Journal of Combinatorial Mathematics
and Combinatorial Computing
www.combinatorialpress.com/jcmcc

A Text-Free Training Method for Generating Face Images Based on Chinese Text

Songlin Tong[1], Meiling Liu[1,✉], Jiyun Zhou[2]

[1] *School of Information and Computer Engineering, Northeast Forestry University, Harbin, Heilongjiang, 150080, China*

[2] *Lieber Institute, Johns Hopkins University, Baltimore, 999039, USA*

ABSTRACT

Generative adversarial network (GAN) technology has enabled the automatic synthesis of realistic face images from text. This paper proposes a model for generating face images from Chinese text by integrating a text mapping module with the StyleGAN generator. The text mapping module utilizes the CLIP model for pre-training Chinese text, employs a convolutional-inverse convolutional structure to enhance feature extraction, and incorporates a BiLSTM model to construct complete sentences as inputs for the StyleGAN generator. The generator interprets semantic features to generate face images. Validation on Face2Text and COCO datasets yields F1 values of 83.43% and 84.97%, respectively, while achieving the lowest FID and FSD scores of 103.25 and 1.26. The combination of CLIP pre-training and word-level semantic embedding improves image quality, offering a novel approach for face recognition applications in public safety.

Keywords: CLIP model, Convolution-inverse convolution, BiLSTM model, StyleGAN, Text generation face

1. Introduction

The proportion of conceptual representations received by the human brain is based on different cognitive ideas, with visual information accounting for up to 87%. However, when some aspects of the content of the information to be conveyed cannot be directly represented by visual information, they can be described in advance by some text, and then the sentences can be converted into corresponding images, providing visual information that matches the input description, and vice versa [20, 16].

✉ Corresponding author.
E-mail addresses: mlliu@nefu.edu.cn (Meiling Liu).

Received 06 August 2024; accepted 09 October 2024; published 31 December 2024.

DOI: 10.61091/jcmcc123-12

Thus, visual and verbal information can potentially provide complete conceptual representations by providing complementary information through appropriate combinations and transformations. The accelerated development of the Internet has led to dramatic changes in people's lifestyles, and the growth of massive data, such as images and videos, has greatly satisfied many directions of research and greatly promoted research progress [6, 7]. However, in the process of development, crude data image resources can no longer meet people's research needs, people with the help of "generative model" on the image data resources already available to repeatedly learn and "create" new image resources to meet the diversified needs of the people, such as image translation, image hyperlinks, and so on. The applications include image translation, image super-resolution, image generation, image restoration, and so on [22, 1].

In the past few decades, significant technological breakthroughs have been made in computer vision and natural language processing. Researchers are increasingly interested in the fusion of traditional semantic and visual information [23, 3]. For example, caption generation, visual storytelling, and visual answering of questions. With the emergence of Generative Adversarial Networks, Text-Image Synthesis, Layout-Guided Image Generation and Text-Guided Image Generation have received sustained attention as challenging "visual + linguistic" tasks that help analyze the relationship between textual descriptions and realistic images [5, 8]. These works have a wide range of applications in art creation, image editing, and image retrieval. Text-image synthesis refers to the conversion of an input text description into a real image with similar semantic information to the keywords of the input sentence, with the aim of establishing an interpretable mapping relationship between the image space and the text semantic space. Text-face image synthesis, as a subfield of text-image synthesis, has a great potential in the field of public security [17, 18, 21].

Face image generation, aims to generate a face image that meets the constraints given the prerequisites (e.g., textual description of face features, facial keypoints, mask segmentation map, etc.) [2]. Currently, researchers are pursuing to achieve two goals in this area. Improving the realism of the generated face and enhancing the controllability of the target face. Among them, realism can be interpreted as the quality and realism of the generated face; controllability is reflected in the degree of matching of the generated face to the conditions [12]. Text-to-face generation aims to generate corresponding face images based on text descriptions. Currently, a multi-stage framework is commonly used for text-to-image generation. Specifically, a text encoder is utilized to extract text features from the text description, the text features are connected to the noise as the input to the generator, and then multiple generators are used to generate images step by step [13, 15].

Text-to-face image generation (T2F) is a meaningful and interesting research area in computer vision, aiming to generate real face images based on the input face description. In this paper, a model architecture for Chinese text-guided face image generation is established based on two modules, the text editor and the StyleGAN generator, and a corresponding loss function is designed to ensure the effectiveness of face image generation. The CLIP model is used to pre-train the Chinese text in the text editor, and the word-level semantic embedding model is constructed by combining the CDWE-BiLSTM model to obtain more comprehensive semantic information of the Chinese text. This is used as the input of StyleGAN generator, thus realizing the model construction of Chinese text to generate face images. In order to analyze the effectiveness of the model, the data are analyzed in terms of text semantic feature extraction and the effect of text-generated face images.

2. Modeling Framework for Text-Generated Face Images

Text Generated Face Images as a subfield of Text Generated Images, like most of the other text generated image approaches, existing T2Fs are based on a similar web framework in which a text encoder is utilized to encode the input face description information into semantic vectors. The image decoder then uses the semantic vectors, along with vectors randomly sampled from a particular distribution (e.g., Gaussian distribution) after stitching, as input vectors, to ultimately generate a photorealistic image. In this paper, the basic modeling framework for text-generated face images is established from the text mapping module, combined with StyleGAN, aiming to improve the effect of text-generated face images.

2.1. Overall framework of the model

2.1.1. Modeling framework. In order to realize text-generated face images, in this paper, a pre-trained StyleGAN generator is used to project text descriptions into the implicit space by training the mapper in order to improve the resolution and fidelity of the generated images. The text features are extracted by a pre-training method based on contrasting text-image pairs (CLIP) combined with the CWDE-BiLSTM model, which makes full use of the textual information, considers finer-grained textual features, and mines the textual semantics more deeply, thus realizing the generation of face images.

