Faces à la Carte: Text-to-Face Generation via Attribute Disentanglement

Tianren Wang, Teng Zhang, Brian Lovell
The University of Queensland
St Lucia, Brisbane, Australia, Qld 4072
tianren.wang@uqconnect.edu.au, patrick.zhang@uq.edu.au, lovell@itee.uq.edu.au

Figure 1: Several examples of synthesised face images produced by our model. We select four groups of images which are arranged according to gender and age. The highlighted features in the text above are from the text annotations provided by the CelebA database [14]. In addition to good rendering accuracy of the specified features, the images show significant variation in terms of unspecified features.

Abstract

Text-to-Face (TTF) synthesis is a challenging task with great potential for diverse computer vision applications. Compared to Text-to-Image (TTI) synthesis tasks, the textual description of faces can be much more complicated and detailed due to the variety of facial attributes and the parsing of high dimensional abstract natural language. In this paper, we propose a Text-to-Face model that not only produces images in high resolution (1024×1024) with text-to-image consistency, but also outputs multiple diverse faces to cover a wide range of unspecified facial features in a natural way. By fine-tuning the multi-label classifier and image encoder, our model obtains the adjustment vectors and image embeddings which are used to transform the input noise vector sampled from the normal distribution. Afterwards, the transformed noise vector is fed into a pre-trained high-resolution image generator to produce a set of faces with the desired facial attributes. We refer to our model as TTF-HD. Experimental results show that TTF-HD generates high-quality synthesised faces from free-form text descriptions with state-of-the-art performance.

1. Introduction

With the advent of Generative Adversarial Networks (GAN) [5], image generation has made huge strides in terms of both image quality and diversity. However, the original GAN model [5] cannot generate images tailored to meet design specifications. To this end, many conditional GAN models have been proposed to fit different task scenarios [7, 25, 17, 22, 21, 19, 24]. Among these works, Text-to-Image (TTI) synthesis is an important, yet less studied, topic. TTI refers to generating a photo-realistic image which matches a given text description. As an inverse image captioning task, TTI aims to establish an interpretable mapping between image space and the text semantic space. TTI has huge potential and can be used in many applications including photo editing and computer-aided design. However, natural language contains high dimensional information which is often less specific but also much more abstract than images. Therefore, this research problem represents a considerable challenge.

Just like TTI synthesis, the sub-topic of Text-to-Face (TTF) synthesis has immense practical value in areas such as criminal investigation and also biometric research. For example, police often need professional artists to sketch likenesses of suspects based solely on the descriptions of eyewitnesses. This task is time-consuming, requires great skill and often results in inferior images. Many police may not have access to such professional artists. However, with a well-trained Text-to-Face model, we could quickly produce a wide variety of high-quality photo-realistic pictures based simply on the descriptions of eyewitnesses. Moreover, TTF can be used to address the emerging issues of data scarcity arising from the growing ethical concerns regarding informed consent for the use of faces scraped from the internet in modern biometrics research.

A major challenge of the TTF task is that the linkage between face images and their text descriptions are much looser than for, say, the bird and flower images commonly used in TTI research. A few sentences of description are hardly adequate to cover all the variations of human facial features. Also, for the same face image, different people may use quite different descriptions. This increases the challenge of finding satisfactory mappings between these descriptions and the facial features. Therefore, in addition to the aforementioned two criteria, a TTF model should have the ability to produce a group of images with high variation conditioned on the same text description. In a real-world application, a witness could choose one image among several possible images which they believe is closest the suspect. Image diversity is also extremely important for biometric researchers to obtain enough photos of rare ethnicities and demographics when synthesising large ethical face datasets that are exempt from the issue of informed consent.

We propose a GAN model which includes a novel TTF framework satisfying: 1) high image quality; 2) improved consistency of synthesised images and their text descriptions; and 3) the ability to generate a group of widely differing face images from the same text description.

