DTGAN: Dual Attention Generative Adversarial Networks for Text-to-Image Generation

Zhenxing Zhang and Lambert Schomaker
University of Groningen
{z.zhang,l.r.b.schomaker}@rug.nl

Abstract

Most existing text-to-image generation methods adopt a multi-stage modular architecture which has three significant problems: 1) Training multiple networks increases the run time and affects the convergence and stability of the generative model; 2) These approaches ignore the quality of early-stage generator images; 3) Many discriminators need to be trained. To this end, we propose the Dual Attention Generative Adversarial Network (DTGAN) which can synthesize high-quality and semantically consistent images only employing a single generator/discriminator pair. The proposed model introduces channel-aware and pixel-aware attention modules that can guide the generator to focus on text-relevant channels and pixels based on the global sentence vector and to fine-tune original feature maps using attention weights. Also, Conditional Adaptive Instance-Layer Normalization (CAdaILN) is presented to help our attention modules flexibly control the amount of change in shape and texture by the input natural-language description. Furthermore, a new type of visual loss is utilized to enhance the image resolution by ensuring vivid shape and perceptually uniform color distributions of generated images. Experimental results on benchmark datasets demonstrate the superiority of our proposed method compared to the state-of-the-art models with a multi-stage framework. Visualization of the attention maps shows that the channel-aware attention module is able to localize the discriminative regions, while the pixel-aware attention module has the ability to capture the globally visual contents for the generation of an image.

1. Introduction

Generating high-resolution realistic images conditioned on given text descriptions has become an attractive and challenging task in computer vision (CV) and natural language processing (NLP). It has various potential applications, such as art generation, photo-editing and video games. Recent work has achieved crucial improvements in the quality of generated samples through generative adversarial network (GAN) [7, 23, 24, 39], while also boosting the semantic consistency between generated visually realistic images and given natural-language descriptions.

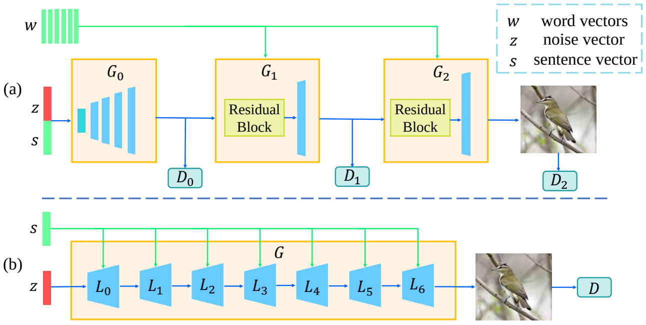


Figure 1. The comparison between the current multi-stage architecture and our model. The multi-stage framework (a) generates final images by training three generators and discriminators. The proposed DTGAN (b) is able to synthesize realistic images only using a single generator/discriminator pair. In (a), G_0-G_2 are generators and D_0-D_2 are discriminators. In (b), L_0-L_6 are the dual-attention layers discussed in Section 3, G and D are our generator and discriminator, respectively.

However, most state-of-the-art approaches in text-to-image generation [12, 18, 22, 33, 35, 37, 38, 41] are based on a multi-stage modular architecture as shown in Figure 1(a). Specifically, the network comprises multiple generators which have corresponding discriminators. Furthermore, the generator of the next stage takes the result of the previous stage as the input. This framework has proven to be useful for the task of text-to-image synthesis, but there still exist three significant problems. Firstly, training many networks increases the computation time compared to a unified model and affects the convergence and stability of the generative model [29]. Even worse, the final generator network cannot be improved if the previous generators do not converge to a global optimum, since the final generator loss does not propagate back. Secondly, this framework ignores the quality of early-stage generator images which plays a vital role in the resolution of finally-generated images [41]. The generator networks for precursor images are only composed of up-sampling layers and convolution layers, lacking the image integration and refinement process with the input natural-language descriptions. Thirdly, multiple discriminators need to be trained.

