Retraction Notice

The Editor-in-Chief and the publisher have retracted this article, which was submitted as part of a guest-edited special section. An investigation uncovered evidence of systematic manipulation of the publication process, including compromised peer review. The Editor and publisher no longer have confidence in the results and conclusions of the article.

ZX did not agree with the retraction. MMK and JS either did not respond directly or could not be reached.

Method of generating face image based on text description of generating adversarial network

Zheng Xu,[a] **M. M. Kamruzzaman**,[b,*] **and Jinyao Shi**[a]
[a]Shanghai Polytechnic University, School of Computer and Information Engineering, Pudong District, Shanghai, China
[b]Jouf University, College of Computer and Information Science, Department of Computer Science, Sakaka, Al Jouf, Saudi Arabia

Abstract. Generative adversarial networks (GANs) can generate sentences that imitate real language features from discrete spaces, and the implicit expression space learned by training can continuously generate credible sentences. We solve the problem of face image restoration of different genders, different skin colors, and different hair colors through a method of generating face images, in which the face image generated by text description has a better generation effect. We propose a face image generation method (T-GAN) based on generating confrontational text. Due to the particulars of a face, a Word-long short-term memory model is used to extract the text features corresponding to the face. Then combined with the idea of generating a game against "game," a counter-text network is created, next a third type of input is added to the discriminator, and the real image with unmatched text is composed to strengthen the discriminator's training effect and force discrimination. The device determines whether the generated image conforms to the text description, so the discriminator can better learn the relationship between the text description and the image content. The experimental results show that, compared with other image generation methods, the face image generated by the proposed method can give high quality generated images with a better effect and less distortion, which can better maintain the original attributes of the image, has the use value, and increases the reduction degree by 3.84%. © *2022 SPIE and IS&T* [DOI: 10.1117/1.JEI.31.5.051411]

Keywords: generative adversarial network; text description; text feature extraction; face image generation.

Paper 220116SS received Feb. 6, 2022; accepted for publication Mar. 28, 2022; published online Apr. 15, 2022.

1 Introduction

With the emergence and rapid development of the generative adversarial network (GAN), many methods based on GAN have been applied to various fields and have solved various problems such as face occlusion, image conversion, and text description to image generation; GAN is more similar to a rapid development in the field of image applications.[1] GAN can generate a competition concept related to the introduction of game theory in the network, and continuously improve the ability to synthesize images and discriminate images by generating a model and discriminating between models.[2–4] A large number of experimental verifications and improvements compared with other generations have been recognized. The model has the ability to generate images that are realistic enough at the photo level, but the effect is not ideal for face images generated by text descriptions,[5–7] so we need a method that can generate face images efficiently and correctly through text description.[8–10]

With the multidisciplinary and wide-ranging application of artificial intelligence, more and more researchers have introduced artificial intelligence technology into the field of image generation.[11,12] The GAN has been shown to produce clearer and higher quality sample images, demonstrating its powerful performance and potential.[13–15] How to use artificial intelligence technology to quickly and accurately restore face images and restore the original feature

*Address all correspondence to M. M. Kamruzzaman, mmkamruzzaman@ju.edu.sa

attributes of face images to make the images more realistic is analyzed. Further, meeting the needs of security related fields such as criminal investigation and security and comprehensive stability has also been studied.[16,17]

2 Proposed Method

2.1 *Generative Adversarial Network*

2.1.1 *Confrontational thinking*

With the rise of artificial intelligence, confrontational ideas have become more and more popular among researchers.[18] The emergence of AlphaGo has caused the public to pay great attention to artificial intelligence. In the training process of AlphaGo, confrontational thinking in which two networks compete with each other and compete against the strategy network to obtain the current game state and predict the next action and corresponding strategies is used.[19] Some researchers have used two neural networks to compete in the process of training neural networks, so the number of hidden nodes in the network is not affected by statistics, and this is used as a regular factor. In fact, in the confrontation sample, there is also a confrontational idea. The so-called confrontation sample refers to the slight modification of the sample in training, which will lead to a wrong classification result that is very different from the real data sample and cannot be recognized by the human eye.[20] Based on confrontational thinking, researchers have achieved many achievements in various fields, and the confrontation of ideas has opened up a unique path for the development of artificial intelligence.

Unlike the traditional single neural network structure, GANs apply two seemingly independent neural networks: a generative network G and a discriminative network D, and the two networks are "linked" as a whole by an optimization function.[21,22] For the generation network G, the output results need to be optimized by the discriminant network D; for the discriminant network D, the output of the generation network G needs to be combined with the real data to complete its own training as a "supervised" input. Throughout the process, the goal of the generative network is to generate results that are closer to the real data, which is reflected in the output of the discriminative network and applied to the optimization process of the generative network G.[23]

2.1.2 *Generative adversarial network definition*

For nonlinear mapping functions (such as multilayer perceptron or convolutional neural networks), the entire system model is trained by backpropagation algorithms. In the GAN network training process, when the generated network is fixed, the discriminating network finds its current minimum value, and when the discriminating network is fixed, the generating network searches for its current minimum value but discriminates the network and generates the network minimum value.

