MF-GAN: Multi-conditional Fusion Generative Adversarial Network for Text-to-Image Synthesis

Yuyan Yang[1,2], Xin Ni[1,2], Yanbin Hao[1,2], Chenyu Liu[3], Wenshan Wang[3], Yifeng Liu[3], and Haiyong Xie[2,4(✉)]

[1] University of Science and Technology of China, Hefei 230026, Anhui, China
{sz987456,nx150012}@mail.ustc.edu.cn
[2] Key Laboratory of Cyberculture Content Cognition and Detection, Ministry of Culture and Tourism, Hefei 230026, Anhui, China
haiyong.xie@ieee.org
[3] National Engineering Laboratory for Risk Perception and Prevention (NEL-RPP), Beijing 100041, China
2011010090@bupt.cn, ww2468@columbia.edu, liuyifeng2@cetc.com.cn
[4] Advanced Innovation Center for Human Brain Protection, Capital Medical University, Beijing 100069, China

Abstract. The performance of text-to-image synthesis has been significantly boosted accompanied by the development of generative adversarial network (GAN) techniques. The current GAN-based methods for text-to-image generation mainly adopt multiple generator-discriminator pairs to explore the coarse/fine-grained textual content (*e.g.*, words and sentences); however, they only consider the semantic consistency between the text-image pair. One drawback of such a multi-stream structure is that it results in many heavyweight models. In comparison, the single-stream counterpart bears the weakness of insufficient use of texts. To alleviate the above problems, we propose a Multi-conditional Fusion GAN (MF-GAN) to reap the benefits of both the multi-stream and the single-stream methods. MF-GAN is a single-stream model but achieves the utilization of both coarse and fine-grained textual information with the use of conditional residual block and dual attention block. More specifically, the sentence and word features are repeatedly inputted into different model stages for textual information enhancement. Furthermore, we introduce a triple loss to close the visual gap between the synthesized image and its positive image and enlarge the gap to its negative image. To thoroughly verify our method, we conduct extensive experiments on two benchmarked CUB and COCO datasets. Experimental results show that the proposed MF-GAN outperforms the state-of-the-art methods.

Keywords: Text-to-Image · GAN · Triplet loss

1 Introduction

Text-to-image synthesis aims at generating high-resolution, photo-realistic and text-consistent images according to natural language descriptions, which is a

B. Þór Jónsson et al. (Eds.): MMM 2022, LNCS 13141, pp. 41–53, 2022.
https://doi.org/10.1007/978-3-030-98358-1_4

challenging task in computer vision and natural language processing. It drives research progress in multimodal learning and also has great potential applications, such as image editing, video games and computer-aided design.

Recently, many methods for text-to-image synthesis [6,13,17] are based on Attentional Generative Adversarial Network (AttnGAN) [16]. AttnGAN consists of two-stage generators, and generally encodes text descriptions into two kinds of vectors, namely, global sentence vector and local word vector. The first stage generator utilizes the sentence vector to generate low-resolution images, and the second stage generator generates high-resolution images based on the initial images and the spatial attention mechanism. More recently, many studies adopt a simple single-stream GAN for text-to-image generation (see, *e.g.*, [14,19,20]). In particular, HDGAN [20] resembles a simple vanilla GAN, which has multiple side outputs and uses complicated hierarchical-nested adversarial objectives for training. DF-GAN [14] fuses image features and sentence features through several deep fusion blocks and adopts a one-way discriminator, instead of a two-way discriminator, to speed up the convergence of the generator. DTGAN [19] employs the visual loss to ensure that the generated images and real images have similar color distribution and shape.

These methods have proven to be useful. The single-stream structure is more efficient than the stacked structure, as the former contains only one generator and one discriminator, while the latter (*e.g.*, AttnGAN) is more complex and consists of three generators, three generators, and a deep attentional multimodal similarity module. Thus the single-stream network is more preferable when it is necessary to decrease the run time and improve the stability of the generative model. However, it still has many shortcomings. Firstly, with the size of the feature map increasing, it becomes more and more difficult to fuse sentence vectors and image features through affine transformation only, and most fine-grained information at the word level may be lost during the generation process. Secondly, the one-way discriminator only pays attention to the matching information between texts and images, but ignores the fact that matching the generated images and real images can improve the quality of the generated images. Although visual loss can alleviate this problem, it neglects the information available from matching real and generated images.