To address the issues mentioned above, we propose a novel Dual Attention Generative Adversarial Network (DTGAN) which can fine-tune the feature maps for each scale according to the given text descriptions, and synthesize high-quality images only using a single generator/discriminator pair. The overall architecture of the DT-GAN is illustrated in Figure 1(b). Our DTGAN consists of four new components, including two new types of attention modules, a new normalization layer, and a new type of visual loss. The first two components in the DT-GAN are our designed channel-aware and pixel-aware attention modules which can guide the generator network to focus more on important channels and pixels, and to ignore text-irrelevant channels and pixels by computing attention weights between the global sentence vector and two aforementioned factors. Different from earlier attention models [12, 33], we apply the attention scores to fine-tune original feature maps rather than adopt the weighted sum of converted word features as new feature maps. We expect that our proposed attention method will significantly improve the semantic consistency of generated images. In the third ingredient, inspired by Adaptive Layer-Instance Normalization (AdaLIN) [9], we present Conditional Adaptive Instance-Layer Normalization (CAdaILN), where the ratio between Instance Normalization [30] and Layer Normalization [2] is adaptively learned during training and the global sentence vector is employed to scale and shift the normalized result. The CAdaILN function is complementary to the attention modules and helps with controlling the amount of change in shape and texture. As a result, armed with the attention modules and CAdaILN, our network can generate photo-realistic images only exploiting a single generator/discriminator pair. The last proposed component is a new variant for computing visual loss. It is introduced to ensure that generated images and real images have similar color distributions and shape. We expect that the choice of this novel visual loss has a considerable impact on the quality of generated results.

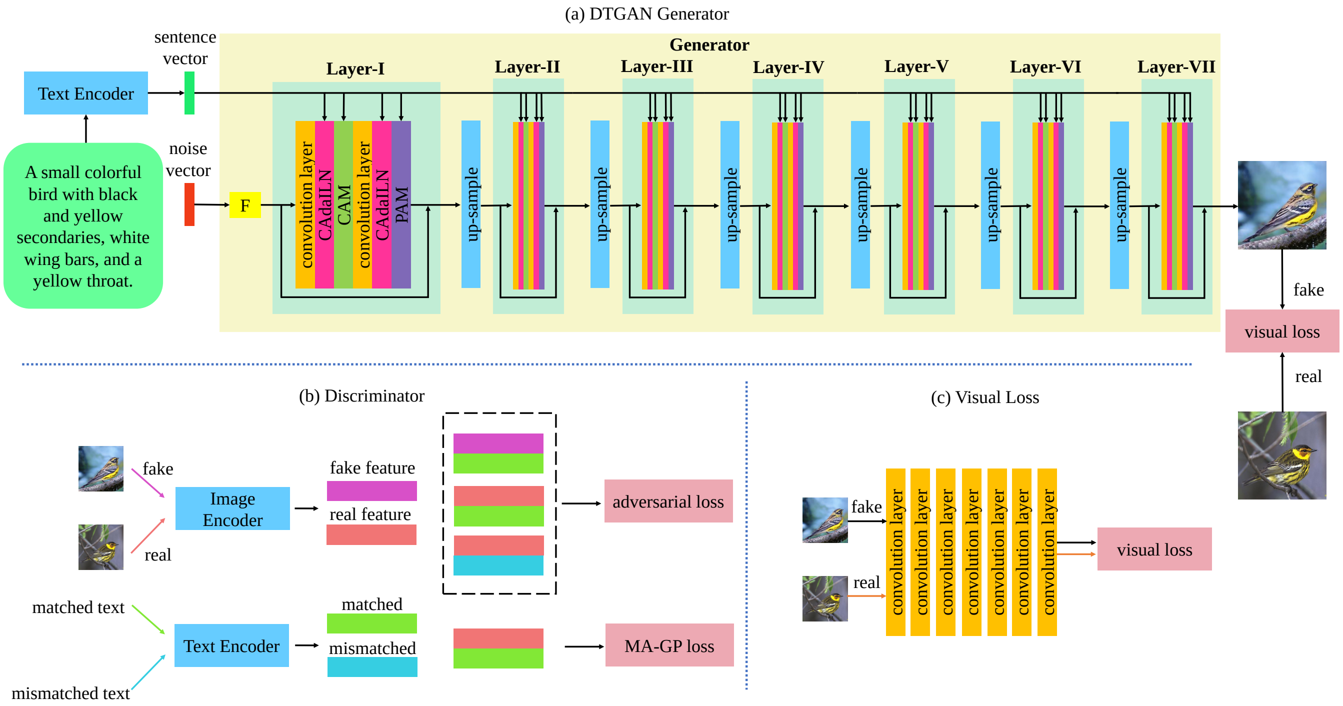
We perform extensive experiments on the CUB bird [32] and MS COCO [17] datasets to evaluate the effectiveness of our proposed DTGAN. Both qualitative and quantitative results demonstrate that our approach outperforms existing state-of-the-art models. In addition, visualization of the attention maps shows that the channel-aware attention module is able to localize the important parts of an image, while the channel-aware attention module has the ability to capture the globally visual contents. The contributions of our work can be summarized as follows:

- To the best of our knowledge, we are the first to propose the fine-tuning on each scale of feature maps using the attention modules and the conditional normalization function, in order to generate high-quality and semantically consistent images only employing a single generator/discriminator pair.
- We design two new types of attention modules to guide the generator to focus on text-relevant channels and pixels, and to refine the feature maps for each scale.
- CAdaILN is presented to help attention modules flexibly control the amount of change in shape and texture.
- We are the first to introduce the visual loss in text-to-image synthesis to enhance the image quality.

2. Related Work

Text-to-Image Generation. In recent years, the task of text-to-image synthesis has attracted rapidly growing attention from both CV and NLP communities. Thanks to the significant improvements in image generation approaches especially GAN, researchers have achieved inspiring advances in the task of text-to-image generation. The conditional GAN [23] was first presented by Reed et al. [24] to generate plausible images from detailed text descriptions. The problem of text-to-image generation was decomposed by Zhang et al. [37, 38] into multiple stages. Each stage accomplished the corresponding task by using different generators and discriminators.We aim to generate high-quality images with photo-realistic details just employing a pair of generator and discriminator. Qiao et al. [22] introduced the image caption model to regenerate the text description from the generated image, in order to enhance the semantic relevancy between the text description and visual content. Zhu et al. [41] applied a dynamic memory module to refine the image quality of the initial stage.

Attention. Attention mechanisms play a vital role in bridging the semantic gap between vision and language. They have been extensively explored in the interdisciplinary fields, such as image captioning [3, 20], visual question answering [1, 10, 13] and visual dialog [4, 19] . Over the past few years, there have been some attention methods for the task of text-to-image generation. Xu et al. [33] utilized a word-level spatial attention mechanism to obtain the relationship between the subregions of the generated image and the words in the input text. The most relevant subregions to the words were very focused. Li et al. [12] designed a word-level channel-wise attention mechanism on the basis of Xu et al. [33], simultaneously taking spatial and channel information into account. However, the aforementioned attention works adopt the weighted sum of converted word features as the new feature map which is largely different from the original feature map. We propose to fine-tune the original feature map using the channel-aware attention weights


(a) DTGAN Generator

(b) Discriminator

(c) Visual Loss

Figure 2. The architecture of the proposed DTGAN. In (a), F is a fully-connected layer, CAM is a channel-aware attention module discussed in Section 3.1, PAM is a pixel-aware attention module discussed in Section 3.2 and CAdaILN is Conditional Adaptive Instance-Layer Normalization discussed in Section 3.3. In (b), MA-GP loss is a Matching-Aware zero-centered Gradient Penalty loss introduced in Section 3.5.

and the pixel-aware attention weights.

3. DTGAN for Text-to-Image Generation

In this section, we elaborate on our proposed DTGAN which is shown in Figure 2. Unlike prior works [12, 18, 22, 33, 35, 37, 38, 41], our goal is to generate a high-quality and visually realistic image which semantically aligns with a given natural-language description only employing a single generator/discriminator pair. To this end, we present four significant components: a channel-aware attention module, a pixel-aware attention module, Conditional Adaptive Instance-Layer Normalization (CAdaILN) and a new type of visual loss. Each of them will be discussed in detail after briefly describing the overall framework of our model.

As shown in Figure 2, our architecture is composed of a text encoder and a generator/discriminator pair. For text encoder, we adopt a bidirectional Long Short-Term Memory (LSTM) network [27] to learn the semantic representation of a given text description. Specifically, in the bidirectional LSTM layer, two hidden states are employed to capture the semantic meaning of a word and the last hidden states are utilized to represent the sentence features.