GANs were originally designed to generate continuous data, but in natural language processing, we want to generate discrete sequences. Because the generator needs to use the gradient obtained from the discriminator for training and both G and D need to be completely differentiable, there will be problems when there are discrete variables, and only Back Propagation (BP) cannot provide G with the gradient for training. In GAN, the data generated by it is made more "realistic" by making small changes to the parameters of G.

The left graph shown in Fig. 1 represents the discriminator D. When we input the real data x, the expected output probability value is very close to 1; the lower graph shows the generator G, when we input the random noise z that obeys the simple distribution (such as uniform distribution and Gaussian distribution). At this time, the output image size of the output is close to the real image, and the result generated by the generated model is input into the discriminant model, and the discriminator will output a low probability (data from the generation model) as much as possible. The model G spoofing discriminant model D is generated such that the discriminant model D outputs a high probability, thereby forming a competitive confrontation relationship.

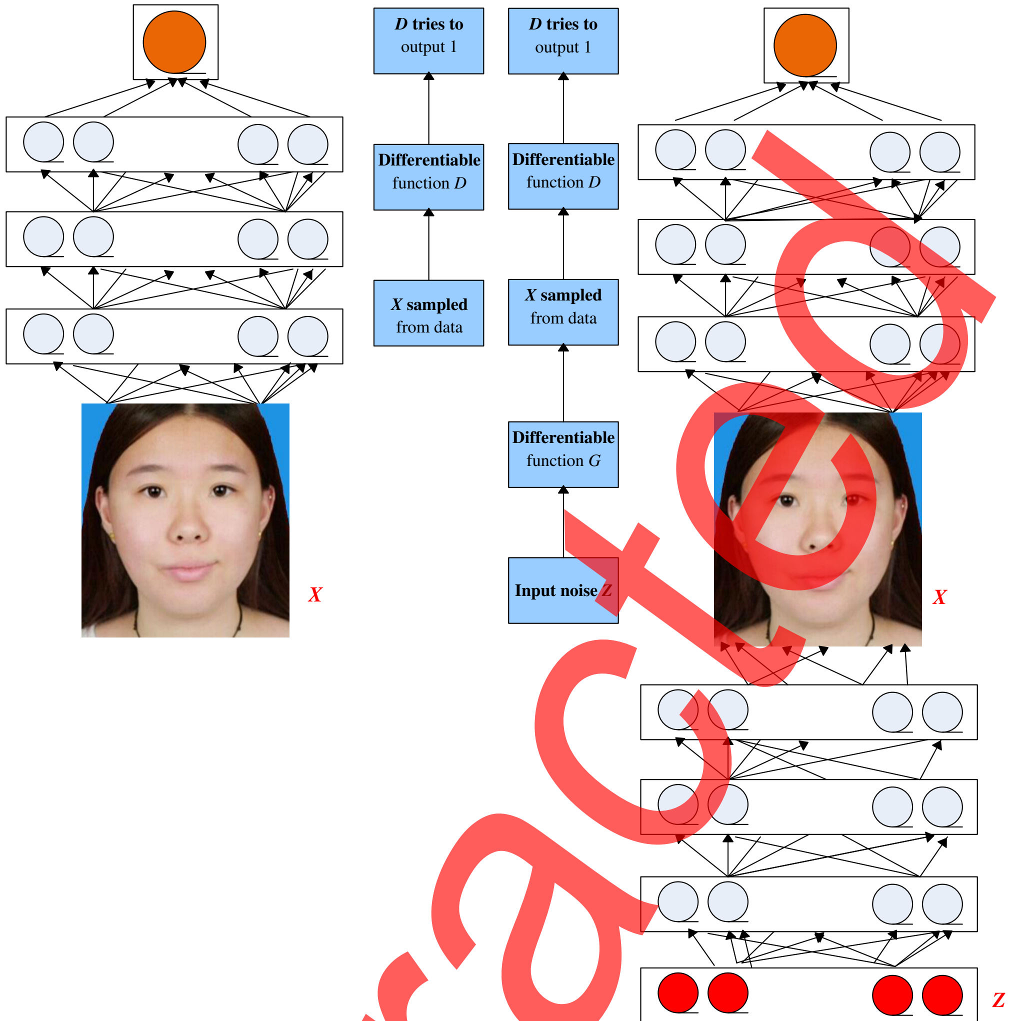

Fig. 1 Generative adversarial network model.

The specific training can be divided into two steps: step A: the real sample r is sampled from the corpus, and the generated sample x is sampled from the generator network; then the supervised discriminator network is trained, and the parameter of the discriminator network is r, the probability of the real sample, and reduce the probability of x being the real sample. In the above process, the parameters of the generator network are fixed. Step B: the generated sample x is sampled from the generator network, and the generator network parameters are updated to increase the probability that the newly generated sample x' is output as a real sample in the discriminator. In the above process, the parameters of the discriminator network are fixed.