To address these problems, we propose a novel Multi-conditional Fusion Generative Adversarial Network (MF-GAN), which has only one generator/discriminator pair without involving extra modules such as object detection. In the generator, we propose a Conditional Residual Block and a Dual Attention Block, respectively, to take advantage of sentence features and word features to model text-to-image mapping. In the discriminator, we first map the input image into the semantic space, then employ a triplet loss to pull the synthesized image towards its corresponding ground-truth image and push it away from another image that is associated with a different text description.

We summarize the contributions of our work as follows. Firstly, we propose a novel Multi-conditional Fusion Generative Adversarial Network (MF-GAN) for text-to-image generation, where sentence features and word features are applied for many times during image synthesis with only a single-stream GAN. Secondly, triplet loss is carried to improve the semantic consistency and image quality by

making the generated image close to its related image and far away from the irrelevant image in the semantic space. To the best of our knowledge, it is the first time to introduce the triplet loss in text-to-image synthesis. Lastly, we conduct extensive experiments to quantify the advantages of MF-GAN. The experimental results show that MF-GAN outperforms the state-of-the-art methods on two standard benchmark datasets.

The remainder of this paper is structured as follows. Section 2 presents related works. Section 3 describes the overall framework of MF-GAN. three important components. Section 4 evaluates MF-GAN using two popular datasets. Section 5 concludes the paper with future works.

2 Related Work

The most popular and efficient text-to-image synthesis methods are GAN based methods. The application of GAN was first proposed by Reed *et al.* [10] in 2014. It contains a generator and a discriminator, where the former generates interrelated images from texts, and the latter tries to distinguish generated images from real images until it reaches the Nash equilibrium. However, the images generated by this method have low resolution. To address this problem, StackGAN [18] adopts a tree-like structure to improve the image quality. AttnGAN [16] adopts an attention-driven, multi-stage GAN for the fine-grained text-to-image generation, obtaining very promising results. Encouraged by the success of AttnGAN, researchers further improve its performance. For example, SEGAN [13] and SDGAN [17] apply siamese network [8] to fulfill low-level semantic diversity. MirrorGAN [9] regenerates the corresponding text description based on the generated images. CPGAN [6] designs a memory construction [1] to learn the meaningful visual features from each relevant image for a given word by using Bottom-Up and Top-Down Attention model and Yolo-V3.

Apart from the methods which take the stacked structure as the backbone as mentioned above, there are many ways to convert the generation process into multiple steps, which may lead to better performance on complex datasets such MSCOCO [7]. More specifically, IGSG [3] builds scene graphs from text descriptions first, which reason about the objects and their relationships. Then it uses a graph convolutional network to generate scene layouts from the scene graphs. Finally, a low-resolution image is generated by a Cascade Refinement Network. InferrGAN [2] decomposes the text-to-image generation into three steps: generating bounding boxes for each object in the text description, generating object shapes based on the bounding boxes, and generating the image conditioned on them. Moreover, ObjGAN [5] proposes an object-driven attention mechanism to provide fine-grained information for different components.

However, the training process of the generation process is slow and inefficient. To simplify the model, HDGAN [20] presents an extensible single-stream generator architecture. DF-GAN [14] also presents a novel simplified text-to-image backbone and Matching-Aware zero-centered Gradient Penalty to achieve the desired results without extra networks. DTGAN [19] adopts the attention modules and conditional normalization to fine-tune on each scale of feature maps.