The network framework for cross-modal text generation of face images proposed in this paper is shown in Figure 1, and the whole network framework is mainly composed of text mapping module and StyleGAN generator. Among them, the text mapping module includes two parts, CLIP model and semantic embedding. First, the text description generates hidden vectors through the text processing network, and then the hidden variables are obtained from the de-entangled hidden vectors through the mapping network, and the de-entangled hidden vectors are input into the StyleGAN generator to generate the face image. In the text mapping module, convolution-inverse convolution is introduced to enhance the extraction of text features, combined with the BiLSTM model to obtain more complete text descriptions, so that the face images generated by StyleGAN are more consistent with the text descriptions, and better face generation results are achieved.

2.1.2. Loss function. The main role of the model's loss function is to help the model to achieve effective training of the data during the training process, the main StyleGAN loss function involved in this paper, mainly including content loss, adversarial loss and structural loss. Parameters α, β are used to control several losses with different weights. The total generator loss is denoted as:

$$L_G = L_{content} + \alpha L_{adversarial} + \beta L_{ssim}. \tag{1}$$

The purpose of the content loss is both to increase the similarity of the face image to the textual description and to increase the similarity of the face image to the textual description, and the content loss is defined as the sum of these two losses. The image pixel loss is defined as follows:

$$L_{content} = Content_{ir} + \gamma Content_{vi}, \tag{2}$$

$$Content_{ir} = \frac{1}{WH} \sum_{x=1}^{W} \sum_{y=1}^{H} \left(I_{i_{(x,y)}} - I_{f_{(x,y)}} \right)^2, \tag{3}$$

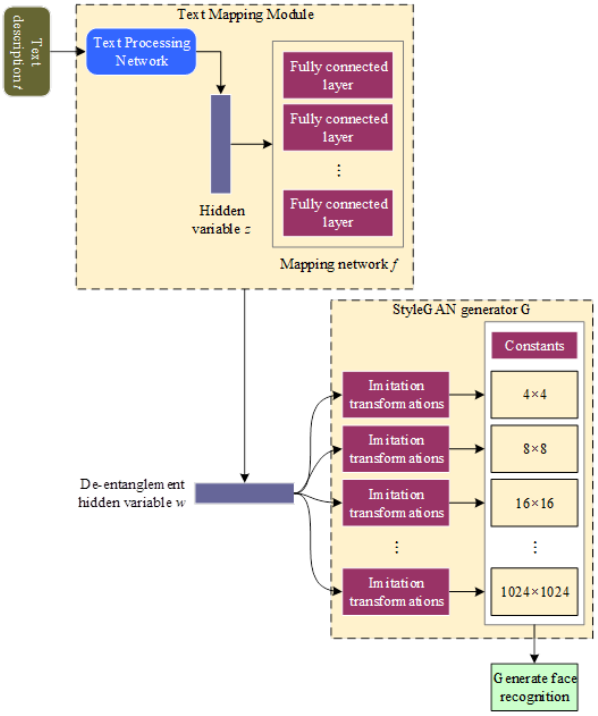

Fig. 1. Global network framework

$$Content_{vi} = \frac{1}{WH}\sum_{x=1}^{W}\sum_{y=1}^{H}\left(I_{vi_{(x,y)}} - I_{f_{(x,y)}}\right)^2, \tag{4}$$

where I_{ir} is the original image, I_{vi} is the face image, I_f is the final output of the generator, W and H denote the width and height of the image, and γ is used to control the weight of pixel loss. The image pixel loss makes the fused image consistent with the original face image in terms of pixel intensity distribution. The quadratic L2 paradigm is chosen here due to the fact that the L2 paradigm is more conductive and easier to train compared to the L1 paradigm.

Adversarial loss is designed for better image generation. Adversarial loss is defined based on the discrimination accuracy of the discriminator i.e. $\log D_{\theta_D}\left(G_{\theta_G}\left(I^{mix}\right)\right)$. Adversarial loss is as follows:

$$L_{adversarial} = \sum_{n=1}^{N}\left(1 - \log D_{\theta_D}\left(G_{\theta_G}\left(I^{mix}\right)\right)\right), \tag{5}$$

where I^{mix} denotes the face image, and $\log D_{\theta_D}\left(G_{\theta_G}\left(I^{mix}\right)\right)$ is the probability that the discriminator will not be able to discriminate between a real and a fake image. N is the amount of data fed into the network each time for training.

The structural similarity metric measures the similarity of two images, which is influenced by three different parts: the brightness of the image, the structure of the image and the contrast of the image, and the structural similarity is defined as:

$$SSIM_{X,F} = \sum_{x,f}\frac{2\mu_x\mu_f + c_1}{\mu_x^2 + \mu_f^2 + c_1}\cdot\frac{2\sigma_x\sigma_f + c_2}{\sigma_x^2 + \sigma_f^2 + c_2}\cdot\frac{\sigma_{xf} + c_3}{\sigma_x\sigma_f + c_3}, \tag{6}$$

where x and f refer to the original image and the generated image, respectively, and the structural similarity between the original image x and the generated image f is represented by $SSIM_{X,F}$, μ_x and μ_f represent the mean of the original image and the generated image, $\sigma_{x,f}$ represents the covariance

of the original image and the generated image, and σ_x and σ_f represent the standard deviation of the original image and the generated image. So, the structural similarity loss in this paper is defined as:

$$L_{ssim} = 2 - \omega_a SSIM_{I,F} - \omega_b SSIM_{V,F}, \tag{7}$$

where $SSIM_{I,F}$ and $SSIM_{V,F}$ represent the structural similarity between the generated image and the original image respectively. ω_a and ω_b represent the different weights of the parts, and the smaller L_{ssim} represents the higher degree of similarity between the generated image and the original image, i.e., the better the generation effect.

2.2. Text mapping module

2.2.1. CLIP model. In the text mapping module, the CLIP model is mainly utilized to pre-train the text features and use them as inputs for word-level semantic embeddings so as to better mine the text semantic features in conjunction with the BiLSTM model.

The CLIP model used in this paper is a visual language model based on multimodal contrastive learning, which demonstrates the potential of learning open vocabulary visual concepts.The CLIP model is constructed with two encoders, one for images and the other for text. The image-text pairs are first given, after which they are fed to the encoder of the corresponding modality. The image encoder can be either ResNet or Visual Transformer for transforming images into feature vectors. The text encoder can be either a continuous bag-of-words model or a Transformer, which takes a sequence of word tokens as input to produce a vectorized representation [10].

