number of model iterations is set to 100. After validation, the recognition accuracy of the FaceNet face model trained using the method in this paper is shown in Fig 13. It can be seen that the recognition accuracy is close to 100when the number of iterations is 60. This result confirms the feasibility of this paper to utilize the GAN model features for small sample data enhancement and to train the face recognition model using the dynamic game features of the GAN model.

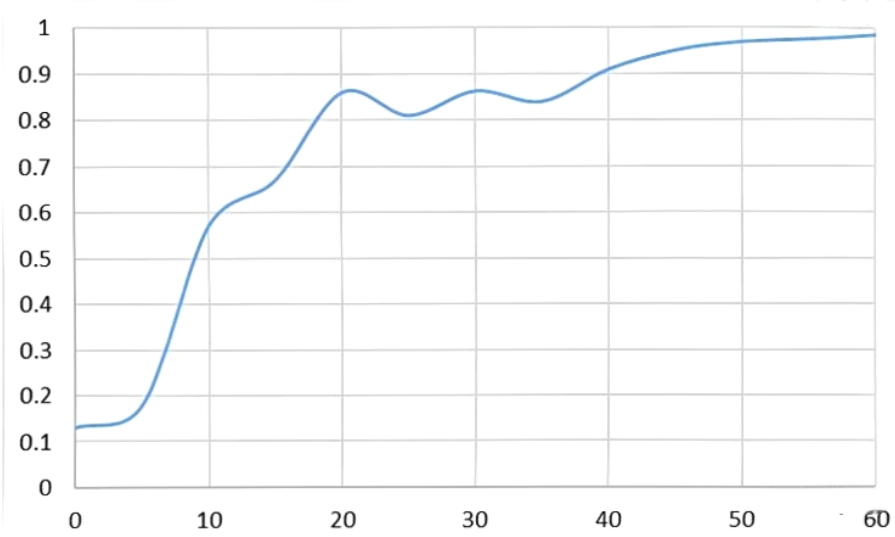

Fig. 13. Model Face Recognition Accuracy

4.6.3 Comparison Test.

To further validate the performance of the face recognition model trained under small sample data enhancement using GAN model features in this paper. The same structure FaceNet model trained on large datasets VGGFace, CASIA-Webface, and the FaceNet model trained by this paper's method are put on the CelebA dataset and CALFW dataset, YaleFace Database dataset for verification. In addition to the face recognition accuracy as the comparison benchmark, the number of faces required to recognize the same person is also used as the reference index for this comparison experiment. The CelebA dataset and the CALFW dataset are used as the datasets for face recognition accuracy, and the Yale Face Database dataset is used as the dataset for the number of images required to recognize the same person. The results of the comparison experiment are shown in Tab 2 and Tab 3.

Meanwhile, Tab 4 shows the FaceNet model trained based on the method of this paper, FaceNet model trained based on the VGGFace dataset, and FaceNet model pre-trained based on CASIA-Webface. the number of dataset images used for the training of the three models.

4.6.4 Analysis of Experimental Results.

Table 2. Face Recognition Accuracy Comparison

Model	Test Dataset	Accuracy (%)
Our Method (FaceNet)	CelebA	97.82
	CALFW	98.93
VGGFace-based	CelebA	99.92
	CALFW	99.94
CASIA-Webface	CelebA	97.52
	CALFW	98.71

Table 3. Minimum Image Requirements for Recognition

Model	Test Dataset	Images Required
FaceNet (Our Method)	YaleFace	13
FaceNet (VGGFace)	YaleFace	10
FaceNet (CASIA)	YaleFace	32

Table 4. Training Dataset Specifications

Model	Training Dataset	Image Count
FaceNet (Our Method)	AR Face Library + Generated Samples	0.5M+
FaceNet (VGGFace)	VGGFace Dataset	2.6M
FaceNet (CASIA)	CASIA-Webface Dataset	0.6M

The validation experiments in section 4.4.2 show that this paper utilizes the knowledge of residual networks to construct the GAN model with good results, which effectively avoids the problems of gradient disappearance and explosion encountered by the ordinary GAN model in the training process. We successfully utilize the dynamic game characteristics of the GAN model, train FaceNet as the discriminator of the GAN model in this paper, and finally test it in the LFW dataset to achieve good recognition results.

Meanwhile, the comparison experiments in section 4.4.2 show that this paper also achieves good results using the GAN model for small-sample data enhancement, and in the final experimental comparison of the face recognition rate and the images needed to recognize the same person with the same structural model trained based on a large dataset, the FaceNet face recognition trained using the method adopted in this paper also has a good performance.