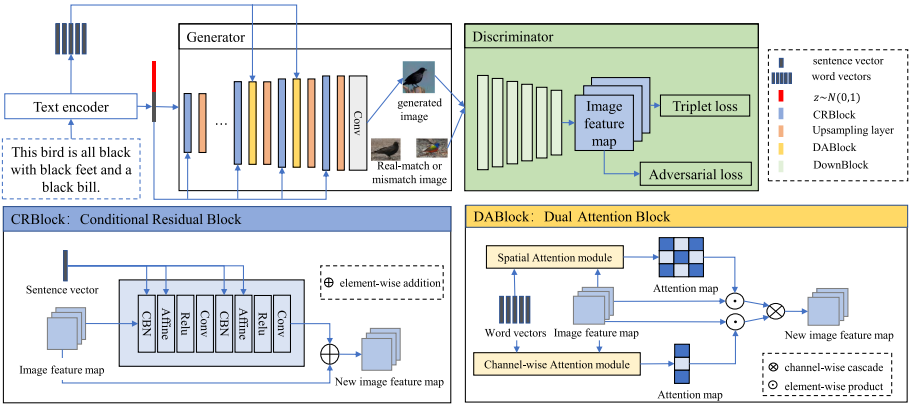

Fig. 1. The architecture of MF-GAN with only one generator/discriminator pair. The generator contains multiple CRBlocks (blue), DABlocks (yellow) and upsampling layers (orange). The discriminator contains multiple DownBlocks, which generate image features from the input images. Adversarial loss is calculated based on (text, generated image), (text, real-match image), and triplet loss is calculated based on (generated image, real-match image), (generated image, real-mismatch image). (Color figure online)

Different from the methods mentioned above, MF-GAN refines image features at the sentence level through conditional batch-normalization and affine transformation. Besides, the dual attention module is used only on the large scale of image features, which can focus on the word-level information. Additionally, inspired by [12], we also employ the triplet loss to generate more realistic and semantic consistent images.

3 Method

In this section, we present MF-GAN, a novel text-to-image adversarial generation network, aiming at refining image features at both sentence and word level and meanwhile improving semantic consistency by triplet loss.

3.1 Overall MF-GAN Architecture

MF-GAN is composed of three main components: a text encoder, a generator, and a discriminator, as illustrated in Fig. 1.

The text encoder aims at extracting the feature representations at both sentence level and word level from the natural language descriptions. We adopt a bi-directional Long Short-Term Memory (LSTM) pre-trained by [16] to learn the text representation. Specifically, it takes a sentence as input, and each word in the sentence corresponds to two hidden states in the bi-directional LSTM, one for each direction. These hidden states are utilized to represent the semantic

meaning of each word, and the last hidden states are concatenated to form the global sentence vector. They are denoted by $\omega \in \mathbb{R}^{D \times N}$ and $s \in \mathbb{R}^{D}$, where D is the dimension of the word vector and sentence vector, N is the number of words. In other words, ω is a feature matrix of all words and its i^{th} column ω_i is the feature vector of the i^{th} word in the given sentence.

The generator has three inputs: the sentence feature vector s, the word feature matrix ω from the text encoder, and the noise vector z sampled from the standard normal distribution. To make full use of all these informations for generating high-quality images, we apply m upsampling layers $(U_1, U_2, ..., U_m)$ to enlarge image features, m Conditional Residual Blocks (CRBlock) $(R_0, R_1, ..., R_m)$ to fuse sentence information and image features, and two Dual Attention Blocks (DABlock) (A_1, A_2) to complement more fine-grained word-level information. Specifically, we have

$$\begin{aligned} &\hat{s} = F^{ca}(s); \quad h_0 = R_0(fc(z, \hat{s}), s); \\ &h_i = R_i(U_i(h_{i-1}))\ for\ i = 1, 2, 3, ..., m-2; \\ &h_j = R_j(U_j(A_j(h_{j-1}, \omega), h_{j-1}))\ for\ j = m-1, m;\ k = 1, 2; \\ &x = G(h_m). \end{aligned} \tag{1}$$

where $(h_0, h_1, ..., h_m)$ are the hidden states generated by $(R_0, R_1, ..., R_m)$, which represents the image features with gradually growing resolutions and h_m is the final output with the highest resolution. F^{ca} is the Conditioning Augmentation [18] which converts the sentence vector s to the N_c-dimensional conditioning vector $\hat{s}$, which is concatenated to the N_z dimensional noise vector z sampled from the standard normal distribution. Then the result is fed into a fully connected layer and the first CRBlock R_0 to generate the 4×4 image feature h_0. CRBlock R_i and DABlock A_j are described in detail in Sect. 3.2 and 3.3, respectively.