During training, CLIP uses contrast loss to learn the joint embedding space of the two modalities. Specifically, for a batch of image-text pairs, CLIP maximizes the cosine similarity of each image to the matching text while minimizing the cosine similarity to all other mismatched texts, and computes the loss for each text in a similar manner. After training, CLIP can be used for zero-sample image recognition, and this powerful zero-sample inference capability gives CLIP flexibility.

Let x be an image feature generated by an image encoder, and $\{w_i\}_{i=1}^{K}$ be a set of embedding vectors generated by a text encoder, where each weight vector represents a class (assuming a total of K classes). In particular, each w_i comes from a cue, e.g., "aphotoofaclass", where the "class" lexical element is populated with the ith class name. The predicted probability can then be expressed as:

$$p\left(y\,|x\right) = \frac{\exp\left(sim(x, w_y)\,/\,\tau\right)}{\sum_{i=1}^{K} \exp\left(sim(x, w_i)\,/\,\tau\right)}, \tag{8}$$

where $sim\left(\cdot,\cdot\right)$ denotes the cosine similarity and τ is the learnable temperature parameter.

2.2.2. Word-level semantic embedding. In order to enhance the feature extraction capability for textual descriptions, this paper introduces a convolutional-inverse convolutional-based word-level semantic embedding approach (CDWE), which is an end-to-end multi-prototype word vector generation model that fuses context-specific and task-specific information to substitute for relevant gaps [19].

The overall process of generating CDWE can be divided into four steps as follows:

(a) One-dimensional convolutional layer. One-dimensional convolution is the most used convolution in Natural Language Processing (NLP), where one of the dimensions of each convolution kernel is the same as that of the pre-trained word vectors, and thus has only one degree of freedom (i.e., the length of the convolution kernel). In this model the convolution kernel length

is set to 1, i.e., the convolution parameter size is $1 \times H$, which does not involve any interaction function between words. Then:

$$s_i^j = ReLU\left(\omega \cdot e_i + b\right). \tag{9}$$

The s_i^j in Eq. represents the semantic feature vector extracted from the ird word vector e_i by the jnd convolutional kernel in the convolutional layer, $j = 0, 1, 2, \ldots, N$, N represent the number of convolutional kernels, b is the bias, and the activation function of the convolutional layer is $ReLU$.

(b) Pooling Layer.The CDWE generation process uses the maximum pooling operation applied to different channels, which allows the layer to capture more complex and learnable interactions across channel information. Then:

$$c_i^{\left[\frac{j}{k}\right]} = \max\left\{s_i^j, s_i^{j+1}, \ldots, s_i^{j+k-1}\right\}, \tag{10}$$

where $c_i^{\left[\frac{j}{k}\right]}$ represents the $\left\lceil \frac{j}{k} \right\rceil$ features extracted from the features generated by the k convolutional filter, and $\lceil\rceil$ is an operation representing upward rounding. The maximum pooled word vector generated by the maximum pooling layer is $\widehat{c_i}$, i.e:

$$\widehat{c_i} = c_i^1 \oplus c_i^2 \oplus \ldots \oplus c_i^{\frac{N}{k}}, \tag{11}$$

where N represents the number of convolution kernels of the convolutional layer, where $N = H$ in this encoder.

(c) Inverse convolution layer. The 2D deconvolution (convolution transpose) applies window size (1,k) and step size (1,k) to the deconvolution operation. Specifically, the spatial resolution is changed from $\widehat{c_i} \in R^{T \times N/k}$ to $V^d \in R^{T \times N}$, which in turn matches the resolution up-sampling of the inverse convolution operation, and the final output semantic information matrix is $V \in R^{T \times N \times D}$ its $d = 0, 1, 2, \ldots, D$. D denotes the number of inverse convolution filters, which denotes the number of prototypes for each word, i.e., the number of alternative vectors [14].

(d) Multiple prototype selection. A word may carry different meanings when it appears in different contexts. Many archetypes should be created for the word so that each archetype has a specific meaning. Thus, in order to learn multiple prototypes, each word can be associated with multiple prototypes. Training the network to distinguish the selected word vectors from each other makes it possible to select one of the word vectors by the match of the vectors to fit a particular context during the inference process. Then:

$$idx_i = \underset{j=1,2....d}{Arg\max}\left(Similarity\left(v_i^j, ctx_i\right)\right), \tag{12}$$

$$v_i = v_i^{idx_i}, \tag{13}$$

Here idx_i represents the index of the selected ind vector, i is the number of the word, $i = 0, 1, 2, \ldots, T$; v_i^c is the final vector selected from all the D different alternatives, i.e., CDWE. ctx_i represents the vector of contextual representations of the word, which is computed as:

$$ctx_i = \left(e_{i-2} + e_{i-1} + e_i + e_{i+1} + e_{i+2}\right)/5. \tag{14}$$

Similarity

$$\left(v_i^j, ctx_i\right) = \frac{v_i^j \cdot ctxx_i}{\left\|v_i^j\right\| \left\|ctx_i\right\|}. \tag{15}$$

The window approach is used in this encoder, which assumes that the meaning of a word depends largely on the 5 words in its context when a word appears in different contexts, it may have different meanings. Many prototypes should be created for the word so that each prototype e_i can have a specific meaning. Thus, in order to learn multiple prototypes, each word can be associated with more than one prototype. The training network distinguishes the selected word vectors from each other so that one of the word vectors e_i is selected during the inference process by the extent to which the vectors fit into a particular context.

2.2.3. Text encoder. Semantic Embedding Approach (CDWE) is fed into a Bidirectional Long Short Term Memory Network Layer (BiLSTM) with 256 hidden neurons.The BiLSTM layer consists of Forward and Backward LSTM units. The inputs to each time step of the BiLSTM layer are the same. The BiLSTM all time steps' output vectors are concatenated and output to the image decoder for subsequent decoding of the synthesized image.