The generator network of the DTGAN takes a global sentence vector and a noise vector as the input and consists of seven dual-attention layers which are responsible for different scales of feature maps. Each dual-attention layer comprises two convolution layers, two CAdaILN layers, a channel-aware attention module and a pixel-aware attention module. Mathematically,

$$h_0 = F_0(z) \tag{1}$$

$$h_1 = F_1^{Dual}(h_0, s) \tag{2}$$

$$h_i = F_i^{Dual}(h_{i-1} \uparrow, s) \quad for \quad i = 2, 3, ..., 7 \tag{3}$$

$$o = G_c(h_7) \tag{4}$$

where z is a noise vector from the normal distribution, F_0 is a fully-connected layer, F_i^{Dual} is our proposed dual-attention layer, G_c is the last convolution layer, h_0 is the output of the first fully-connected layer, h_1-h_7 are the outputs of dual-attention layers and o is the generated image.

In order to take into account both channel information and spatial pixels, we present the channel-aware and pixel-aware attention modules. Different from AttnGAN [33] and ControlGAN [12], we attend to fine-tune original feature maps for each scale using attention modules, rather than adopt the weighted sum of converted word features as the new feature maps. The experiments conducted on benchmark datasets show the superiority of our proposed attention modules compared to AttnGAN and ControlGAN.

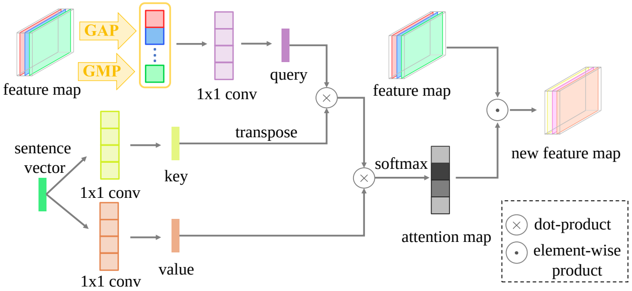


Figure 3. Overview of the proposed channel-aware attention module. GAP and GMP denote global average pooling and global max pooling, respectively.

3.1. Channel-aware Attention Module

The feature map of each channel at the convolution layer plays different roles in generating the image which semantically aligns with the given text description. Without fine-tuning the channel maps at the generative stage according the text description, the generated result can lack the semantic relevancy to the given text description. Thus, we introduce a channel-aware attention module to guide the generator to focus on text-relevant channels and ignore minor channels.

The process of the channel-aware attention module is shown in Figure 3. The channel-aware attention module takes two inputs: the feature map h and the global sentence vector s . Firstly, we perform global average pooling and global max pooling on h to extract the channel features: $x_a \in R^{C \times 1 \times 1}$ and $x_m \in R^{C \times 1 \times 1}$. Global average pooling is to obtain the information of the whole feature map, while global max pooling focuses on the most discriminative part [40]. Mathematically,

$$x_a = \text{GAP}(h) \tag{5}$$
$$x_m = \text{GMP}(h) \tag{6}$$

where GAP denotes global average pooling, GMP is global max pooling.

Then, we adopt a query, key and value setting to capture the semantic relevancy between channels and the input text, where x_a and x_m are used as the query and s is selected as the key and the value. It is defined as:

$$q_{ac} = W_{qa} x_a, q_{mc} = W_{qm} x_m \tag{7}$$
$$k_c = W_{kc} s, v_c = W_{vc} s \tag{8}$$

where W_{qa}, W_{qm}, W_{kc} and W_{vc} are the projection matrixes which are implemented as 1×1 convolutions.

Assuming that the dot products between the sentence-level key k_c^T and the average-pooling query q_{ac}, the max-pooling query q_{mc} can capture meaningful features, the attention scores of channel maps are achieved through the following attention mechanism [31]:

$$\tilde{\alpha}_a^c = q_{ac} \cdot k_c^T, \tilde{\alpha}_m^c = q_{mc} \cdot k_c^T \tag{9}$$
$$\alpha_a^c = \text{softmax}(\tilde{\alpha}_a^c \cdot v_c) \tag{10}$$
$$\alpha_m^c = \text{softmax}(\tilde{\alpha}_m^c \cdot v_c) \tag{11}$$

where $\tilde{\alpha}_a^c$ and $\tilde{\alpha}_m^c$ represent the semantic similarity between channel maps and the global sentence vector, $\alpha_a^c \in R^{C \times 1 \times 1}$ and $\alpha_m^c \in R^{C \times 1 \times 1}$ denote the final attention weights of channels for global average pooling and global max pooling, respectively, $\tilde{\alpha}_a^c$, $\tilde{\alpha}_m^c$, α_a^c and α_m^c are all computed by dot products.