The discriminator in our MF-GAN mainly contains several DownBlocks, each of which consists of several down-sampling layers and residual networks, converting input images into 4×4 image feature maps. Then the image features of the generated images, real-match images, and real-mismatch images are used to calculate the triplet loss (see Sect. 3.4) to improve semantic consistency. As a result, the discriminator needs to judge whether the corresponding image and text match according to the input image features and sentence features. In order to do so, we first compress the sentence vector s to N_d dimensions and spatially replicate it to form a $4 \times 4 \times N_d$ tensor. Then this tensor is concatenated with the image feature maps mentioned above, which is then fed into two convolutional layers to calculate the adversarial loss for the discriminator and generator:

$$\begin{aligned} L^D_{adv} = &\ \mathbb{E}_{x \sim P_{data}}[\max(0, 1 - D(x, s))] \\ &+ \frac{1}{2}\mathbb{E}_{G(z,x,\omega) \sim P_G}[\max(0, 1 + D(G(z, s, \omega), s))] \\ &+ \frac{1}{2}\mathbb{E}_{x \sim P_{misdata}}[\max(0, 1 + D(x, s))], \end{aligned} \tag{2}$$

$$L^G_{adv} = \mathbb{E}_{x \sim P_G}[D(G(z, s, \omega), s)], \tag{3}$$

where z is a noise sampled from the standard normal distribution, s is the sentence vector, ω is the word feature matrix, $P_{data}, P_G, P_{misdata}$ respectively denote the synthetic data distribution, real data distribution, and mismatching data distribution.

3.2 Conditional Residual Block

Our CRBlock aims at refining image features using text information as guidance. Specifically, we choose Conditional Batch Normalization (CBN) and Affine Transformation (Affine), which can take sentence vectors as conditions to predict the parameters of batch-normalization and linear transformation, respectively. As shown in Fig. 1, CRBlock receives two inputs: sentence vector and image feature map. Next we describe the CBN and Affine in detail.

CBN takes a batch of image features $x \in \mathbb{R}^{N\times C\times H\times W}$ and sentence vector s as input. It achieves the feature fusion as follows: First, it normalizes the mean and standard deviation for each feature channel; second, it learns the parameters γ_s and β_s from the conditions s; third, it learns a set of affine parameters $\gamma, \beta \in \mathbb{R}^C$ from data. The modified normalization function is formatted as

$$CBN(x|s) = (\gamma + \gamma_s) \times \frac{x - \mu(x)}{\sigma(x)} + (\beta + \beta_s), \tag{4}$$

where $\mu(x), \sigma(x) \in \mathbb{R}^C$ are the mean and standard deviation respectively. They are computed across the dimension of batch and spatial independently for each feature channel.

Affine is similar to CBN, which also receives two inputs: image features $x \in \mathbb{R}^{N\times C\times H\times W}$ and sentence vector s. It first predicts $\gamma_s, \beta_s \in \mathbb{R}^C$ from s via two one-hidden-layer MLPs. Then it fuses text information and image features, which can be formally expressed as follows:

$$AFF(x|s) = \gamma_s \times x + \beta_s. \tag{5}$$

3.3 Dual Attention Block

The purpose of our Dual Attention Block is to draw different sub-regions of the image condition on words that are most relevant to those sub-regions, and drawing the connection between words and channels. In implementation, we apply spatial attention and channel-wise attention to image features.

Spatial attention module takes the word feature $\omega \in \mathbb{R}^{D\times T}$ and the hidden image feature $h \in \mathbb{R}^{\hat{D}\times(H\times W)}$ as input. Note that ω_i is the D-dimensional feature vectors of the i^{th} word (a total of T words), and h_i is the $\hat{D}$-dimensional feature vectors of i^{th} sub-region of the image. We first map the word feature to the common semantic space of the image feature by a perception layer $U \in \mathbb{R}^{\hat{D}\times D}$, producing $\omega' = U\omega$. Then, we calculate the word-context vector for h_j, which represents the relevance between the word vectors and the j^{th} sub-region:

$$c_j = \sum_{i=0}^{T-1} \beta_{j,i}\omega'_i, \ where \ \beta_{j,i} = \frac{\exp(s'_{j,i})}{\sum_{k=0}^{T-1} \exp(s'_{j,k})}, \tag{6}$$

where $s'_{j,i} = h_j^T \omega'_i$, and $\beta_{j,i}$ means the weight of the i^{th} word when generating j^{th} sub-region of the image. Therefore, the output of spatial attention module is $(c_0, c_1, ..., c_{(H\times W)-1}) \in \mathbb{R}^{\hat{D}\times(H\times W)}$.