The structure of the forward LSTM cell and the backward LSTM cell is formulated such that each LSTM network consists of the cell state, three gates (input gate, forgetting gate, and output gate) and the process of updating the input information at each time step t [9]. Then:

$$\begin{cases} i_t = \sigma\left(W_i \cdot [h_{t-1}; x_t] + b_i\right), \\ f_t = \sigma\left(W_f \cdot [h_{t-1}; x_t] + b_f\right), \\ e_t^{sent} = \sigma\left(W_{e^{e=0}} \cdot [h_{t-1}; x_t] + b_{e^{emt}}\right), \\ g_t = \tanh\left(W_r \cdot [h_{t-1}; x_t] + b_r\right), \\ c_t = i_t \odot g_t + f_t \odot c_{t-1} h_t = e_t^{sent} \odot \tanh\left(c_t\right), \end{cases} \tag{16}$$

where i_t, f_t, e_t^{sent} represent the input gate, forgetting gate and output gate of the three gates respectively, g_t is the candidate value of the cell state, c_t is the updated cell state at moment t, and h_t is the value of the hidden state at moment t. The output of the last hidden state of the bidirectional LSTM network is spliced as the semantic feature vector of the whole sentence, and the final output matrix of the text encoder is e^{sent}, $e^{sent} \in R^{256}$. Sentence vector e^{sent} is connected with a segment of noise vector that conforms to normal distribution as the input of the image decoder.

2.3. StyleGAN Generator

2.3.1. Generating Adversarial Networks.

(a) Generative Adversarial Network (GAN)

GAN combines a generative model and a discriminative model.The purpose of the generator is to learn the distributional characteristics of the training set data, and through the input random noise, finally generate data with almost the same distribution as the original training set data. The discriminator is similar to a deep neural network, its purpose is to distinguish the truth of the input training set data from the generated data, which is equivalent to a classification model. The GAN network realizes the generation of data through the mutual training between the generator and the discriminator [11].

The optimization function of a GAN network is a splittable max-min problem which is expressed as:

$$\min_G \max_D V(D,G) = E_{p_{data}}(x)\left[\log D(x)\right] + E_{p_z}(z)\left[\log\left(1 - D\left(G\left(z\right)\right)\right)\right], \tag{17}$$

where, p_{data} is the distribution of the real training set data, p_z is the distribution of the generated data, $G(z)$ is the data generated by the generator, $D(x)$ is the probability that the

discriminator discriminates the real data as true, and $D(G(z))$ is the probability that the discriminator discriminates the generated data as true;

As can be seen from Eq. (17) During the training of the GAN network, the parameters of the generator are fixed first, and the value of the optimization function is maximized by continuously updating the parameters of the discriminator. Then the parameters of the discriminator are fixed and the parameters of the generator are optimized so that the optimization function reaches the minimum value. When the generated data distribution of the generator and the data distribution of the original training set are equal, the output probability of the discriminator is 0.5. According to the above confrontation analysis, since the real data is not considered in the generator, the above equation can be split as follows:

$$\begin{cases} \max\limits_{D} V(D) = E_{p_{data}}[\log D(x)] + E_{p_t}[\log(1 - D(G(z)))], \\ \min\limits_{G} V(G) = E_{p_t}[\log(1 - D(G(z)))]. \end{cases} \tag{18}$$

(b) StyleGAN

StyleGAN, as one of the models derived from GAN, follows the generator/discriminator relationship of the GAN model. During each iteration, the generator and the discriminator are essentially a pair of game relations, and the objective functions of the generator and the discriminator are respectively:

$$G^* = \arg\min_{G}\max_{D} V(G, D), \tag{19}$$

$$D^* = \arg\max_{D} V(G, D). \tag{20}$$

In Style GAN, a mapping network is a neural network used to map z vectors in the latent space to a controlled intermediate representation w of the vectors. This intermediate representation w appears as a "style" concept throughout the layers of the generator (r) network [4].

2.3.2. StyleGAN Generator. In this paper, the output of CDWE-BiLSTM model is used as the input of StyleGAN model as a way to realize the generation of face images.The network structure of StyleGAN generator is shown in Figure 2. It consists of two parts, the first one is Mapping network, which is a process of generating intermediate hidden variables from hidden variables, and this intermediate hidden variable is used to control the style of the generated image. The second one is Synthesis network, which is used to generate the image, and the innovation is that each sub-network layer is inputted with A and B, and A is the affine transformation obtained by transforming the intermediate hidden variables. The innovation is that each sub-network is fed with A and B. A is the affine transform transformed by the intermediate hidden variable, which is used to control the style of the generated image, and B is the transformed random noise, which is used to enrich the details of the generated image, such as wrinkles and so on, that is, each convolutional layer can adjust the "style" according to the input of A, and adjust the details via B.

The workflow of the model is as follows:

Input the output of the CDWE-BiLSTM model, first convert to get a hidden code, and then input the hidden code into the Mapping network decoupling, to get an intermediate vector, these intermediate vectors will be subsequently passed to the generative network to get 18 control vectors, these 18 control vectors two by two, passed into the Synthesis network of nine convolutional layers

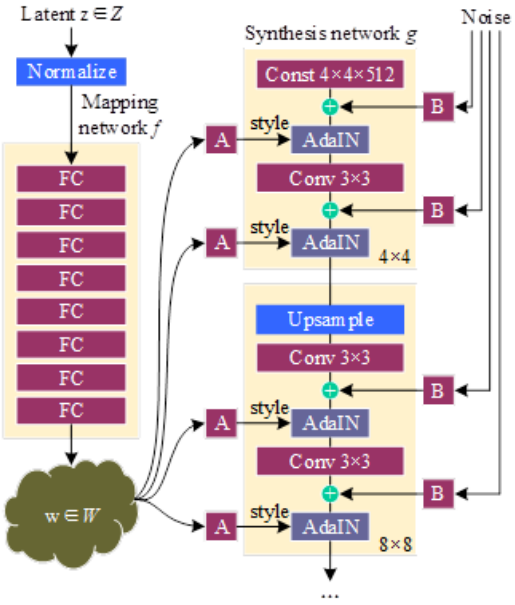

Fig. 2. The network structure of the StyleGAN generator

These 18 control vectors are passed in pairs to the AdaIN module of the Synthesis network, and a scaled noise is added to each channel before each AdaIN module, and finally a virtual face image based on the text description is output.