After acquiring the attention weights of channels, we multiply them and the original feature maps to update the feature maps. It is denoted as:

$$o_{ac} = \alpha_a^c \odot h \tag{12}$$
$$o_{mc} = \alpha_m^c \odot h \tag{13}$$

where $\odot$ is the element-wise multiplication. By doing so, the network will focus on the channels which are more semantically related to the given text description.

Meanwhile, the results of global average pooling and global max pooling are fused through concatenation. Specifically,

$$o_c = \sigma(W_c[o_{ac}; o_{mc}]) \tag{14}$$

where W_c is implemented as 1×1 convolution, σ is a non-linear function, such as ReLU.

We further apply an adaptive residual connection [36] to generate the final result. It is defined as follows:

$$y_c = \gamma_c * o_c + h \tag{15}$$

where γ_c is a learnable parameter which is initialized as 0.

As can be seen from above, our designed channel-aware attention model is a fine-tuning module based on channel information and text features. Moreover, it is applied on each scale of feature maps to improve the semantic consistency of generated samples at the generative stage.

3.2. Pixel-aware Attention Module

An image is composed of correlated pixels which are of central importance for the quality and semantic consistency of synthesized images. Thus, we propose a pixel-aware attention module to effectively model the relationships between spatial pixels and the given natural-language description, and to make the important pixels receive more attentions from the generator.

The framework of the pixel-aware attention module is illustrated in Figure 4. Given the feature map $\hat{h}$ and the global sentence vector s, we first exploit average pooling and max pooling to process $\hat{h}$. Specifically,

$$e_a = \text{SAP}(\hat{h}) \tag{16}$$
$$e_m = \text{SMP}(\hat{h}) \tag{17}$$

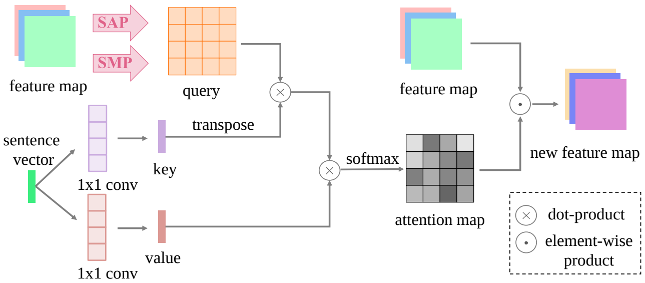


Figure 4. Overview of the proposed pixel-aware attention module. SAP and SMP denote average pooling and max pooling in the spatial dimension, respectively.

where SAP and SMP represent average pooling and max pooling in the spatial dimension, respectively. $e_a \in R^{1\times H\times W}$ and $e_m \in R^{1\times H\times W}$ are the new feature maps.

Then, s is adopted as the key and the value:

$$k_p = W_{kp}s, v_p = W_{vp}s \tag{18}$$

where W_{kp} and W_{vp} are the learnable matrixes which are implemented as 1×1 convolutions.

After that, we compute the dot products of the new feature maps and the key to get the semantic similarity $\tilde{\alpha}_a^p$ and $\tilde{\alpha}_m^p$ between spatial pixels and the global sentence vector. Furthermore, the attention weights are calculated through a softmax function on the dot products of the semantic similarity and the value. It is defined as:

$$\tilde{\alpha}_a^p = e_a \cdot k_p^T, \tilde{\alpha}_m^p = e_m \cdot k_p^T \tag{19}$$

$$\alpha_a^p = \text{softmax}(\tilde{\alpha}_a^p \cdot v_p) \tag{20}$$

$$\alpha_m^p = \text{softmax}(\tilde{\alpha}_m^p \cdot v_p) \tag{21}$$

where α_a^p and α_m^p represent the final attention weights of spatial pixels for average pooling and max pooling, respectively.

Next, same as the channel-aware attention module, we perform a matrix multiplication between the attention weights and the original feature maps to derive the new features o_{ap} and o_{mp}:

$$o_{ap} = \alpha_a^p \odot \hat{h} \tag{22}$$

$$o_{mp} = \alpha_m^p \odot \hat{h} \tag{23}$$

In addition, we concatenate o_{ap} and o_{mp}, and apply a nonlinear function σ to compute the result o_p. Finally, an adaptive residual connection [36] is utilized to combine $\hat{h}$ and o_p. This process is denoted as:

$$o_p = \sigma(W_p[o_{ap}; o_{mp}]) \tag{24}$$

$$y_p = \gamma_p * o_p + \hat{h} \tag{25}$$

where W_p is implemented as 1×1 convolution, σ is a nonlinear function, such as ReLU, γ_p is a learnable parameter which is initialized as 0.