Channel-wise attention module has the same inputs as spatial attention module: the word features $\omega \in \mathbb{R}^{D\times T}$ and hidden image features $h \in \mathbb{R}^{\hat{D}\times(H\times W)}$. But it uses a different perception layer $V \in \mathbb{R}^{(H\times W)\times D}$ to convert the word feature into the common semantic space of the image features, producing $\omega'' = V\omega$. Then we apply similar method to calculate word-context vector c_j for the j^{th} channel of the image feature, which is a dynamic representation of word vectors relevant to channels:

$$c_j = \sum_{i=0}^{T-1} \alpha_{j,i}\omega''_i, \ where \ \alpha_{j,i} = \frac{\exp(r'_{j,i})}{\sum_{k=0}^{T-1} \exp(r'_{j,k})}, \tag{7}$$

where $r'_{j,i} = h_j^T \omega''_i$, and $\alpha_{j,i}$ represents correlation values between channels and words across all spatial locations. Hence, the output of channel-wise attention module is $(c_0, c_1, ..., c_{\hat{D}-1}) \in \mathbb{R}^{(H\times W)\times\hat{D}}$.

Then we compute element-wise products of these two output attention maps from the above two modules and the original image features respectively. Finally, a new feature map is obtained after a channel-wise cascade processing.

3.4 Triplet Loss

To enhance the generation consistency, we apply triplet loss working on our discriminator and focus on the hardest negatives in a mini-batch. In practice, given a pair of generated image and corresponding real image (I_g, I_r), we choose its hardest negative image in this batch of real images by $I' = argmax_{I\neq I_r} d(D(I_g), D(I))$, where D means generating image features by a set of downsampling blocks in the discriminator and d means calculating the differences between two features. Then with the predefined margin α, we adopt the triplet loss as following:

$$L_{triplet} = \max(d(D(I_g), D(I_r)) - d(D(I_g), D(I')) + \alpha, \ 0). \tag{8}$$

The generator loss consists of triplet loss and adversarial loss:

$$L^G = L^G_{adv} + \lambda_T L_{triplet}, \tag{9}$$

where λ_T is a hyper-parameter for triplet loss.

The discriminator loss contains adversarial loss and Matching-aware zero-center gradient penalty (MA-GP) loss [14], which enables us to synthesize more realistic images through a zero-centered gradient penalty on real data:

$$L^D = L^D_{adv} + \lambda_M \mathbb{E}_{x\sim P_{data}}[(||\nabla_x D(x,s)|| + ||\nabla_s D(x,s)||)^p], \tag{10}$$

where p and λ_M are the hyper-parameters to balance two kinds of loss.

4 Experiment

4.1 Datasets and Training Details

We evaluate our MF-GAN for text-to-image generation on two widely used datasets. The first dataset is the CUB-200-2011 bird dataset [15], which contains 200 bird species with 11788 images. We split them into 8,855 training images and 2933 test images. Each image is annotated with 10 descriptions, 15 part locations, 312 binary attributes and 1 bounding box. We pre-process the CUB dataset to ensure that the bounding boxes of birds have greater-than-0.75 object-image size rations suggested by [18]. The second dataset is the COCO dataset [7], which contains 82783 images for training and 40504 images for validation, and each image has 5 descriptions. The greater number and types of images make the COCO dataset more challenging than the CUB dataset.

As for training details, we set $D = 256, N_c = 128, N_z = 100, N_d = 256$ and $W_0 = H_0 = 256$ by default. Then we train MF-GAN for 800 epochs on CUB and 120 epochs on COCO dataset by using Pytorch. Besides, we use Adam with $\beta_1 = 0.0$ and $\beta_2 = 0.9$ to optimize our training process. The learning rate is set to 0.0001 for generator and 0.0004 for discriminator according to Two Time-scale Update Rule (TTUR) [14]. The text encoder and its parameters are all same as the previous works [16].