More specifically, we propose a pre-trained BERT [3] multi-label model for natural language processing. This model outputs sparse text embeddings of length 40. We then fine-tune a pre-trained MobileNets [6] model using CelebA's [14] training data where the images have paired labels. Next, we predict labels from the input images. Then we structure a feature space with 40 orthogonal axes based on the noise vectors and the predicted labels. After this operation, the input noise vectors can be moved along specified adjustment directions to render output images which exhibit the desired features. Last, but certainly not least, we use the state-of-the-art image generator, StyleGAN2 [12], which maps the noise vectors into a feature disentangled latent space, to generate high-resolution images. As Fig. 1 shows, the synthesised images match the features of the description while exhibiting both good variation and excellent image quality.

1.1. Motivation and Ethical Considerations

There are emerging ethical concerns in the face research community including 1) scraping faces from the internet without user consent; 2) constructing face datasets with significant racial bias where some minorities are often neglected; and 3) offensive slurs from the meta-data associated with real people's faces being harvested from the internet. As a pioneering work targeting at resolving some of these ethical issues, our research starts by resolving the issue of violating user consent first and, to some extent, helps combat the racial bias problem by providing the ability to generated synthetic images of large numbers of desired ethnicities conditioned by text descriptions.

This work is a part of the Ethical Database of Interactive Training Heads (EDITH) project which has been reviewed by the Office of Research Ethics and is deemed to be outside the scope of ethics review under the National Statement on Ethical Conduct in Human Research and University policy due to the synthetic nature of the data.

1.2. Contributions

Our work has the following main contributions.

- Proposes a novel TTF-HD framework comprising a multi-label text classifier, an image label encoder, and a feature-disentangled image generator to generate high-quality faces with a wide range of variation.

- Adds a novel 40-label orthogonal coordinate system to guide the trajectory of the input noise vector.

- Uses state-of-the-art StyleGAN2 [12] as the generator to map the manipulated noise vectors into the disentangled feature space to generate 1024×1024 high-resolution images.

This paper is continued as follow. In Section 2, we review the important works in TTI, TTF, and models of the generators. In Section 3, we describe our proposed framework in detail. In Section 4, experimental results are presented both qualitatively and quantitatively and an ablation study is conducted to show the importance of the vector manipulating operations. In Section 5, we conclude our work by summarising our contributions and the limitations of our approach.

2. Related Works

2.1. Text-to-Image Synthesis

In the area of TTI, Reel *et al*. [17] first proposed to take advantage of GAN, which includes a text encoder and an image generator and they simply concatenated the text embedding to the noise vector as input. Unfortunately, this model failed to establish good mappings between the keywords and the corresponding image features. Moreover, due to the final results being directly generated from the concatenated vectors, the image quality was so poor that images were easily spotted as a fake. To address these two issues, StackGAN [22] proposed to generate images hierarchically by utilising two pairs of generators and discriminators. Later, Xu *et al*. proposed AttnGAN [21]. By introducing the attention mechanism, this model successfully matched keywords with the corresponding image features. Their interpolation experimental results indicated that the model could correctly render the image features according to the selected keywords. For example, by changing the description from "blue bird" to "red bird", the image features can be rendered accordingly. This model works remarkably well in translating bird and flower descriptions. However, in these cases, the descriptions are mostly just one sentence. If the descriptions are longer, the efficacy of text encoding deteriorates because the attention map becomes harder to train.

2.2. Text-to-Face Synthesis

Compared to the number of works in TTI, the published works in TTF are far fewer. The main reason is that a face description has a much weaker connection to facial features compared to that of, say, bird or flower images. Typically, the descriptions of birds and flowers are primarily about the colour of the feathers or petals. Descriptions of faces can be much more complicated with gender, age, ethnicity, pose, and other important facial attributes. Moreover, most of the TTI models are trained on Oxford-102 [16], CUB [20], and COCO [13] which are not face image datasets. When dealing with faces, the only face dataset that is suitable is Face2text [4] which has only five thousand pairs of samples — this not large enough to train a satisfactory model.

With all of the challenges mentioned above, there are still several inspiring works engaging in Text-to-Face synthesis. In a project named T2F [8], Akanimax proposed to encode the text descriptions into a summary vector using the LSTM network. ProGAN [10] was adopted as the generator of the model. Unfortunately, the final output images exhibited poor image quality. Later, the author improved his work, which he named T2F 2.0, by replacing ProGAN with MSG-GAN [9]. As a result, both image quality and image-text consistency improved considerably, but the output showed low variation with regard to facial appearance.