(a) Mapping network: With the increase in the number of convolutional layers, the phenomenon of feature entanglement between a large number of subsequent control vectors occurs, and the Mapping network cleverly solves the problem of feature entanglement, and the Mapping network consists of eight fully connected layers.

(b) Synthesis network is the core part of StyleGAN, its main role is to merge the potential vectors of Mapping network with the noise vectors and pass the merged vectors to the generator network to generate images.

(c) Fine-tuning. AdaIN is an important module in StyleGAN and its formula is:

$$AdaIN\left(x_i, y\right) = y_{s,i}\frac{x_i - \mu\left(x_i\right)}{\sigma\left(x_i\right)}. \tag{21}$$

Expanding w by a learnable affine transform into a scaling factor $y_{s,i}$ and a deviation factor $y_{b,i}$, which will do a weighted sum with the normalized convolutional output, completes the process of w influencing the original output x_i at a time. By fine-tuning the scaling factor, the adjustment of face image information can be realized.

3. Validation of Text-Generated Face Image Models

With the development of deep convolutional technology, the application of text-driven face image synthesis and editing in the fields of live video broadcasting and AI attack and defense has been widely expanded to achieve semantically consistent editing of face images. The goal of text-driven face image generation is to generate face images through text manipulation, and face images have a wide range of applications in various fields, including but not limited to identity authentication, pedestrian re-identification, and security monitoring. The richness of this series of application scenarios not only

highlights the importance of face images in social life, but also its far-reaching impact in multiple fields, which has become a distinctive symbol of today's digital era. This chapter focuses on the validation of the relevant models for text-generated face images, which provides a certain foundation for the research of text-generated face images.

3.1. Text semantic feature verification

In order to verify the effectiveness and accuracy of the CDWE-BiLSTM text encoder in encoding textual semantic features in the task of text-generated face images, we conducted a series of validation experiments on two representative public datasets Face2Text, and COCO text descriptions used in this paper, and the specific details of the experiments and the experimental results will be shown in the following subsections.

3.1.1. Changes in training losses. Taking Face2Text dataset as an example, it is divided into training set and testing set according to the ratio of 8:2. The CDWE-BiLSTM model is trained on the training set using the gradient descent method, and the accuracy and F1 value are selected as the evaluation indexes. After the training is completed the trained CDWE-BiLSTM model is validated on the test set and the changes in the metrics are recorded. Figure 3 shows the changes of the evaluation indexes of the CDWE-BiLSTM model during the training process, in which Figures 3(a)-(b) show the trends of the accuracy rate and F1 value and loss value, respectively.

As can be seen from the figure, with the increase of the number of iterations, the training accuracy and F1 value of the model are increasing, and when the number of iterations reaches 65 Epochs its training accuracy and F1 value are maintained at about 85%, and the testing accuracy and F1 value stabilizes at about 86% after 72 Epochs of iterations. This indicates that the CDWE-BiLSTM model has been optimized for training at this time. Similarly, in the iteration of training and testing loss of the model, it reaches the minimum value after 70 Epoch iterations, i.e., it has reached the expected goal of model training.

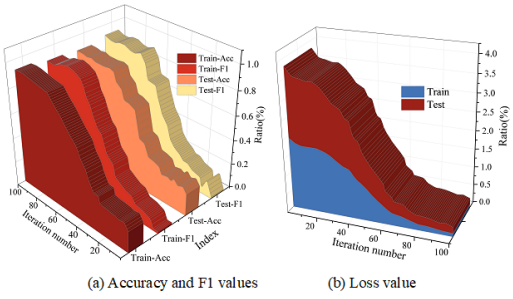

Fig. 3. Changes in evaluation indicators

3.1.2. Semantic extraction performance. After the training of the CDWE-BiLSTM model is completed, the two datasets Face2Text and COCO are taken as examples, and CNN-SVM, NLM-RAE, ARC-I, ARC-II, AM-CNN, BRNN, Bi-CNN-MI, and BR-BiLSTM are selected as the comparison models of the models in this paper, as a way to verify the CDWE- BiLSTM model is effective in acquiring semantic features of text to provide reliable semantic features for text generation of face images. Table 1 shows the comparison results of the semantic feature extraction performance of different models on two datasets.

As can be seen from the table, the performance of this paper's model on both datasets is sig-

nificantly higher than the rest of the comparison models, with F1 values of 83.43% and 84.97% on Face2Text and COCO datasets, respectively.The CNN-SVM model utilizes WordNet-based lexical similarity metrics, which mainly employs a kind of textual entailment idea.The NLM-RAE is calculated by utilizing a recursive autoencoder for computation.ARC-I is based on the twin system, which has a convolution process trained on the paraphrasing task, but the method deals with inter-sentence interactions at the end, i.e., two sentence vectors can interact with each other only after they have been computed.ARC-II uses a multilayered neural network instead of a twin structure, and the approach can deal with the individual representations that are generated through the intertwining of the two sentence forms. Although ARC-II uses cross-convolution, it is not able to deal with mining the depth of information in a sentence. The model in this paper starts from "convolution-inverse convolution" and combines with BiLSTM model to extract semantic features, which can not only obtain word information, but also fully explore the sequential information in the sentence, and thus achieve better results. computation.BRNN model utilizes word similarity information derived from WordNet.Bi-CNN-MI method is based on paraphrase recognition and uses convolution to learn multi-granularity sentence representation and computes interaction feature matrix at each level.BR-BiLSTM method is to predict the target word by using all the n-grams in the window, and the average of the final vector representation of all the n-grams which is the sentence vector. Compared with other comparison methods, the CDWE-BiLSTM model proposed in this paper achieves better results. It can be seen that the CDWE-BiLSTM model in this paper performs word-level semantic embedding by convolution-inverse convolution method, and combines the deep model of text semantic feature extraction with BiLSTM model for semantic learning, which can effectively enhance the information interaction and thus enrich the semantic understanding, and it can provide reliable semantic support of text encoding for text-to-face image generation task.