5 Future work

The next step will be to explore the possibilities of this paper's method for cross-age-based face recognition. This is of great significance for the identification of missing children, matching the childhood photos of the missing children with their adult photos, even if the children's appearance has changed greatly after they have grown up, they can still be found by technical means. At the same time, dynamic face recognition is also the next direction of this paper's efforts, real-life face recognition is usually carried out in dynamic environments, and at the same time, the use of AI

face-switching technology to carries out fraudulent video cheating behavior is also an application of dynamic face recognition.

By utilizing the existing methods, we explore the possibility of realizing dynamic face recognition as well as contend with the prevention of AI face-swapping technology. At the same time, we design relevant systems to deploy the trained face recognition models to relevant hardware devices and apply them to real life. In addition to using the FaceNet model as a GAN model discriminator, we will explore the effect of other face recognition models as GAN model discriminators.

6 Conclusion

Through several months of related literature reading and experiments, we realized the construction of the GAN model using the residual network and solved the problems of gradient disappearance and explosion encountered in the training process of the GAN model. At the same time contend with the face recognition model under complex conditions, the training samples are difficult to obtain, and the lack of sufficient samples leads to poor model robustness and generalization ability. In this paper, the possibility of small sample data enhancement using the GAN model. The significant feature extraction module is added to the GAN model realized by the residual network to extract the significant features of the face and randomly add perturbations to it. Small sample data enhancement is successfully performed to expand the face recognition model training set. Using the dynamic game characteristics of the GAN model, the FaceNet model based on the Inception Resnet V1 structure is used as the discriminator of the GAN, which further improves the recognition ability of the face recognition model. Finally, the feasibility of the work done in this paper is verified through validation experiments and comparison experiments.

The work in this paper also has deficiencies and areas for improvement. For the data enhancement in a single way, it is only realized by randomly adding antagonistic perturbations in the region of significant features. For other more complex environments, such as angle change slight occlusion, and other complex conditions, this paper has not designed relevant experiments and solutions. Meanwhile, the recognition of dynamic faces is also a place not yet considered in this work, real-life face recognition is usually carried out in dynamic environments.

Acknowledgments

The author would like to express sincere gratitude to Prof. Tang Wan for her invaluable guidance and support during the foundational stages of this research.

References

[1] Zhengwei Wang, Qi She, and Tomas E Ward. Generative adversarial networks in computer vision: A survey and taxonomy. *ACM Computing Surveys (CSUR)*, 54(2):1–38, 2021.

[2] Yuanzheng Niu, Xiaoqi Li, Hongli Peng, and Wenkai Li. Unveiling wash trading in popular nft markets. In *Companion Proceedings of the ACM Web Conference 2024*, pages 730–733, 2024.

[3] Haoyu Wang and Lulu Guo. Research on face recognition based on deep learning. In *2021 3rd International Conference on Artificial Intelligence and Advanced Manufacture (AIAM)*, pages 540–546, 2021.

[4] Naval Kishore Mehta, Shyam Sunder Prasad, Sumeet Saurav, Ravi Saini, and Sanjay Singh. Three-dimensional densenet self-attention neural network for automatic detection of student's engagement. *Applied Intelligence*, 52(12):13803–13823, 2022.

[5] Dechao Kong, Xiaoqi Li, and Wenkai Li. Characterizing the solana nft ecosystem. In *Companion Proceedings of the ACM Web Conference 2024*, pages 766–769, 2024.

[6] Fan Liu, Delong Chen, Fei Wang, Zewen Li, and Feng Xu. Deep learning based single sample face recognition: a survey. *Artificial Intelligence Review*, 56(3):2723–2748, 2023.

[7] Hamed Alqahtani, Manolya Kavakli-Thorne, and Gulshan Kumar. Applications of generative adversarial networks (gans): An updated review. *Archives of Computational Methods in Engineering*, 28:525–552, 2021.

[8] Xiaoqi Li, Ting Chen, Xiapu Luo, and Chenxu Wang. Clue: towards discovering locked cryptocurrencies in ethereum. In *Proceedings of the 36th Annual ACM Symposium on Applied Computing*, pages 1584–1587, 2021.

[9] Zongwei Li, Wenkai Li, Xiaoqi Li, and Yuqing Zhang. Stateguard: Detecting state derailment defects in decentralized exchange smart contract. In *Companion Proceedings of the ACM Web Conference 2024*, pages 810–813, 2024.

[10] Yingjie Mao, Xiaoqi Li, Wenkai Li, Xin Wang, and Lei Xie. Scla: Automated smart contract summarization via llms and semantic augmentation. *arXiv preprint arXiv:2402.04863*, 2024.