To address the data scarcity issue, O.R. Nasir *et al*. [15] proposed to utilise the labels of CelebA [14] to produce structured pseudo-text descriptions automatically. In this way, the samples in the dataset are paired with sentences which contain positive feature names separated by conjunctions and punctuation. The results are 64×64 pixel images showing a certain degree of variation in appearance. The best output image quality so far is from Chen *et al*. [2] which also adopted the model structure of AttnGAN [21]. Therefore, this work has the same issues with text encoding mentioned previously.

2.3. Feature-Disentangled Latent Space

Conventionally, the generator will produce random images from noise vectors sampled from a normal distribution. However, we desire to control the rendering of the images in response to the feature labels. To do this, Chen *et al*. [1] proposed to disentangle the desired features, by maximising the mutual information between the latent code $\boldsymbol{c}$ of the desired features and the noise vector $\boldsymbol{x}$. In his experiments, he introduced a variation distribution $Q(\boldsymbol{c}|\boldsymbol{x})$ to approach $P(\boldsymbol{c}|\boldsymbol{x})$. Finally, the latent code indicates that it has managed to learn interpretable information by changing the value in a certain dimension. However, the latent code in this work has only 3 or 4 dimensions; we require 40 features, which is much more complicated. Later, Karras *et al*. [11] established a novel style-based generator architecture, named StyleGAN, which does not take the noise vector as input like the previous works. The input vector is mapped into an intermediate latent space through a non-linear network before being fed into the generator network. The non-linear network consists of eight fully connected layers. A benefit for such a setting is that the latent space does not have to support sampling according to any fixed distribution [11]. In other words, we have more freedom to combine the desired features.

3. Proposed Method

Our proposed model, named TTF-HD, comprises a multi-label classifier $\boldsymbol{T}$, image encoder $\boldsymbol{E}$, and a genera-