Model	Face2Text		COCO	
	ACC (%)	F1 (%)	ACC (%)	F1 (%)
CNN-SVM	65.41	75.21	66.48	76.71
NLM-RAE	70.42	81.34	73.35	83.24
ARC-I	69.38	80.62	68.85	81.53
ARC-II	69.46	80.95	69.47	81.82
AM-CNN	73.41	81.23	72.06	83.05
BRNN	74.78	81.75	75.38	82.78
Bi-CNN-MI	72.34	81.49	73.47	82.64
BR-BiLSTM	72.66	80.36	73.92	81.53
Ours	75.98	83.43	78.65	84.97

Table 1. Semantic feature extraction performance comparison results

3.1.3. Parameter sensitivity. Different parameters in the CDWE-BiLSTM model may have an impact on the semantic perceptual ability of the text encoder, making the semantic features extracted by the model ineffective in providing reliable semantic descriptions for the task of text generation of face images. In this regard, the parameter sensitivity of the CDWE-BiLSTM model is explored in the middle of this section, and the experiment will explore the sensitivity of the model from four parts: the number of layers, the sliding window size, the learning rate and the Dropout ratio. Figure 4 shows the validation results of the parameter sensitivity of the model, where Figures 4(a)-(d) shows

the validation results of the number of graph layers, sliding window size, learning rate and Dropout ratio, respectively.

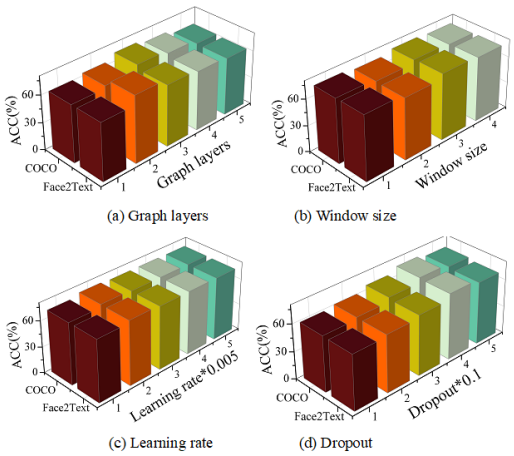

Fig. 4. Validation results of parameter sensitivity

(a) Graph layers represent the number of layers of the CDWE-BiLSTM model, and it can be seen from the figure that as the number of layers of the CDWE-BiLSTM model changes, the accuracy of the model on different datasets is different. When Graph layers are equal to 2, the model performs best on Face2Text dataset (73.64%) and when Graph layers are equal to 3, the model performs best on COCO dataset (72.13%). This indicates that as the number of layers increases, the nodes can obtain more information about their neighbors, but due to the limited learning ability of the nodes, the test accuracy starts to decrease when the number of layers reaches a certain value.

(b) Window size indicates the sliding window size, and the performance of the CDWE-BiLSTM model is different for different window sizes. From the figure, it can be seen that with the increase of Window size, the test accuracy of the model on Face2Text and COCO dataset increases accordingly, and when the test accuracy reaches the peak (75.78%, 74.94%), the model performance also begins to decline, which indicates that the model's performance is also affected by the Window size.

(c) Under the influence of different Learning rate, the CDWE-BiLSTM model learns more and more semantic information as the learning rate increases. If there is no early assignment, the model will reach the fitting point earlier, which affects the actual performance of the model.

(d) On Face2Text and COCO datasets, different Dropout values affect the accuracy of the model.The increase of Dropout value is similar to the trend of the learning rate.In the experiments of the model, the model has the best performance on the two datasets when the Dropout value is 0.4, and its accuracy is 72.19% and 73.25%, respectively.

In summary, when using the CDWE-BiLSTM model for the text editor for text generation of face images, when the Graph layers of the CDWE-BiLSTM model are set to 2 or 3 layers, Window size is set to 2, Learning rate is set to 0.01, and the Dropout ratio is set to 0.4, we can Obtaining the optimal text semantic perception capability allows the model to deeply acquire the relevant information of the text description, providing reliable text semantic support for the accurate generation of text-generated face images task.

3.2. Text generated face image analysis

3.2.1. Comparison of generation performance. Text-guided face image generation usually evaluates the model based on objective evaluation criteria. In this section, the evaluation will be done in terms of quality of generated face images, semantic consistency of face images and text descriptions, and similarity between face images and real images. In this paper, FID is used to evaluate the quality of generated face images, R-PRE is used to evaluate the semantic consistency of face images and text descriptions, and Face Similarity Score (FSS) and Face Similarity Distance (FSD) are used to evaluate the similarity between face images and real images.

In this paper, Multi-Modal CelebA and CelebAText are selected as the experimental dataset, and gradient descent and CLIP models are used to pre-train the model for text-guided face image generation, and the effect of the model on the generation of face images is evaluated after the training is completed. StackGAN++, AttnGAN, MirrorGAN, ControlGAN, and DFGAN are mainly selected as the comparison models, and the comparative results of the performance of different methods for generating face images on Multi-Modal CelebA and CelebAText datasets are obtained as shown in Table 2.

Model	FID↓	FSD↓	FSS↑	R-PRE ↑
Multi-Modal CelebA				
StackGAN++	136.42	1.51	0.35	-
AttnGAN	126.94	1.42	0.42	46.35%
MirrorGAN	122.37	1.38	0.46	51.48%
ControlGAN	115.38	1.37	0.53	53.27%
DFGAN	136.74	1.59	0.46	43.51%
Ours	103.25	1.26	0.57	59.69%
CelebAText				
StackGAN++	140.14	1.79	0.25	-
AttnGAN	129.81	1.63	0.33	42.43
MirrorGAN	119.39	1.54	0.41	48.95
ControlGAN	107.92	1.38	0.53	53.24
DFGAN	141.53	1.51	0.38	40.17
Ours	98.35	1.22	0.59	60.48

Table 2. The results of the performance comparison of face image

From the table, it can be seen that the text-generated face image model proposed in this paper achieves the lowest FID and FSD on the Multi-Modal CelebA dataset with the values of 103.25 and 1.26 respectively, as well as the highest FSS and R-PRE with the values of 0.57 and 59.69% respectively. This indicates that the text-generated face image model proposed in this paper outperforms the comparison methods in terms of the quality of the generated face image, the semantic consistency of the face image and the text description, and the similarity with the real image, which validates the effectiveness of the model in this paper. Among them, compared with ControlGAN, which is the best performer among the comparison methods, this paper's model reduces 12.13 in the FSD index, improves 0.04 in the FSS index, and improves 6.42% in the R-PRE index. This indicates that the face images generated by this paper's model are closer to the real images, as well as the semantic consistency between the textual descriptions and the face images is also higher. In addition, although

the method in this chapter achieves the best results on two evaluation metrics, Face Similarity Score (FSS) and Face Similarity Distance (FSD), there is still a certain gap between the generated face images and the real images, which is due to the fact that there is a certain degree of randomness in the generation process, and it is impossible for the generated face images to be completely identical to the real images.