3.3. Conditional Adaptive Instance-Layer Normalization (CAdaILN)

In order to stabilize the training of GAN [5], most existing text-to-image generation models [12, 22, 33, 34, 41] employ Batch Normalization (BN) [8] which applies the normalization to a whole batch of generated images instead for single ones. However, the convergence of BN heavily depends on the size of a batch [15]. Furthermore, the advantage of BN is not obvious for text-to-image generation since each generated image is more pertinent to the given text description and the feature map itself. To this end, CAdaILN, inspired by U-GAT-IT [9], is designed to perform the normalization in the layer and channel on the feature map and its parameters γ and β are computed by a fully-connected layer from the global sentence vector. CAdaILN is able to help with controlling the amount of change in shape and texture based on the input natural-language text. Mathematically,

$$\hat{a}_I = \frac{a-\mu_I}{\sqrt{\sigma_I^2+\epsilon}}, \hat{a}_L = \frac{a-\mu_L}{\sqrt{\sigma_L^2+\epsilon}} \tag{26}$$

$$\gamma = W_1 s, \beta = W_2 s \tag{27}$$

$$\hat{a} = \gamma \odot (\rho \odot \hat{a_I} + (1-\rho) \odot \hat{a_L}) + \beta \tag{28}$$

where a is the processed feature map, μ_I, μ_L and σ_I, σ_L respectively denote the mean and standard deviation in the channel and layer on the feature map, $\hat{a}_I$ and $\hat{a}_L$ represent the output of Instance Normalization (IN) and Layer Normalization (LN) respectively, γ and β are determined by the global sentence vector s, W_1 and W_2 are fully-connected layers, $\hat{a}$ is the output of CAdaILN. The ratio of IN and LN is dependent on a learnable parameter ρ, whose value is constrained to the range of [0, 1]. Moreover, ρ is updated together with generator parameters.

3.4. Visual Loss

To ensure that generated images and real images have similar color distributions and shape, we propose a new type of visual loss for the generator which is illustrated in Figure 2. The visual loss plays a vital role in improving the quality and resolution of finally-generated images. It is based on the image features of the real image I and the generated sample $\hat{I}$, and defined as:

$$L_{vis} = \left| f(I) - f(\hat{I}) \right|_1 \tag{29}$$

where $f(I)$ and $f(\hat{I})$ denote the image features of the real image and the the fake image which are extracted by the discriminator. We impose a L1 loss to minimize the distance between these two image features. To the best of our knowledge, we are the first to present this type of visual loss and apply it in the task of text-to-image generation.

3.5. Objective Function

Adversarial Loss. An adversarial loss is employed to match generated samples to the input text. Inspired by [16, 29, 36], we utilize the hinge objective [16] for stable training instead of the vanilla GAN objective. The adversarial loss for the discriminator is formulated as:

$$\begin{aligned}\mathcal{L}_{\text{adv}}^{D} =& \mathbb{E}_{x\sim p_{\text{data}}}\left[\max(0,1-D(x,s))\right] \\ &+\frac{1}{2}\mathbb{E}_{x\sim p_{G}}\left[\max(0,1+D(\hat{x},s))\right] \\ &+\frac{1}{2}\mathbb{E}_{x\sim p_{\text{data}}}\left[\max(0,1+D(x,\hat{s}))\right]\end{aligned} \tag{30}$$

where s is a given text description, $\hat{s}$ is a mismatched natural-language description.