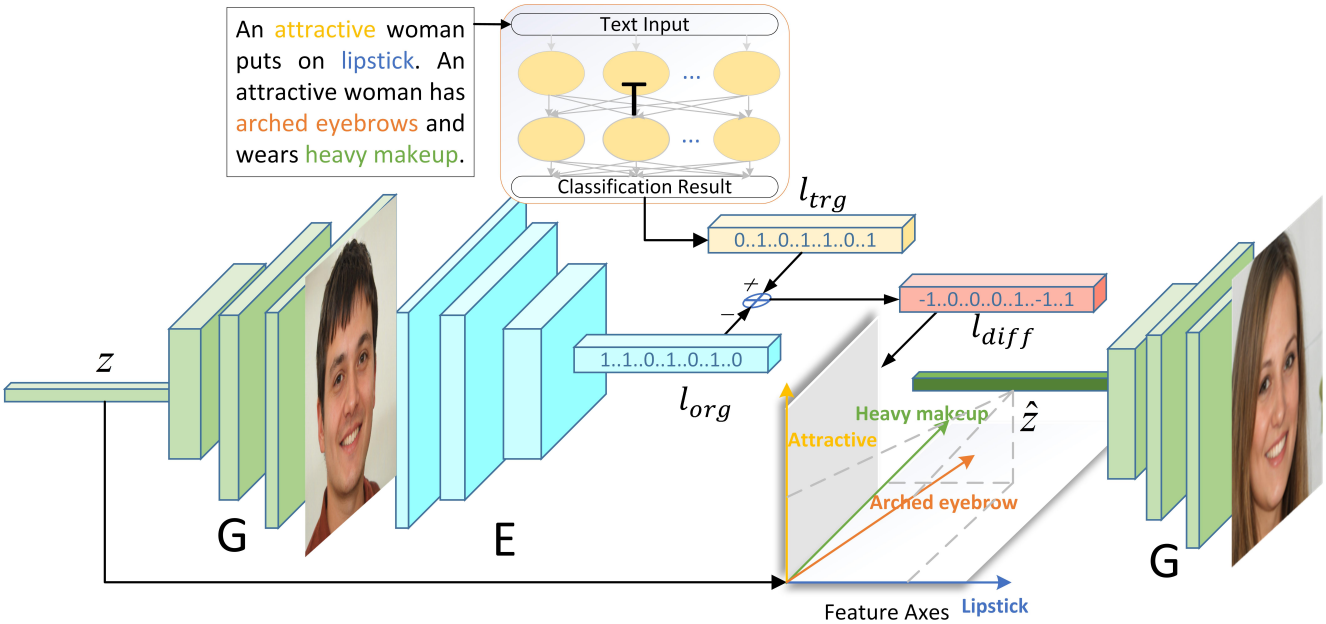

Figure 2: TTF-HD diagram. The text is fed into the multi-label classifier $\boldsymbol{T}$ which then outputs a text vector l_{trg} that represents 40 facial attributes. The image generator $\boldsymbol{G}$ firstly synthesises an image from a random noise vector $\boldsymbol{z}$. Then the image encoder $\boldsymbol{E}$ outputs the image embeddings l_{org}. The differentiated embedding l_{diff} is used to manipulate the original noise vector from $\boldsymbol{z}$ to $\hat{\boldsymbol{z}}$. Finally, the generator synthesises an image with the desired features from $\hat{\boldsymbol{z}}$.

tor $\boldsymbol{G}$ is shown in Fig. 2. Details will be discussed in the following subsections.

3.1. Multi-Label Text Classification

To conduct the TTF task, it is of vital importance to have sufficient facial attribute labels to fully describe a face. We propose to use the CelebA [14] dataset which includes 40 facial attribute labels for each face. To map the free-form natural language descriptions to the 40 facial attributes, we propose to fine-tune a multi-label text classifier $\boldsymbol{T}$ to obtain text embeddings of length 40. Note that the keywords in the descriptions are the same words or synonyms of text labels in the CelebA dataset. Some labels might be considered offensive to some people, but we need to use these labels so we can compare our approach to the work of others — we truly do not wish to cause offence even to synthesized people.

With these considerations, we adopt the state-of-the-art natural language processing model, Bidirectional Transformer (BERT) [3]. In light of the fact that this is a 40-class classification task, we choose to use the large network of the BERT model as it has better performance on high-dimensional training data. Some features have different names for their opposites attributes. For example, when training the model $\boldsymbol{T}$, the feature "age" could be represented by either "young" or "old". However, there is no "Age" label in the CelebA dataset [14] but only "Young". Therefore, in practice, "young" might be represented by a value close to 1 and "old" might be close to 0 in the real classification results. If a feature is not specified, it is set to 0. This process is shown in Fig. 3. Finally, the classifier outputs a text vector of length 40 for each description.

Figure 3: A possible classification result of the text classifier $\boldsymbol{T}$.

Note that one advantage of our text classifier compared to the earlier text encoders is that there is no restriction on the length of text descriptions. In previous works, the text is mostly crammed into one or two sentences. For face descriptions, the length is generally much longer than for bird and flower descriptions, which makes traditional text encoders inappropriate.

3.2. Image Multi-Label Embeddings

In the proposed framework, an image encoder $\boldsymbol{E}$ is required to predict the feature labels of the generated images. To do this, we fine-tune a MobileNet model [6] with the samples of CelebA [14]. The reason for choosing MobileNet is that it is a light-weight network model that has

a good trade-off between accuracy and speed. With this model, we can obtain the image embeddings which have the same length as the text vectors of the images generated from the noise vectors.

3.3. Feature Axes

After training the image encoder, now we can find the relationship between the noise vectors and the predicted feature labels by logistic regression. The length of the noise vectors is 512 ($\boldsymbol{x} \in \mathbb{R}^{512}$) and the length of thefeature vectors is 40 ($\boldsymbol{y} \in \mathbb{R}^{40}$). Therefore, we obtain:

$$\boldsymbol{y} = \boldsymbol{x} \cdot \boldsymbol{B} \tag{1}$$

where $\boldsymbol{B}$ is the matrix of dimension 512×40 to be solved.