2.2 *Text Feature Extraction*

The text generation method based on sequence model refers to modeling along the one-dimensional dimension of the text tag sequence, constructing a language model, and then generating text. Common forms of sequence models include recurrent neural networks, sequence-to-sequence models, attention-sequence models, etc. Recurrent neural network belongs to the category of deep learning, and it is also a more common type of sequence model. It has achieved good results in many tasks of natural language processing. A neural network is a network model that performs output inference in a sequence dimension with shared parameters.

Because of the language expression, not every word in a sentence can describe the text well. Therefore, extracting useful text information is very critical. With the rapid development of deep learning, deep learning models such as convolutional neural network (CNN) and long short-term memory (LSTM) have also been applied in the field of text and have achieved good results.

Feature extraction is a process of further screening for the remaining text words after preprocessing. The principle followed in the screening is to retain the words with the richest textual information as much as possible to minimize the interference of noise terms and improve the classification accuracy. Therefore, feature extraction selects words that contain a large amount of semantic information and are most valuable for reflecting the central idea of the text from a large collection of text-specific testimony and defines such words as the characteristic words for judging the text category: efficiency of feature extraction.

Feature extraction generally includes two processes: first, an evaluation function is created according to the attributes of the feature words, and the attribute evaluation value of each word in the feature set is calculated according to this function. The values are sorted in descending order, the feature words with evaluation values that are less than the reference value are deleted, and the words saved after filtering are used as the feature words of the text.

Most of the traditional text generation models use the maximum likelihood estimation (MLE) method for sequence optimization: given a text sequence $s_t = [x_0, x_1, \ldots, x_{t-1}]$, the next element of the sequence comes from sampling, $x_t \sim \pi_\theta(x|s_t)$ and then the MLE-based standard pattern text generation model uses the likelihood method for estimating, which is given as

$$\max_\theta \sum_x \sum_t \log \pi_\theta(x_t|s_t). \tag{1}$$

The method of traditionally generating text using LSTM includes the following steps, as shown in Fig. 2:

1. The hidden state of the LSTM (hidden state) h_0 and the memory unit c_0 (cell state) are initialized.
2. The results predicted by the respective LSTM calculation units are used as inputs.
3. The hidden state and the memory unit are combined, and the current step output is calculated as

$$p = \text{softmax}(h). \tag{2}$$

The traditional LSTM method is used for calculation. This approach faces the challenge of exposure bias, and an element in the sequence generated by the model needs to rely on information from all previous sequence elements, but this information is often overlooked during training, thus creating a situation in which the generated text deviates from the true sample distribution and is too close to the training sample.

The LSTM unit has a memory unit to hold historical information. The update, maintenance, and utilization of historical information are controlled by three gates, namely, the input gate, the output gate, and the forgetting gate. To explain the working principle of LSTM more conveniently, the output unit value, memory unit value, and input data value of LSTM are set to h, c, and x, respectively.

To obtain a visually discernible vector representation of the text description, we use a deep neural encoder to generate the inner product of features, learn a function corresponding to the

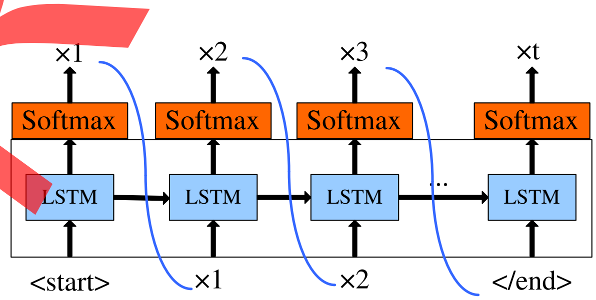

Fig. 2 LSTM text generation method.

image, and then classify the text generated by the learned corresponding function f_t to optimize training and its structural loss, which is given by

$$\frac{1}{N}\sum_{n=1}^{N}\Delta(y_n, f_v(v_n)) + \Delta(y_n, f_t(t_n)), \tag{3}$$

where $\{(v_n, t_n, y_n): n = 1, \cdots, N\}$ is the training data set, Δ is 0 to 1 loss, v_n is the image, t_n is the corresponding text description, y_n is the class label, and f_v and f_t are the image and text classifiers, respectively. The parameters are calculated as follows:

$$f_v(v) = \arg\max_{y\in\gamma} E_{t-\tau(y)}[F(v,t)], \tag{4}$$

$$f_t(t) = \arg\max_{y\in\gamma} E_{v-v(y)}[F(v,t)], \tag{5}$$

$$F(v,t) = \varphi(v)^T\phi(t), \tag{6}$$

where ϕ is the image encoder, $\tau(y)$ is the text description data set, and $v(y)$ is the image data set. $\phi(v)$ is an image feature, and $\varphi(t)$ is a text feature. f_v and f_t share this function, extract corresponding features for image and text descriptions, respectively, and then combine to give each image a score that satisfies the text description. If the classifier is correctly classified, the score of the image and the matching text should be significantly higher than the scores of other matches that cannot be matched. Due to the special and complex nature of the face, this paper uses CNN to extract features for each image. Word-LSTM extracts the features of the text description as shown in Fig. 3. Through the scores, we can see the accuracy between the image and the matched text features.