In addition, in the results of the comparison experiments on the CelebAText dataset, when the text description becomes more complex and fine-grained, the performance of StackGAN++, AttnGAN, MirrorGAN, and DFGAN degrades to varying degrees, which is due to the inability of their discriminators to provide the generator with fine-grained training feedbacks, which leads to the deterioration in the robustness of the generator. Since the ControlGAN method proposes a word-level discriminator, its performance on the CelebAText dataset is somewhat improved, but the word-level discriminator of the method ignores the word information within the textual features, which makes the generator unable to establish an accurate connection between the face attributes and words, and thus its performance improvement is very limited. The model in this paper outperforms the ControlGAN method on the CelebAText dataset, while the performance is also improved. This is because this paper combines the convolution-inverse convolution method with BiLSTM to deeply mine the text semantic information as a way to realize the construction of the text encoder, and the resulting CDWE-BiLSTM model can establish an accurate connection between each word and the face attributes, which makes the generator robust, and the generator's performance will not degrade even if the text description becomes more complex and abstract.

3.2.2. Ablation experiments. In order to demonstrate the effectiveness of the CDWE, BiLSTM model, and StyleGAN model proposed in this paper, ablation experiments are conducted on Multi-Modal CelebA and CelebAText datasets. A total of four sets of component ablation experiments are set up for the baseline model, +CDWE, +BiLSTM, and +StyleGAN, respectively, and the base model is set up as the GAN model. Table 3 shows the comparison results of the ablation experiments.

As can be seen from the table, adding convolution-inverse convolution, BiLSTM model, and StyleGAN model to the original GAN model have positive moderating effects on the face image generation results, and adding the corresponding modules to both datasets respectively can effectively improve the FID, FSS, and R-PRE scores. On the Multi-Modal CelebA dataset, if only CDWE is added, the FSS score of the baseline model increases from 0.33 to 0.42, the FID score decreases from 163.51 to 151.28, and the R-PRE score improves from 52.18% to 53.24%. If only the BiLSTM model is augmented at the baseline model, the baseline model FSS score grows to 0.49, the FID score decreases to 128.24, and the R-PRE score improves to 54.39%. If the baseline model is improved to StyleGAN model, the baseline model FSS score grows to 0.48, FID score decreases to 128.24, and R-PRE score improves to 53.95%. Ultimately combining the three modules gives the best results of the method in this paper and proves the effectiveness of the method in the paper, combining the CDWE-BiLSTM model with the StyleGAN model can realize the text generation face task.

3.2.3. Realism for different sample sizes. In order to further investigate the realism of face images generated by this model, we set up the following experiments. Using the FID score as a qualitative analysis index, 5000∼40000 face images are extracted from the original database respectively, and the text label and key point of each image are marked, and the same number of virtual face images are generated using this label and key point as input. Then the FID metrics of real image data and virtual image data are calculated separately, from which the approximation

Model	Multi-Modal CelebA			CelebAText	
	FID↓	FSS↑	R-PRE ↑	FID↓	FSS↑
GAN	163.51	0.33	52.18%	175.35	0.31
+CDWE	151.28	0.42	53.24%	163.24	0.38
+BiLSTM	128.24	0.49	54.39%	139.71	0.42
+StyleGAN	129.43	0.48	53.95%	133.48	0.41
+CDWE +BiLSTM	114.85	0.44	55.98%	119.24	0.36
+CDWE+StyleGAN	112.64	0.53	56.46%	115.38	0.48
+BiLSTM+StyleGAN	108.51	0.51	57.13%	110.63	0.47
+CDWE+BiLSTM+StyleGAN	103.25	0.57	59.69%	98.35	0.59

Table 3. The comparison of ablation experiment

between the distribution of generated face images and the real distribution is evaluated. Table 4 shows the results of the realism exploration experiments under different sample numbers.

For the case of data volume of 5000∼40000, it can be seen that the virtual data distribution matches the real data distribution by more than 95%. This fully demonstrates that the face images generated by the model already have a high degree of realism and authenticity, which further verifies the superiority of the model in this paper.

Sample size	FID-Truth	FID-Virtual	Matching degree (%)
5000	58.37	59.57	97.99%
10000	59.42	61.68	96.34%
15000	54.65	56.91	96.03%
20000	63.16	64.54	97.86%
25000	48.95	51.39	95.25%
30000	57.51	59.13	97.26%
35000	61.64	63.66	96.83%
40000	48.83	50.05	97.56%

Table 4. The truth explores the results of the experiment

4. Conclusion

In this paper, a text-generated face image model combining a text encoder with CDWE-BiLSTM model and StyleGAN generator is proposed, and pre-training is carried out using CLIP model. And the validation analysis is carried out for the effectiveness of the model, and the conclusions are as follows:

(a) In word-level semantic embedding using the CDWE-BiLSTM model, the test accuracy and F1 value are stabilized at around 86% after 72 Epoch iterations, and the F1 values on the Face2Text and COCO datasets are 83.43% and 84.97%, respectively. The use of convolution-inverse convolution for deep parsing of text descriptions, combined with BiLSTM model acquisition to achieve semantic perception, which is used as the input to the text generation face image model, can better help the model to understand the semantic information of the text, so as to generate face images that are more consistent with the text description.

(b) On the Multi-Modal CelebA dataset, the FID and FSD of the text-generated face image model in this paper have the lowest values of 103.25 and 1.26, and the highest values of 0.57 and 59.69% for the FSS and R-PRE, respectively. The ablation experiment verifies the effectiveness of the CDWE-BiLSTM model in the text editor for the model to realize semantic perception, and provides an accurate and efficient semantic feature representation method to enhance the effect of text-generated face images, so as to generate more realistic face images.