This matrix needs to be orthogonalised because we must disentangle all the attributes so that the noise vectors can move along a certain feature axis without affecting the others. By the Gram-Schmidt process, the projection operator is:

$$\mathrm{proj}_{\mathbf{u}}(\mathbf{v}) = \frac{\langle \mathbf{v}, \mathbf{u} \rangle}{\langle \mathbf{v}, \mathbf{v} \rangle} \mathbf{u} \tag{2}$$

where $\mathbf{v}$ is the axis to be orthogonalised and $\mathbf{u}$ is the reference axis. Then, we obtain:

$$\begin{aligned} \mathbf{u}_k &= \mathbf{v}_k - \sum_{j=1}^{k-1} \mathrm{proj}_{\mathbf{u}_j}(\mathbf{v}_k), \\ \mathbf{w}_k &= \frac{\mathbf{u}_k}{\|\mathbf{u}_k\|}, (k = 1, 2, ...40)\,. \end{aligned} \tag{3}$$

In (3), the matrix $\boldsymbol{W} = [\mathbf{w}_1, \mathbf{w}_2, ...\mathbf{w}_k]$ is normalised so that $\boldsymbol{W}$ is unitary.

After these steps, we obtain the feature axes which are used to guide the update direction of the input noise vectors to obtain the desired features in the output images.

3.4. Noise Vector Manipulation

Manipulating the noise vectors is vital to our work because this determines whether the output images will have the described features as the text corpus. In the model diagram Fig. 2, this is the process of changing the random noise vector from $\boldsymbol{z}$ to $\hat{\boldsymbol{z}}$ by (4) where l is a column vector which determines the direction and magnitude of the movement along the feature axes.

$$\hat{\boldsymbol{z}} = \boldsymbol{z} + \boldsymbol{W} \cdot l \tag{4}$$

To ensure that the model will produce an image with the desired features no matter where the noise vectors are located in the latent space, we introduce four operations.

Differentiation. As shown in Fig. 2, the text classifier embedding output is denoted l_{trg} and the predicted embedding from the initial random vector is given by $l_{org} = \boldsymbol{E}(\boldsymbol{G}(\boldsymbol{z}))$. Intuitively, we can use l_{trg} to guide the movement of noise vectors along the feature axes. However, the value range of l_{trg} is $[0, 1]$. This means that the model cannot render features in the opposite direction, say, young versus old, because there are no labels corresponding to the opposite values. To solve this, we use differentiated embeddings l_{diff} to guide the feature editing obtained by (5)

$$l_{diff} = l_{trg} - l_{org}. \tag{5}$$

In this way, the noise vectors can be moved in both positive and negative directions along the feature axes because the value range of the differentiated embeddings is $[-1, 1]$. For the features which have a similar probability value in both the text embeddings and the image embeddings, their probability value is cancelled out and they will not be rendered repeatedly in the output images. This operation is shown in Fig. 2. For each feature, according to its probability level in l_{trg} and l_{org}, the movement direction can be positive, negative or neutral.

Note that to minimize interference of the unspecified features in the text descriptions, we do not apply the differentiation operation to such features. Instead, we keep their value as zero in the differentiated embeddings.

Nonlinear Reweighting: In the differentiated embeddings, the labels with values approaching -1 or 1 are the specified features where the text descriptions are specified in either a positive or negative way. Apart from these labels, there may be some other labels whose values are between -1 and 1 which tend to interfere with the desired feature rendering. Therefore, we need to emphasize the specified features. To do this, we scale the differentiated embeddings range slightly from $[-1, 1]$ to $\left[-\frac{\pi}{3}, \frac{\pi}{3}\right]$. Then we compute the $\tan(.)$ of the mapped differentiated embeddings. As a result, values approaching the ends of the range will get a higher weighting. In our case, since $\tan(pi/3) = \sqrt{3}$, the reweighed value range is now $\left[-\sqrt{3}, \sqrt{3}\right]$.