The corresponding generator loss is:

$$\mathcal{L}_{\text{adv}}^{G} = \mathbb{E}_{x\sim p_G}\left[D(x,s)\right] \tag{31}$$

Matching-Aware zero-centered Gradient Penalty (MA-GP) Loss. To enhance the quality and semantic consistency of generated images, we adopt the MA-GP loss [29] for the discriminator. The MA-GP loss applies gradient penalty to real images and given text descriptions. It is as follows:

$$\mathcal{L}_{\text{M}} = \mathbb{E}_{x\sim p_{\text{data}}}\left[(\|\nabla_x D(x,s)\|_2 + \|\nabla_s D(x,s)\|_2)^p\right] \tag{32}$$

Generator Objective. The generator loss comprises an adversarial loss $\mathcal{L}_{\text{adv}}^{G}$ and a visual loss $\mathcal{L}_{\text{vis}}$:

$$\mathcal{L}_G = \mathcal{L}_{\text{adv}}^{G} + \lambda_1 \mathcal{L}_{\text{vis}} \tag{33}$$

Discriminator Objective. The final objective function of the discriminator is defined as follows:

$$\mathcal{L}_{\text{D}} = \mathcal{L}_{\text{adv}}^{D} + \lambda_2 \mathcal{L}_{\text{M}} \tag{34}$$

4. Experiments

In this section, we carry out a set of experiments on the CUB bird [32] and MS COCO [17] datasets, in order to quantitatively and qualitatively evaluate the effectiveness of the proposed DTGAN. The previous state-of-the-art GAN models in text-to-image synthesis, GAN-INT-CLS [24], GAWWN [25], StackGAN++ [38], AttnGAN [33] and ControlGAN [12], are first compared with our approach. Then, we analyze the significant components of our designed architecture.

4.1. Datasets

Two popular datasets in text-to-image generation, CUB bird and MS COCO datasets, are employed to test our method. The CUB dataset encompasses 11,788 images which are split into 8,855 training images and 2,933 test images. The MS COCO dataset contains 123,287 images which are split into 82,783 training images and 40,504 validation images. Each image in the CUB dataset and MS COCO dataset has ten corresponding text descriptions and five corresponding text descriptions, respectively. We preprocess the CUB dataset using the method in StackGAN [37].

4.2. Evaluation metric

Inception score (IS) [26] and Fréchet inception distance (FID) [28] score are extensively employed in the assessment of text-to-image generation. We adopt theses two indexes as the quantitative evaluation measure and generate 30000 images from unseen text descriptions for each metric.

IS. The IS is to evaluate the visual quality of the generated images via the KL divergence between the conditional class distribution and the marginal class distribution. It's defined as:

$$I = \exp(\mathbb{E}_x[D_{KL}(p(y|\boldsymbol{x}) \parallel p(y))]) \tag{35}$$

where $\boldsymbol{x}$ is a generated sample and y is the corresponding label obtained by a pre-trained Inception v3 network [28]. The generated samples are meant to be diverse and meaningful if the IS is large.

FID. Same as the IS, the FID is also to assess the quality of generated samples by computing the Fréchet distance between the generated image distribution and the real image distribution. We use a pre-trained Inception v3 network to achieve the FID. A lower FID means that the generated samples are closer to the corresponding real images.

However, it is important to note that the IS on the COCO dataset fails to evaluate the image quality and can be saturated, even over-fitted, which is observed by ObjGAN [14] and DFGAN [29]. Therefore, we do not utilize the IS as the evaluation metric on the COCO dataset. We further find that R-precision [6], presented by AttnGAN [33], can not reflect the semantic relation between generated images and given text descriptions, since experimental results show that the R-precision of real images is only 22.22%. Thus, R-precision is not applied on the validation of our model.

4.3. Implementation details

For text encoder, the dimension D is set to 256 and the length of words is set to 18. We implement our model using PyTorch [21]. In the experiments, the network is trained using Adam optimizer [11] with $\beta_1 = 0.0$ and $\beta_2 = 0.9$. We follow the two timescale update rule (TTUR) [6] and set the learning rate of the generator and the discriminator to 0.0001 and 0.0004. The batch size is set to 24. The hyper-parameters p , λ_1 and λ_2 are set to 6, 0.1 and 2, respectively.