The difference between before and after the improvement of LSTM is not big, but it is much better than tan h. However, for different data sets, the performance of the improved model and LSTM is slightly different. The multiple gate mechanism of LSTM makes it advantageous when dealing with large data sets, that is, LSTM requires more data to adjust parameters, while the improved model shows its performance on small data sets and trains faster than LSTM.

For “game” idea generation and text network design, this paper adds a third discriminative input. It consists of real images and does not match the text, which strengthens the discriminative training effect to some extent. The mandatory discriminant is to see whether the generated image conforms to the text description. It needs to study the relationship between the image generated by the discriminator and the input image, and at the same time, it also needs to strengthen the learning to generate image and text description. Based on the text description generated in the face image process, it is first necessary to quantify a text description, and then extract text

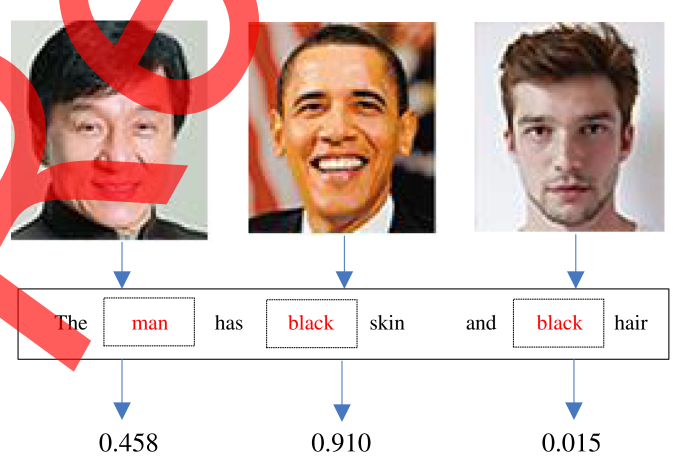

Fig. 3 Word-LSTM feature extraction.

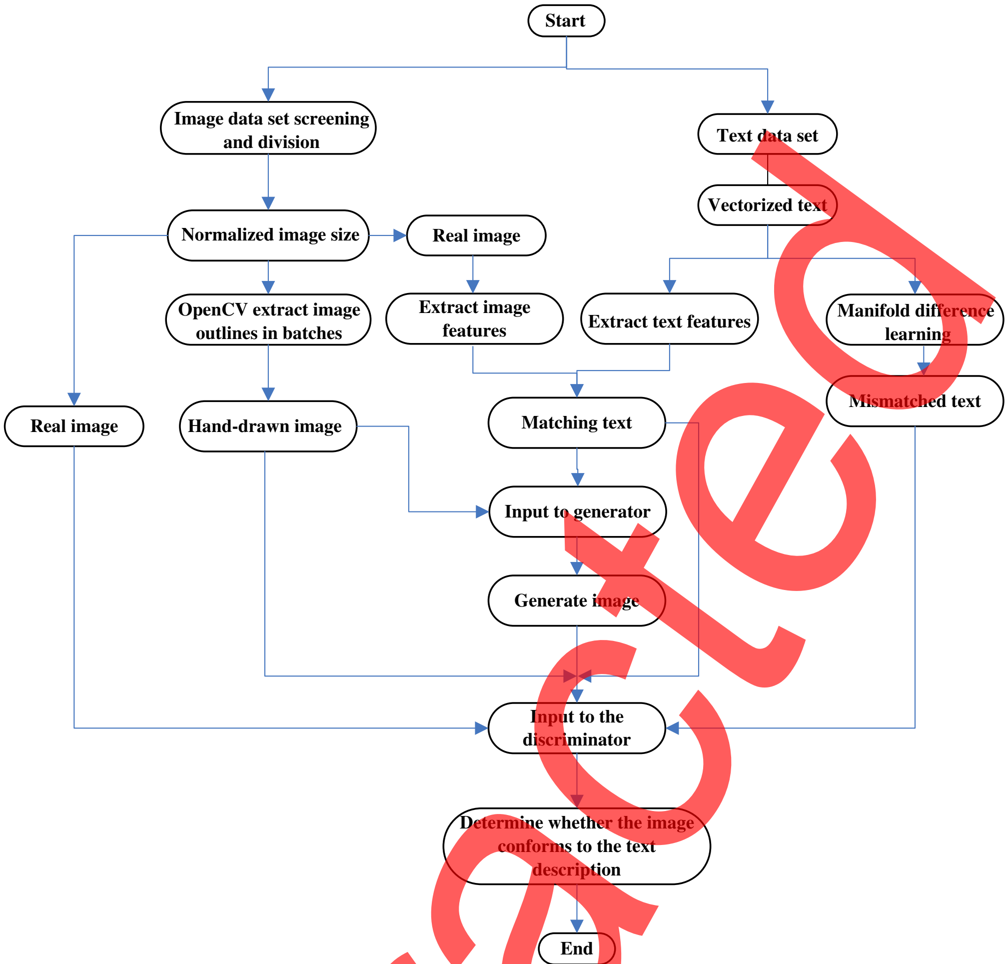

Fig. 4 Flowchart of the T-GAN method.

features. In order to obtain better text features, the Word-LSTM model is used to extract features from each image. The matching text of the image is then obtained by combining with the image features extracted from the preprocessed image and put into the Text Generative Adversarial Network (T-GAN) together with the hand-drawn effect image of the image, as shown in Fig. 4. T-GAN trains a Deep Convolutional Generative Adversarial Network based on text features. Generator G and discriminator D both perform feedforward inference based on text features and combine the convolution network in supervised learning with GAN in unsupervised learning.