Normalization: As the noise vectors are sampled from a normal distribution, they have a higher probability to be sampled near the origin where the probability density is high. However, the more steps we move the vectors along different feature axes, the larger the distance becomes between these vectors and the origin, which will lead to more artefacts in the generated images. That is why we need to renormalise the vectors after each movement along the axes. This distance can be denoted as L_1 distance. Therefore, for the noise vector $\boldsymbol{X} = [\mathbf{x}_1, \mathbf{x}_2, ...\mathbf{x}_n]$, we get $\boldsymbol{X}' = [\mathbf{x}'_1, \mathbf{x}'_2, ..., \mathbf{x}'_n]$ with (6)

$$\begin{aligned} \|\mathbf{x}\|_1 &= \sum_{i=1}^{N=512} |\mathbf{x}_i| \\ \mathbf{x}'_i &= \frac{\mathbf{x}_i}{\|\mathbf{x}\|_1} (i = 1, 2, ..., 512) \end{aligned} \tag{6}$$

Figure 4: Images produced with single-sentence input. With fewer specified labels in the text, the model generates samples with higher variation.

Feature lock: To make the face morphing process more stable, we have a feature lock step each time we move the vectors along a certain axis. In other words, the model only uses the axes along which the vectors have been moved as the basis axes to disentangle the following feature axis. While for other axes of unspecified attributes in the textual descriptions, the movement direction and step size along such axes are not fixed to ensure diversity in the generated images. In this way, noise vectors are locked only in terms of the features mentioned in the descriptions.

3.5. High Resolution Generator

The generator $\boldsymbol{G}$ we use is a pre-trained model of StyleGAN2 [12]. On the basis of mapping the noise vectors which are sampled from the normal distribution to the intermediate latent space, StyleGAN2 improves the small artefacts by revisiting the structure of the network. With this generator, not only can the model synthesise high-resolution images, but it can also render the desired features from the manipulated input vectors.

4. Experiments and Evaluation

Dataset: The dataset we use is CelebA [14] which contains over 200k face images. For each sample, there is a paired one-shot label vector whose length is 40. In addition, there is another paired text description corpus set in which every description has up to 10 sentences. There may be some redundant sentences in some of them, but every description includes all the features the paired label vector indicates. We use this dataset to fine-tune the pre-trained multi-label text classifier and the pre-trained image encoder.

Experimental setting: In our evaluation experiments, we randomly choose 100 text descriptions. With each of them, the model will randomly generate 10 images. Therefore, the test set has 1000 images in total. As the experiments show, there will be significant image morphing when the noise vector moves twice along certain feature disentangled axis. Thus, we set the step size as 1.2, which multiplies the reweighted output of the differentiated vector. This guarantees a final weight which is used to move along the axis of around 2 ($\sqrt{3} \times 1.2$).

4.1. Qualitative Evaluation

Image quality: Fig. 1 also shows the paired descriptions in each group. We can see that most of the generated images are correctly rendered with the specified features.

Image diversity. To show the proposed method has great feature generalisation capacity, we conduct the image synthesis conditioned on the single-sentence description. In other words, apart from the key features that the sentences describe, the model should diversify the other facial features in the output. As Fig. 4 shows, for each single-sentence description, the proposed model produces images with high diversity.