4.2 Evaluation Metrics

Similar to previous works (*e.g.*, [14]), we adopt two metrics, Inception Score(IS) [11] and Frechet Inception Distance (FID) [14], to evaluate the performance.

Inception Score (IS). The Inception score aims to measure two indicators of GAN: the quality and diversity of the synthesized images. It is formulated as:

$$I = \exp(\mathbb{E}_x D_{KL}(p(y|x)||p(y))), \tag{11}$$

where x is a synthesized image and y is the label predicted by a pre-trained Inception v3 model. IS computes the KL-divergence between the distribution of $p(x|y)$ and $p(y)$. A higher IS score means that each generated image clearly belongs to a specific class and the labels are evenly distributed. But when the mode collapses, this result will have no reference value.

Frechet Inception Distance (FID). The FID has the same function as IS, but the difference is that it calculates the Frechet distance between the distribution of the real image r and the generated image x in the feature space of a pre-trained Inception v3 network. The FID is formulated as:

$$F(r, x) = ||\mu_r - \mu_x||_2^2 + Tr(\Sigma_r + \Sigma_x - 2\sqrt{\Sigma_r \Sigma_x}), \tag{12}$$

where $\mu_r, \mu_x, \Sigma_r, \Sigma_x$ are respective means and covariance of real data distribution and generated data distribution. Lower FID means that two distributions

Table 1. The IS and FID scores on CUB and COCO datasets.

Method	CUB		COCO	
	IS↑	FID ↓	IS↑	FID↓
AttnGAN [16]	4.36 ± 0.03	–	25.89 ± 0.47	35.49
ControlGAN [4]	4.58 ± 0.09	–	24.06 ± 0.60	–
SD-GAN [17]	4.67 ± 0.09	–	$\mathbf{35.69 \pm 0.50}$	–
SE-GAN [13]	4.67 ± 0.04	18.167	27.86 ± 0.31	32.28
DF-GAN [14]	4.86 ± 0.04	16.09	–	28.92
DTGAN [19]	4.88 ± 0.03	16.35	–	**23.61**
MF-GAN	$\mathbf{4.94 \pm 0.07}$	**15.52**	28.70 ± 0.22	27.95

are closer, the quality of synthesized images is higher and the diversity is better. Moreover, the FID is more sensitive to the model collapse, because only one category of images will cause really high FID score.

4.3 Quantitative Results

We compare MF-GANwith numerous text-to-image synthesis methods, including the classic method (*i.e.*, AttnGAN [16]), two AttnGAN based improved methods (*i.e.*, SD-GAN [17] and SE-GAN [13]), DF-GAN [14], controlGAN [4], and DTGAN [19]. Note that we do not choose CPGAN as it does not experiment on the CUB dataset and needs two extra pre-trained: Bottom-Up and Top-Down (BUTD) Attention model and Yolo-V3. The test results of all these approaches on CUB and COCO dataset are from their corresponding published results.

Table 1 shows the IS and FID scores on the CUB and COCO dataset. We make the following observations. First, MF-GAN outperforms SE-GAN which employs sliding loss to enrich image ID information in image synthesis by achieving inception scores of 4.94 and FID scores of 15.52. Second, MF-GAN improves the IS from 4.86 to 4.94 and reduce the FID from 16.09 to 15.52 compared to DF-GAN which also has a single-stream structure. These quantitative results on CUB dataset show that our MF-GAN generates images that have higher quality and diversity than other models. Third, Compared with AttnGAN, the FID value of our MF-GAN is greatly reduced. Moreover, our model decreases the FID from 28.92 to 27.95 compared with DF-GAN. It demonstrates that our DF-GAN outperforms the art-of-the-state methods on the COCO dataset.

4.4 Qualitative Results

We now compare the images generated by AttnGAN, DF-GAN, and MF-GAN. Figure 2 shows the qualitative results on the CUB dataset (the first four columns) and the COCO dataset (the last four columns). Note that the first row shows 8 texts extracted from the test set of CUB and COCO, and the following two rows show the sample images generated by AttnGAN, DF-GAN, and our MF-GAN, respectively, from the corresponding text.