For face image generation, we introduce a condition variable, which is a text description. The text description is used as a condition variable, and the original GAN structure is compared. For image generation, the input of G is image x; it receives not only image x but also the text $\varphi(t)$, and together image x and the text $\varphi(t)$ generate an image, which is given by

$$\hat{x} := G(x, \varphi(t), z). \tag{7}$$

The discriminator D is originally the received image $G(z)$, and the output score $D(G(z))$, where not only the generated image but also the input image x and the text $\varphi(t)$ are received, and it is necessary to ensure that the image generated by G is not only the input. The images are paired and matched to the text description, and finally the output is scored, which is given by

$$D(x, x, \hat{\varphi}(t)). \tag{8}$$

The T-GAN method introduces text descriptions as condition variables to enhance discriminator training, which is given by

$$V_{T\text{-GAN}}(G, D) = \genfrac{}{}{0pt}{}{E_{x,y-P\mathrm{data}(x,y)}[\log\ D(x,y,\varphi(t))]}{+E_{x-P\mathrm{data}(x),z-Pz(Z)[\log(1-D(x,x,\hat{\varphi}(t)))]}}, \tag{9}$$

$$V_{T\text{-GAN}} = \arg\ \min_G \max_D V_{T\text{-GAN}}(G, D). \tag{10}$$

To make the image generated by the generator clearer, the degree of reduction is higher, and it is closer to the real image. In the Eq. (10), the $L1$ norm loss function is added, and the final objective function is given as

$$V = \arg\ \min_G \max_D V_{T\text{-GAN}}(G, D) + \lambda V_{L1}(G). \tag{11}$$

The text generation model based on the GAN framework and reinforcement learning mainly has two defects: one is the slow convergence speed and the other is that the quality of the generated text is not high enough. The main reason for these two problems is the insufficient feedback information obtained by the generator network from the discriminator network in the GAN. When the target length of the generated text is long, the scalar information obtained by the generator network is not enough to guide it to update the parameters in the correct direction. In this paper, the sparse feedback obtained by the generator network is obtained through insufficient sampling, which leads to a further reduction in the effective information of the feedback signal, so that the update of the generator requires more trial and error, and the generation of the quality of the text is not easily improved.

Text preprocessing is the process of preparing for the entire text classification, and the effect of preprocessing indirectly affects the execution of each link below. Text preprocessing first removes non-text factors in the text and then performs word segmentation and stop word removal operations on the processed text. This process eliminates all of the interfering words in the text that are useless for classification. Finally, the text is described as a reasonable model so that it can be read and processed by the computer. In general, preprocessing is performed to eliminate noisy data that negatively affects classification, thereby improving the efficiency of feature collection and effectively improving the accuracy of text classification while reducing memory space usage.

When training the Recurrent Neural Network (RNN) network model, solving the gradient of the loss function with respect to the model parameters is the core step. The Back Propagation Through Time (BPTT) algorithm is used in the training of the RNN model. The BPTT algorithm is a simple BP algorithm, that is, the "chain rule" is used to solve the parameter gradient, but the parameters are shared at each time-step in the solution. From a mathematical point of view, the BP algorithm is a single-variable derivation process, while the BPTT algorithm's derivation process is a composite function.

3 Experiments

3.1 *Experimental Environment*

The experiment in this paper is done on a Linux | Ubuntu 16.04 LTS CPU, clocked at 3.04 GHz with an Ubuntu 16.04 operating system, a GTX 1060 GPU, with a 64 G memory, and the Python 3.5.2 development language. The specific configuration is shown in Table 1.

3.2 *Experimental Data Set*

The experimental data in this paper uses the CelebA data set, which is a data set of celebrity face attributes. The CelebA face data set is a large data set of more than 200,000 celebrity face images that is widely used. After screening, 10,000 face images were selected for the experiment, which included 8000 images for training. Each image is labeled, and each data has 40 feature text attribute annotations to ensure image diversity, including if the face is smiling, wearing glasses, wearing a hat, having long hair, or having short hair. It can be divided into large-scale attitude

Table 1 Experimental environment configuration.

Project	Configuration
Operating system	Linux I Ubuntu 16.04 LTS
CPU	Intel Core i7-6800 K at 3.40 GHz
Graphics processor	GTX 1060
RAM	64 G
Hard disk	175.1 GB solid-state drive
Development language	Python 3.5.2

change and background change. Select multiple feature tags include gender, skin color, hair color, and glasses for training.