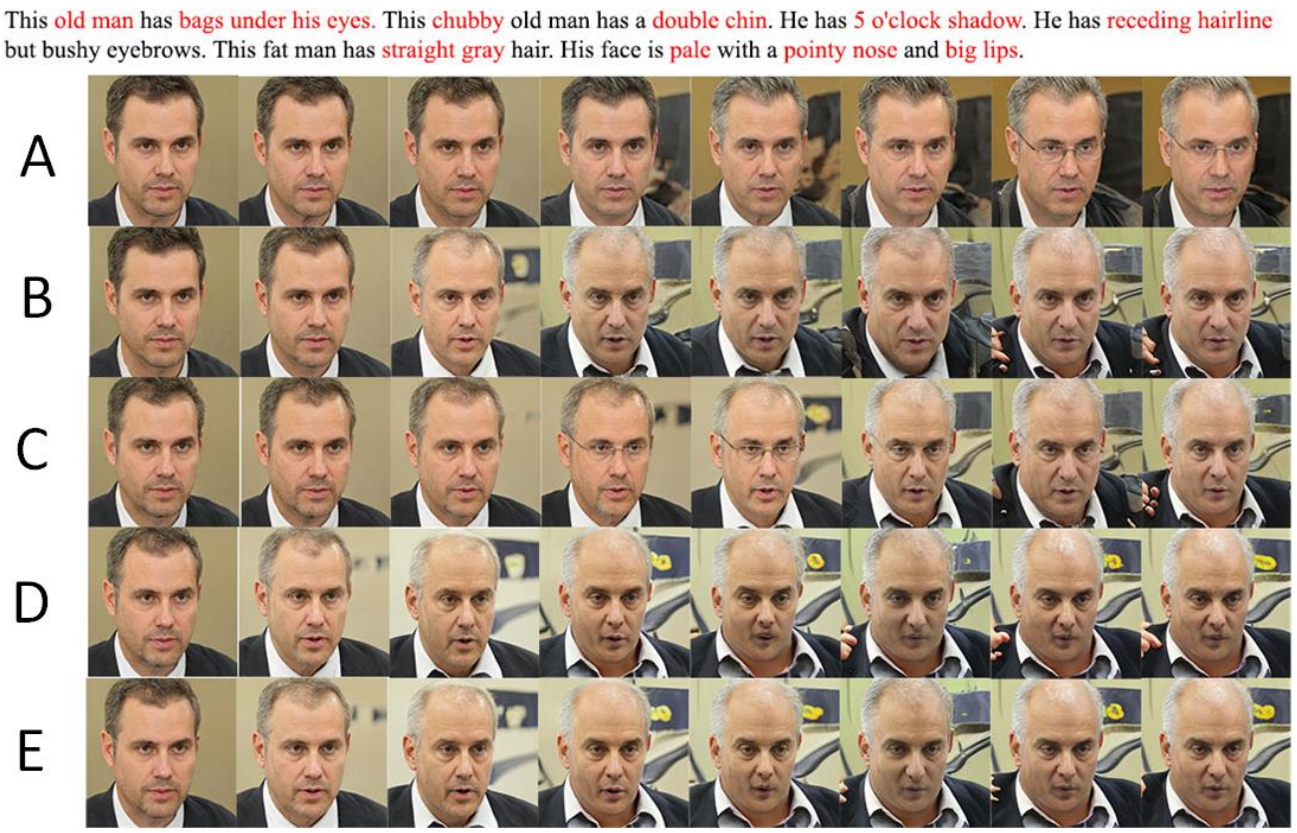

Figure 5: Image morphing via GAN of each group in the ablation study. (A) group with all operations applied (the default for TTF-HD); (B) group with reweighting, differentiation, and normalisation operations; (C) group with reweighting, differentiation operations; (D) group with the reweighting operation; and (E) group with no operations applied. We fix the noise vector input of each group. The figure shows the GAN morphing process from the randomly generated image on the left column to the final output on the right column.

4.2. Quantitative Evaluation

In this section, we apply three metrics to evaluate the above three criteria respectively. They are Inception Score (IS) [18] which is used in many previous works, Learned Perceptual Image Patch Similarity (LPIPS) [23] which is for evaluating the diversity of the generated images, and Cosine Similarity which is widely used to evaluate the similarity of two chunks of a corpus in natural language processing. Due to the lack of the source code for most of the works in the TTF area such as T2F 2.0 [8], we compare our experimental results with the TTF implementation of AttnGAN [21] which has produced the best results so far.

Table 1 shows the evaluation results of different models. We see the proposed TTF-HD method outperforms the state-of-the-art method AttnGAN [21] in terms of both image quality and Text-to-Image similarity.

Table 1: Evaluation results of different models

Methods	***IS***	***CS****	***LPIPS***
TTF-HD (ours)	**1.117±0.127**	**0.664**	0.583±0.002
AttnGAN	1.062±0.051	0.511	——

*Maximum for each group.

4.3. Ablation Study

In Section 3, we propose four operations to manipulate the noise vector to get the desired features. In this subsection, we conduct the ablation study and discuss the effects of the different operations applied.

To conduct the ablation study, we have 5 experiment settings. We choose one face description and produce 100 random images under each experimental setting respectively. Then, we use the above three metrics to evaluate the effect

of different operations.