Funding

This work was supported by the National Natural Science Foundation of China (Grant No. 61702091) and the Natural Science Foundation of Heilongjiang Province (Grant No. LH2022F002).

References

[1] J. Bao, D. Chen, F. Wen, H. Li, and G. Hua. Cvae-gan: fine-grained image generation through asymmetric training. *Proceedings of the IEEE international conference on computer vision*:2745–2754, 2017.

[2] O. Bar-Tal, D. Ofri-Amar, R. Fridman, Y. Kasten, and T. Dekel. Text2live: text-driven layered image and video editing. *European conference on computer vision*:707–723, 2022. https://doi.org/10.1007/978-3-031-19784-0_41.

[3] S. Y. Cheong, A. Mustafa, and A. Gilbert. Upgpt: universal diffusion model for person image generation, editing and pose transfer. *Proceedings of the IEEE/CVF International Conference on Computer Vision*:4173–4182, 2023.

[4] A. M. M. Chowdhury, M. J. A. Khondkar, and M. H. Imtiaz. Advancements in synthetic generation of contactless palmprint biometrics using stylegan models. *Journal of Cybersecurity and Privacy*, 4(3):663–677, 2024. https://doi.org/10.3390/jcp4030032.

[5] M. Ding, Z. Yang, W. Hong, W. Zheng, C. Zhou, D. Yin, J. Lin, X. Zou, Z. Shao, H. Yang, and J. Tang. Cogview: mastering text-to-image generation via transformers. 34:19822–19835, 2021. M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors.

[6] O. Gafni, A. Polyak, O. Ashual, S. Sheynin, D. Parikh, and Y. Taigman. Make-a-scene: scene-based text-to-image generation with human priors. *European Conference on Computer Vision*:89–106, 2022. https://doi.org/10.1007/978-3-031-19784-0_6.

[7] S. Gu, D. Chen, J. Bao, F. Wen, B. Zhang, D. Chen, L. Yuan, and B. Guo. Vector quantized diffusion model for text-to-image synthesis. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*:10696–10706, 2022.

[8] M. Kang, J.-Y. Zhu, R. Zhang, J. Park, E. Shechtman, S. Paris, and T. Park. Scaling up gans for text-to-image synthesis. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*:10124–10134, 2023.

[9] Z. Li, X. Zhang, and W. Gao. State of health estimation of lithium-ion battery during fast charging process based on bilstm-transformer. *Energy*, 311:133418, 2024. https://doi.org/10.1016/j.energy.2024.133418.

[10] H. Liao, J. Yuan, C. Liu, J. Zhang, Y. Yang, H. Liang, H. Liu, S. Chen, and Y. Li. One novel transfer learning-based clip model combined with self-attention mechanism for differentiating the tumor-stroma ratio in pancreatic ductal adenocarcinoma. *La radiologia medica*:1–16, 2024. https://doi.org/10.1007/s11547-024-01902-y.

[11] Z. Liu, X. Zhou, H. Yang, Q. Zhang, L. Zhou, Y. Wu, Q. Liu, W. Yan, J. Song, M. Ding, et al. Reconstruction of reflection ultrasound computed tomography with sparse transmissions using conditional generative adversarial network. *Ultrasonics*, 145:107486, 2025. https://doi.org/10.1016/j.ultras.2024.107486.

[12] J. Ma, J. Liang, C. Chen, and H. Lu. Subject-diffusion: open domain personalized text-to-image generation without test-time fine-tuning. *ACM SIGGRAPH 2024 Conference Papers*:1–12, 2024. https://doi.org/10.1145/3641519.3657469.

[13] L. Ma, Q. Sun, S. Georgoulis, L. Van Gool, B. Schiele, and M. Fritz. Disentangled person image generation. *Proceedings of the IEEE conference on computer vision and pattern recognition*:99–108, 2018.

[14] Á. Madarász and G. Laczkó. Application of deconvolution in path integral simulations. *Journal of Chemical Theory and Computation*, 20(21):9562–9570, 2024. https://doi.org/10.1021/acs.jctc.4c00564.

[15] S. Nam, Y. Kim, and S. J. Kim. Text-adaptive generative adversarial networks: manipulating images with natural language. *Advances in neural information processing systems*, 31, 2018.

[16] O. Patashnik, Z. Wu, E. Shechtman, D. Cohen-Or, and D. Lischinski. Styleclip: text-driven manipulation of stylegan imagery. *Proceedings of the IEEE/CVF international conference on computer vision*:2085–2094, 2021.

[17] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman. Dreambooth: fine tuning text-to-image diffusion models for subject-driven generation. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*:22500–22510, 2023.

[18] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman, P. Schramowski, S. Kundurthy, K. Crowson, L. Schmidt, R. Kaczmarczyk, and J. Jitsev. Laion-5b: an open large-scale dataset for training next generation image-text models. 35:25278–25294, 2022. S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors.

[19] T. Tichter, A. Tichter, D. Andrae, and C. Roth. Simulating cyclic voltammetry at rough electrodes by the digital-simulation–deconvolution–convolution algorithm. *Electrochimica Acta*, 508:145175, 2024. https://doi.org/10.1016/j.electacta.2024.145175.

[20] W. Xia, Y. Yang, J.-H. Xue, and B. Wu. Tedigan: text-guided diverse face image generation and manipulation. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*:2256–2265, 2021.

[21] X. Xu, D. Sun, J. Pan, Y. Zhang, H. Pfister, and M.-H. Yang. Learning to super-resolve blurry face and text images. *Proceedings of the IEEE international conference on computer vision*:251–260, 2017.

[22] H. Zhang, J. Y. Koh, J. Baldridge, H. Lee, and Y. Yang. Cross-modal contrastive learning for text-to-image generation. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*:833–842, 2021.

[23] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas. Stackgan: text to photo-realistic image synthesis with stacked generative adversarial networks. *Proceedings of the IEEE international conference on computer vision*:5907–5915, 2017.