3.3 *Experimental Steps*

1. Image preprocessing is performed on the face image of the experiment first. The image size of the original data set is not uniform, and some image poses or occlusion are significant. This paper tries to select a better face posture, compare positive images, and reduce the influence of face posture. Images that do not look right or match well are removed, and only the information such as faces and hair is saved.
2. All data set face images have the same GAN architecture and are uniformly adjusted. The size of the adjustment refers to the size of the optimized model training image. The size of the last used training image is normalized to 256×256, which will be "extracted." The "text feature description" and the "real face map" are used as input to the network.
3. For the text description, the text encoder is pre-trained to speed up the training of other components to speed up the experiment. The 1024-dimensional text encoding generated by the text encoder is compressed to 128 dimensions, stitched together before the image feature mapping, and placed in the generation network and the discriminant network.
4. The performance of three main methods of T-GAN, CGAN, pix2pix, and CycleGAN, are compared. Through the comparison of skin color, hair color, hair length, restoration of face images, realism, and hue saturation, GAN has a higher degree of reduction and is more suitable for text description methods for generating face images.

4 Discussion

4.1 *Text Description Generate Image Analysis*

4.1.1 *Text feature analysis*

All images in the data set have the same GAN architecture, and the size of the training image is normalized to 256×256. Due to the particulars and complexity of the face, we must extract the very key words when extracting the text features to better generate a face image that matches the text description. Therefore, from the 10,000 face data sets with characteristic attributes, as shown in Fig. 5, 2000 face images of different genders, different skin colors, and different hair colors are selected for testing. No problems occurred during the test, and the tests were all successful.

4.1.2 *Face image hand-painted*

For the problem of generating face images, the existing methods all use the ability to generate realistic images against the network or its extended network. However, due to the particulars and complexity of the face, it is often impossible to achieve better results. Therefore, using a face image method based on generating a textual description, the method uses the text description as a condition variable and uses the text to describe the face accordingly, which indirectly reduces the

Fig. 5 Legend of the text attribute part of the CelebA face data set.

Fig. 6 Face image results based on textual descriptions GAN.

difficulty of generating the sample during the training process. The evaluation results are shown in Fig. 6.

It can be seen from the entered texts such as gender, skin color, hair color, etc., due to the different specific conditions of the face, the corresponding text descriptions are also different. The facial features are described by text, combined with the strong generation ability of generative confrontation. The forced discrimination is to see whether the generated image conforms to the text description. It needs to learn the relationship between the generated image and the text description, so that the generated result has a higher degree of restoration and is closer to the content of the real image and text description.

4.2 *Analysis of Various Methods for Generating Face Images*

4.2.1 *Comparison of face images generated by various methods*

To highlight the feasibility of the proposed method in generating face images, it is compared with other methods for generating face images, as shown in Fig. 7. CGAN easily leads to a color shift

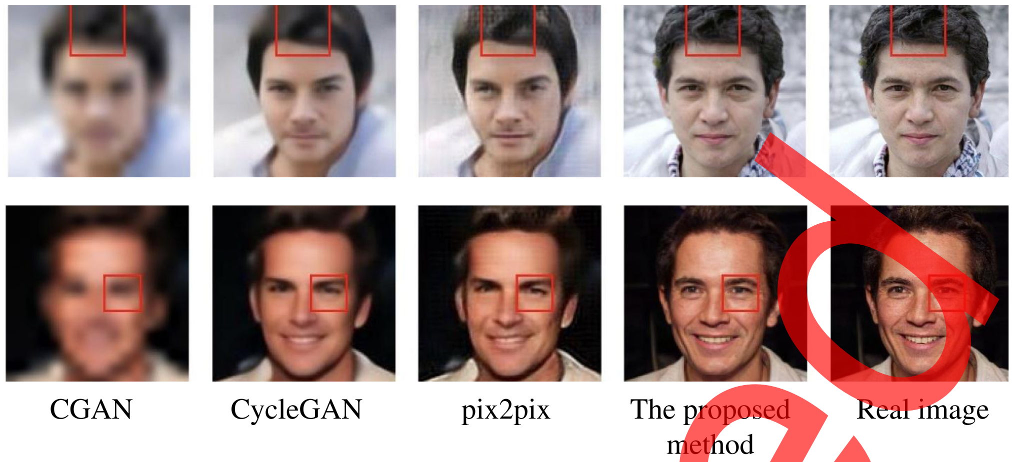

Fig. 7 Comparison of various methods for generating face images.

Table 2 Comparison of the average indicators of sample images by a variety of different generation methods.

Index	CGAN	CycleGAN	pix2pix	Proposed method
SSIM	0.911	0.8854	0.9555	0.9706
PSNR	20.3025	19.772	28.6652	32.5052

and produces more artifacts and ambiguities; CycleGAN has a more serious distortion in the process of generating face images, and the generated images are unrealistic. The face images generated by pix2pix are better. The skin color and hair color characteristics of the face are maintained, and a good image effect can be achieved on the face of the yellow skin and the white skin, but for the black skin, the image result still has a certain deviation, with the color being too saturated and the face color becoming white. As can be seen from Fig. 7, the proposed method has a higher degree of reduction in generating face images and is closer to the image generated by the real image.