Fig. 5 shows the GAN morphing process of the generated images. We can see that with all the proposed four manipulating operations, Group A can obtain an output with all desired features. While for other groups, the final images all suffer from artefacts on the rendering of the face and the background. This is because, with too many feature axis adjustment steps, the noise vector has been moved to a low-density region of the latent space distribution, which leads to the mode collapse problem.

Table 2: Ablation study evaluation results

Exp. Settings	**Evaluation Metrics**		
	IS	***CS****	***LPIPS***
Group A	1.122±0.043	0.754	**0.634±0.005**
Group B	1.116±0.080	0.739	0.608±0.005
Group C	**1.187±0.062**	**0.762**	0.603±0.005
Group D	1.101±0.095	0.683	0.521±0.006
Group E	1.102±0.033	0.706	0.532±0.005

*Maximum for each group

Table 2 shows the quantitative evaluation metrics on different groups of TTF-HD. We can see that Group A has the best diversity score as well as the second-best performance in terms of both IS and CS score. This suggests that applying all of the proposed operations leads to a good trade-off between image quality, text-to-face similarity and diversity.

5. Conclusion

In this paper, we set three main goals in the text-to-face image synthesis task: 1) high image resolution; 2) good text-to-image consistency; and 3) high image diversity. To this end, we propose a model, named TTF-HD, comprising a multi-label text classifier, an image encoder, a high-resolution image generator, and feature-disentangled axes. From both qualitative and quantitative evaluative comparisons, we see that the generated images exhibit good image quality, text-to-image similarity, and image diversity.

However, the model is still not entirely robust. There are always some images in the batch that are far more consistent with the text descriptions. This is possibly caused by insufficient accuracy of the text classifier and image encoder due simply to the lack of training data. In addition, features in the latent space are still not well disentangled, so that when you are moving the noise vector along one feature axis, other features which are highly correlated with it may also change. These issues must be addressed in future research.

References

[1] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pages 2172–2180, 2016.

[2] Xiang Chen, Lingbo Qing, Xiaohai He, Xiaodong Luo, and Yining Xu. Ftgan: A fully-trained generative adversarial networks for text to face generation. *arXiv preprint arXiv:1904.05729*, 2019.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[4] Albert Gatt, Marc Tanti, Adrian Muscat, Patrizia Paggio, Reuben A Farrugia, Claudia Borg, Kenneth P Camilleri, Mike Rosner, and Lonneke Van der Plas. Face2text: Collecting an annotated image description corpus for the generation of rich face descriptions. *arXiv preprint arXiv:1803.03827*, 2018.

[5] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[6] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[7] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[8] A. Karnewar. blog: https://medium.com/@animeshsk3/t2f-text-to-face-generation-using-deep-learning-b3b6ba5a5a93. 2018.

[9] Animesh Karnewar and Raghu Sesha Iyengar. Msg-gan: Multi-scale gradients gan for more stable and synchronized multi-scale image synthesis. *arXiv preprint arXiv:1903.06048*, 2019.

[10] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[11] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[12] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020.

[13] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In

European conference on computer vision, pages 740–755. Springer, 2014.

[14] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015.

[15] Osaid Rehman Nasir, Shailesh Kumar Jha, Manraj Singh Grover, Yi Yu, Ajit Kumar, and Rajiv Ratn Shah. Text2facegan: Face generation from fine grained textual descriptions. In *2019 IEEE Fifth International Conference on Multimedia Big Data (BigMM)*, pages 58–67. IEEE, 2019.

[16] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE, 2008.

[17] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. *arXiv preprint arXiv:1605.05396*, 2016.

[18] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in neural information processing systems*, pages 2234–2242, 2016.

[19] Tianren Wang, Teng Zhang, Liangchen Liu, Arnold Wiliem, and Brian Lovell. Cannygan: Edge-preserving image translation with disentangled features. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 514–518. IEEE, 2019.

[20] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. Caltech-ucsd birds 200. 2010.

[21] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1316–1324, 2018.

[22] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 5907–5915, 2017.

[23] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[24] Teng Zhang, Arnold Wiliem, Siqi Yang, and Brian Lovell. Tv-gan: Generative adversarial network based thermal to visible face recognition. In *2018 international conference on biometrics (ICB)*, pages 174–181. IEEE, 2018.

[25] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.