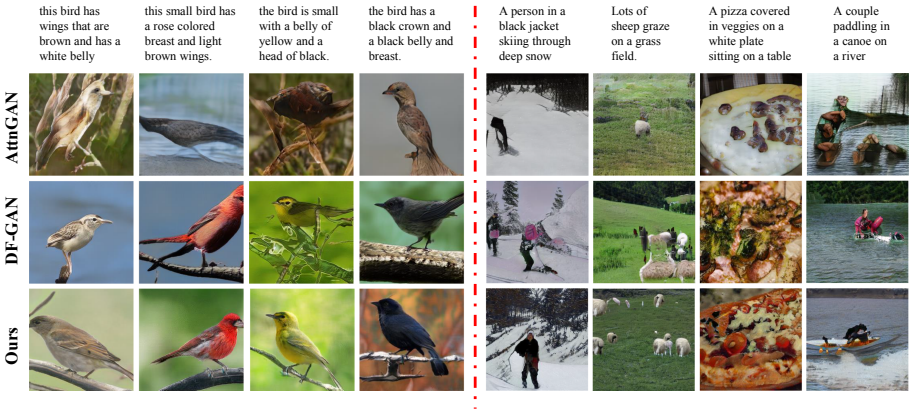

Fig. 2. Sample images synthesized by AttnGAN, DF-GAN and MF-GAN conditioned on text descriptions from CUB (1–4 columns) and COCO (5–8 columns) datasets.

AttnGAN uses a stacked network structure to generate low-resolution images and add more details to it. In this way, once the initial generated images is completely distorted, it is difficult to further improve by the subsequent generators. In comparison, both DF-GAN and our MF-GAN adopt a single network structure, and the generated images are more realistic. For example, the images generated by AttnGAN in columns 1 to 3 lack the shape of the bird, and their color information does not match the corresponding text; however, the images generated by DF-GAN and our MF-GAN have the key characteristics of a bird.

In addition, when compared against DF-GAN, MF-GAN generates more details in both the background and the target object. As shown in columns 1 to 3, the birds in the last row are more complete than the previous row. Obviously, the COCO dataset is more challenging than the CUB bird dataset. It is difficult for all methods to generate complete and realistic images for all target objects in the text. However, the images generated by AttnGAN are distorted with greater probability (*e.g.*, images generated by AttnGAN in columns 5 to 8 can no longer find the approximate shapes of the person, sheep, pizza, and canoe). DF-GAN has made great progress on this basis, the generated images are more realistic and contain more objects, while our MF-GAN generates even more details than DF-GAN, and the overall details are richer (*e.g.*, in the 5th and 8th columns, the person and canoe generated by our model are more realistic).

4.5 Ablation Study

We next perform ablation experiments on the CUB dataset to verify the effectiveness of each component in our MF-GAN, which contains Conditional Residual Network (CRBlock), Dual Attention Block (DABlock) and triplet loss (Tloss) working on the discriminator. We first remove triplet loss from our MF-GAN, and then remove CRBlock by retaining affine transformation and removing

Table 2. Quantitative results of the models that remove the CRBlock, DABlock, Triplet loss (Tloss) from MF-GAN and replace with visual loss (Vloss) on CUB dataset.

Method	IS↑
MF-GAN without (DAB + CRB + Tloss)	4.31 ± 0.05
MF-GAN without (CRB + Tloss)	4.51 ± 0.03
MF-GAN without Tloss	4.70 ± 0.03
MF-GAN replace Tloss with Vloss	4.52 ± 0.04
MF-GAN	$\mathbf{4.94 \pm 0.04}$

conditional batch-normalization. Finally, we continue to remove DABlock. We test their performance on the CUB dataset, and the results are shown in Table 2.

After removing the triplet loss, the IS score decreases from 4.94 to 4.70, suggesting that the triplet loss is able to improve image quality and semantic consistency. Then we continue to remove our CRBlock, and the IS score further drops from 4.70 to 4.51, suggesting that the CRBlock is more effective than affine transformation only in the text-to-image generation task. Finally, we remove the DABlock, the IS score drops to 4.31, which shows that DABlock can indeed improve the quality of the generated images.