To objectively prove that the method can better maintain image attributes after image generation, the structural similarity index measure (SSIM) and peak signal-to-noise ratio (PSNR) of the method and other image generation methods in the training samples are shown in Table 2 and Fig. 8. The SSIM is used to measure the similarity of the two graphs, and the degree of distortion of the image can be judged by detecting the structural information changes. The PSNR is the ratio of the maximum power of a signal to the corrupted noise power that affects its fidelity, and it is typically used to measure the quality of the processed image. It can be seen that, compared with other methods, the proposed method has higher SSIM and PSNR, indicating that the generated effect is better, the degree of reduction is higher, and the influence of noise on the image is smaller.

It can be seen from the table that the PSNRs of CGAN, CycleGAN, pix2pix, and the proposed method are 20.3025, 19.772, 28.6652, and 32.5052, respectively.

4.2.2 *Face generation results under the influence of external factors*

In real life, there are often face images that are obscured by external factors. To further verify the self-applicability of the proposed method, the experiment is performed by obscuring the face image with external factors, as shown in Fig. 9. It can be seen from the experiment that the proposed method can basically restore the original attributes of the face image when the face image is covered by the external factors, with a higher degree of restoration, greater realism, strong robustness, and self-applicability.

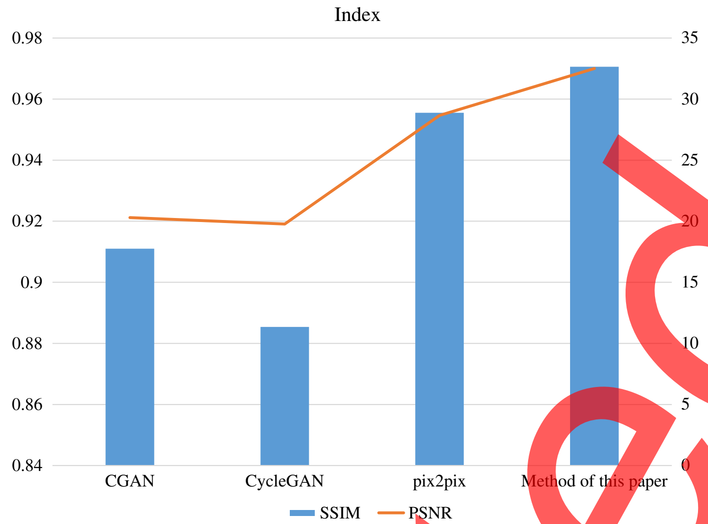

Fig. 8 Comparison of the average indicators of sample images by different generation methods.

Fig. 9 Face generation results under the influence of external factors.

5 Conclusions

This paper mainly studies the method of generating face images based on text description of the GAN. This paper studies and analyzes the traditional text generation method, designs a method suitable for this paper, and proposes a face image generation method based on the generation of anti-network text description for face image generation. The effectiveness of this method is proved by comparing it with the traditional text generation method on the CelebA face data set.

The work of this thesis is mainly based on the particulars of the face, extracting the features in the text description corresponding to the face combined with the confrontational idea, forcing the discriminant network to determine whether the generated image is consistent with the text description, and learning to generate the image and the input image and text description. The relationship between the two plays an optimization role. The proposed method can solve the problem of face image restoration of different genders, skin color, and hair color and can better maintain the original attributes of the image.

The generator network does not train efficiently after receiving the high-order feature vectors extracted by the discriminator. Whether the high-order features are connected to the input of the recurrent neural network at each time point or the high-order features are transformed and incorporated into the hidden layer, the generator model training cannot be converged. This paper believes that the reason for this phenomenon is that the feature extraction layer of the discriminator network is completely trained for the text classification task, and this feature extraction layer directly accesses the completely irrelevant text generation layer, which may lead to incongruous network training. Based on this analysis, this paper will introduce the feature transformation layer as a transition module between the feature extraction layer and the text generation layer.

The text generation model in this paper is still based on uncontrollable generation. How to introduce the conditional generation of text into the model and make it adapt to the parameter adjustment method of reinforcement learning still has theoretical and engineering difficulties. This paper overcomes a lot of practical problems and completes the controllable generation model of text.

Acknowledgments

This work was funded by the Deanship of Scientific Research at Jouf University under Grant No. DSR-2021-02-0342. There are no conflicts of interests in our paper. And all authors have seen the manuscript and have approved it for submission to your journal. We confirm that the content of the manuscript has not been published or submitted for publication elsewhere.

Code, Data, and Materials Availability

This article does not cover data research. No data were used to support this study.