[11] Ren-Hung Hwang, Jia-You Lin, Sun-Ying Hsieh, Hsuan-Yu Lin, and Chia-Liang Lin. Adversarial patch attacks on deep-learning-based face recognition systems using generative adversarial networks. *Sensors*, 23(2):853, 2023.

[12] Jiuyang Bu, Wenkai Li, Zongwei Li, Zeng Zhang, and Xiaoqi Li. Smartbugbert: Bert-enhanced vulnerability detection for smart contract bytecode. *arXiv preprint arXiv:2504.05002*, 2025.

[13] Lixiang Li, Xiaohui Mu, Siying Li, and Haipeng Peng. A review of face recognition technology. *IEEE access*, 8:139110–139120, 2020.

[14] Yishun Wang, Xiaoqi Li, Shipeng Ye, Lei Xie, and Ju Xing. Smart contracts in the real world: A statistical exploration of external data dependencies. *arXiv preprint arXiv:2406.13253*, 2024.

[15] Insaf Adjabi, Abdeldjalil Ouahabi, Amir Benzaoui, and Abdelmalik Taleb-Ahmed. Past, present, and future of face recognition: A review. *Electronics*, 9(8):1188, 2020.

[16] Zongwei Li, Xiaoqi Li, Wenkai Li, and Xin Wang. Scalm: Detecting bad practices in smart contracts through llms. *arXiv preprint arXiv:2502.04347*, 2025.

[17] Lingyu Yan, Jiarun Fu, Chunzhi Wang, Zhiwei Ye, Hongwei Chen, and Hefei Ling. Enhanced network optimized generative adversarial network for image enhancement. *Multimedia Tools and Applications*, 80:14363–14381, 2021.

[18] Chunming Wu and Ying Zhang. Mtcnn and facenet based access control system for face detection and recognition. *Automatic Control and Computer Sciences*, 55:102–112, 2021.

[19] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[20] Wenkai Li, Zhijie Liu, Xiaoqi Li, and Sen Nie. Detecting malicious accounts in web3 through transaction graph. In *Proceedings of the 39th IEEE/ACM International Conference on Automated Software Engineering*, pages 2482–2483, 2024.

[21] Zekai Liu, Xiaoqi Li, Hongli Peng, and Wenkai Li. Gastrace: Detecting sandwich attack malicious accounts in ethereum. In *2024 IEEE International Conference on Web Services (ICWS)*, pages 1409–1411, 2024.

[22] Amina Kammoun, Rim Slama, Hedi Tabia, Tarek Ouni, and Mohmed Abid. Generative adversarial networks for face generation: A survey. *ACM Computing Surveys*, 55(5):1–37, 2022.

[23] Jiuyang Bu, Wenkai Li, Zongwei Li, Zeng Zhang, and Xiaoqi Li. Enhancing smart contract vulnerability detection in dapps leveraging fine-tuned llm, 2025.

[24] Xiaoqi Li et al. Hybrid analysis of smart contracts and malicious behaviors in ethereum. 2021.

[25] Zekai Liu and Xiaoqi Li. Sok: Security analysis of blockchain-based cryptocurrency. *arXiv preprint arXiv:2503.22156*, 2025.

[26] Zongyong Cui, Mingrui Zhang, Zongjie Cao, and Changjie Cao. Image data augmentation for sar sensor via generative adversarial nets. *IEEE Access*, 7:42255–42268, 2019.

[27] Xiaoqi Li, L Yu, and XP Luo. On discovering vulnerabilities in android applications. In *Mobile Security and Privacy*, pages 155–166. Elsevier, 2017.

[28] Samik Banerjee and Sukhendu Das. Lr-gan for degraded face recognition. *Pattern Recognition Letters*, 116:246–253, 2018.

[29] Huanhuan Zou, Zongwei Li, and Xiaoqi Li. Malicious code detection in smart contracts via opcode vectorization. *arXiv preprint arXiv:2504.12720*, 2025.

[30] Guoxiang Tong, Fangning Hu, and Hongjun Liu. Dagan: A gan network for image denoising of medical images using deep learning of residual attention structures. *International Journal of Pattern Recognition and Artificial Intelligence*, 38(02), 2024.

[31] Wenkai Li, Xiaoqi Li, Zongwei Li, and Yuqing Zhang. Cobra: interaction-aware bytecode-level vulnerability detector for smart contracts. In *Proceedings of the 39th IEEE/ACM International Conference on Automated Software Engineering*, pages 1358–1369, 2024.