We also compare our triplet loss with visual loss. Note that we employ triplet loss to improve the quality of the generated images and the semantic consistency between texts and images by matching image ID information; however, visual loss proposed by [19] has the similar function, which computes the L1 loss based on the image features of real image and the generated image. In experiments, we keep the backbone of our model and hyper-parameters λ_T, but replace triplet loss with visual loss. The results are shown in Table 2, suggesting that the triplet loss is more efficient than the visual loss.

5 Conclusion

In this paper, we propose a novel and simple text-to-image synthesized method, Multi-conditional Fusion Generative Adversarial Network (MF-GAN), to model the image feature maps at both sentence and word level. In addition, we introduce the triplet loss to improve image quality and semantic consistency. Extensive experiments demonstrate that our MF-GAN outperforms the state-of-the-art methods on the CUB dataset and COCO dataset. Results of ablation study show the effectiveness of each module in MF-GAN on improving the image quality and semantic consistency. For future works, we plan to investigate object detection and semantic alignment for further improving semantic consistency.

Acknowledgments. We would like to thank the anonymous reviewers for their valuable suggestions. Haiyong Xie is the correspondence author. This work is supported in part by the National Key R&D Project (Grant No. SQ2021YFC3300088) and the Natural Science Foundation of China (Grant No. U19B2036).

References

1. Gulcehre, C., Chandar, S., Cho, K., Bengio, Y.: Dynamic neural turing machine with continuous and discrete addressing schemes. Neural Comput. **30**(4), 857–884 (2018)
2. Hong, S., Yang, D., Choi, J., Lee, H.: Inferring semantic layout for hierarchical text-to-image synthesis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7986–7994 (2018)
3. Johnson, J., Gupta, A., Fei-Fei, L.: Image generation from scene graphs. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1219–1228 (2018)
4. Li, B., Qi, X., Lukasiewicz, T., Torr, P.H.: Controllable text-to-image generation. arXiv preprint arXiv:1909.07083 (2019)
5. Li, W., et al.: Object-driven text-to-image synthesis via adversarial training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12174–12182 (2019)
6. Liang, J., Pei, W., Lu, F.: CPGAN: content-parsing generative adversarial networks for text-to-image synthesis. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12349, pp. 491–508. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58548-8_29
7. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
8. Melekhov, I., Kannala, J., Rahtu, E.: Siamese network features for image matching. In: 2016 23rd International Conference on Pattern Recognition (ICPR), pp. 378–383. IEEE (2016)
9. Qiao, T., Zhang, J., Xu, D., Tao, D.: Mirrorgan: learning text-to-image generation by redescription. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1505–1514 (2019)
10. Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., Lee, H.: Generative adversarial text to image synthesis. In: International Conference on Machine Learning, pp. 1060–1069. PMLR (2016)
11. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. arXiv preprint arXiv:1606.03498 (2016)
12. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: a unified embedding for face recognition and clustering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 815–823 (2015)
13. Tan, H., Liu, X., Li, X., Zhang, Y., Yin, B.: Semantics-enhanced adversarial nets for text-to-image synthesis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10501–10510 (2019)
14. Tao, M., Tang, H., Wu, S., Sebe, N., Wu, F., Jing, X.Y.: DF-GAN: deep fusion generative adversarial networks for text-to-image synthesis. arXiv preprint arXiv:2008.05865 (2020)
15. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The Caltech-UCSD Birds-200-2011 Dataset. Technocal Report CNS-TR-2011-001, California Institute of Technology (2011)
16. Xu, T., et al.: Attngan: fine-grained text to image generation with attentional generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1316–1324 (2018)

17. Yin, G., Liu, B., Sheng, L., Yu, N., Wang, X., Shao, J.: Semantics disentangling for text-to-image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2327–2336 (2019)
18. Zhang, H., et al.: Stackgan: text to photo-realistic image synthesis with stacked generative adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 5907–5915 (2017)
19. Zhang, Z., Schomaker, L.: DTGAN: dual attention generative adversarial networks for text-to-image generation. arXiv preprint arXiv:2011.02709 (2020)
20. Zhang, Z., Xie, Y., Yang, L.: Photographic text-to-image synthesis with a hierarchically-nested adversarial network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6199–6208 (2018)