References

1. Y. Zhou et al., "A novel method to generate the tooth surface model of face-milled generated spiral bevel gears," *Int. J. Adv. Manuf. Technol.* **102**(11), 1205–1214 (2019).
2. J. Martínez-Ferrero, S. Banerjee, and I. M. García-Sánchez, "Corporate social responsibility as a strategic shield against costs of earnings management practices," *J. Business Ethics* **133**(2), 305–324 (2016).
3. A. Batushansky, D. Toubiana, and A. Fait, "Correlation-based network generation, visualization, and analysis as a powerful tool in biological studies: a case study in cancer cell metabolism," *Biomed. Res. Int.* **2016**(12), 1–9 (2016).
4. V. N. Possani et al., "Graph-based transistor network generation method for supergate design," *IEEE Trans. Very Large Scale Integration Syst.* **24**(2), 692–705 (2016).
5. X. Han, "MR-based synthetic CT generation using a deep convolutional neural network method," *Med. Phys.* **44**(4), 1408–1419 (2017).
6. I. Vazaios, N. Vlachopoulos, and M. S. Diederichs, "Integration of lidar-based structural input and discrete fracture network generation for underground applications," *Geotech. Geol. Eng.* **35**(5), 2227–2251 (2017).
7. J. Yuan and A. M. Cheriyadat, "Image feature based GPS trace filtering for road network generation and road segmentation," *Mach. Vis. Appl.* **27**(1), 1–12 (2016).
8. A. Aparna and S. Ajish, "Cryptographic key generation based on contextual information: a review," *Int. J. Comput. Appl.* **134**(15), 1–5 (2016).
9. Z. Xu et al., "Fault recovery of distribution network containing distributed generation based on heuristic search algorithm and multi-population genetic algorithm," *Syst. Sci. Control Eng.* **6**(3), 100–108 (2018).
10. C. Kang et al., "Carbon emission flow from generation to demand: a network-based model," *IEEE Trans. Smart Grid* **6**(5), 2386–2394 (2015).
11. Y. Gao et al., "Multi-objective bi-level coordinated planning of distributed generation and distribution network frame based on multi-scenario technique considering timing characteristics," *IEEE Trans. Sustainable Energy* **8**, 1415–1429 (2017).

12. W. U. Lizhen, L. Jiang, and X. Hao, "Reactive power optimization of active distribution network based on optimal scenario generation algorithm," *Power Syst. Protect. Control* **45**(15), 152–159 (2017).
13. G. Muñoz-Delgado, J. Contreras, and J. M. Arroyo, "Multistage generation and network expansion planning in distribution systems considering uncertainty and reliability," *IEEE Trans. Power Syst.* **31**(5), 3715–3728 (2016).
14. D. Wang et al., "From IoT to 5G I-IoT: the next generation IoT-based intelligent algorithms and 5G technologies," *IEEE Commun. Mag.* **56**(10), 114–120 (2018).
15. C. Li, H. D. Chiang, and Z. Du, "Network-preserving sensitivity-based generation rescheduling for suppressing power system oscillations," *IEEE Trans. Power Syst.* **32**(5), 3824–3832 (2017).
16. N. G. Cobos, J. M. Arroyo, and A. Street, "Least-cost reserve offer deliverability in day-ahead generation scheduling under wind uncertainty and generation and network outages," *IEEE Trans. Smart Grid* **9**, 3430–3442 (2018).
17. J. Wu, X. Yin, and H. Xiao, "Seeing permeability from images: fast prediction with convolutional neural networks," *Sci. Bull.* **63**(18), 1215–1222 (2018).
18. G. Mai et al., "Face image reconstruction from deep templates," *IEEE Trans. Pattern Anal. Mach. Intell.* 1–5 (2017).
19. T. Zhou, "An image recognition model based on improved convolutional neural network," *J. Comput. Theor. Nanosci.* **13**(7), 4223–4229 (2016).
20. N. Cordier et al., "Extended modality propagation: image synthesis of pathological cases," *IEEE Trans. Med. Imaging* **35**(12), 2598–2608 (2016).
21. F. Zhang et al., "Human face sketch to RGB image with edge optimization and generative adversarial networks," *Intell. Autom. Soft Comput.* **26**(4), 1391–1401 (2020).
22. T. Nagasawa et al., "fNIRS-GANs: data augmentation using generative adversarial networks for classifying motor tasks from functional near-infrared spectroscopy," *J. Neural Eng.* **17**(1), 016068 (2020).
23. J. Dai et al., "Machinery health monitoring based on unsupervised feature learning via generative adversarial networks," *IEEE/ASME Trans. Mechatron.* **25**(5), 2252–2263 (2020).

M. M. Kamruzzaman received his BE and MS degrees in computer science and engineering and his PhD in information and communication technology from Southwest Jiaotong University, China. Currently, he is working as an associate professor at Jouf University, Saudi Arabia. He worked as a post-doctoral research fellow at Shenzhen University, China. He is a senior member of IEEE and serves as a member of the editorial board of several international journals. He also serves as a TPC and reviewer of a few international journals and conferences. His areas of interest include 5G, IoT, AI, ML, deep learning, image processing, remote sensing, GIS, cloud computing, and big data.

Biographies of the other authors are not available.