4.4. Comparison with State of the Art

Quantitative Results. We compare our model with prior state-of-the-art GAN approaches in text-to-image synthesis

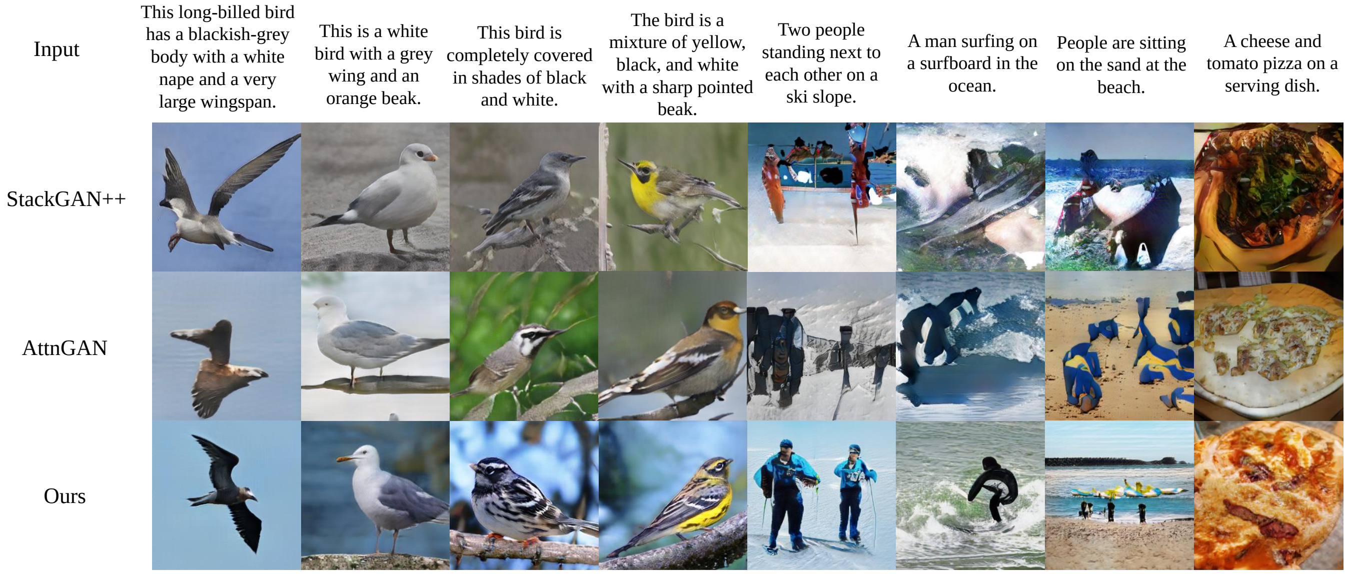

Input
This long-billed bird has a blackish-grey body with a white nape and a very large wingspan.
This is a white bird with a grey wing and an orange beak.
This bird is completely covered in shades of black and white.
The bird is a mixture of yellow, black, and white with a sharp pointed beak.
Two people standing next to each other on a ski slope.
A man surfing on a surfboard in the ocean.
People are sitting on the sand at the beach.
A cheese and tomato pizza on a serving dish.
StackGAN++
AttnGAN
Ours

Figure 5. Qualitative comparison of three approaches conditioned on the text descriptions on the CUB and COCO datasets.

Methods	IS ↑
GAN-INT-CLS [24]	2.88±0.04
GAWWN [25]	3.62±0.07
StackGAN++ [38]	4.04±0.05
AttnGAN [33]	4.36±0.03
ControlGAN [12]	4.58±0.09
Ours	**4.88 ± 0.03**

Table 1. The IS of state-of-the-art approaches and our model on the CUB dataset. The best score is in bold

Datasets	StackGAN++ [38]	AttnGAN [33]	Ours
CUB	26.07	23.98	**16.35**
COCO	51.62	35.49	**23.61**

Table 2. The FID of StackGAN++, AttnGAN and our model on the CUB and COCO datasets. The best results are in bold.

on the CUB and MS COCO datasets. Table 1 reports the IS of our proposed DTGAN and other compared methods on the CUB dataset. We can observe that our model has the best score, significantly improving the IS from 4.58 to 4.88 on the CUB dataset. The experimental results demonstrate that our DTGAN can generate visually realistic images with higher quality and better diversity than state-of-the-art models.

The comparison between our method, StackGAN++ [38] and AttnGAN [33] with respect to FID on the CUB and COCO datasets is shown in Table 2. We can see that our DTGAN achieves a remarkably lower FID than compared approaches on both datasets, which indicates that our generated data distribution is closer to the real data distribution. Specifically, we impressively reduce the FID from 35.49 to 23.61 on the challenging COCO dataset and from 23.98 to 16.35 on the CUB dataset.

Qualitative Results. In addition to quantitative experiments, we perform qualitative comparison with StackGAN++ [38] and AttnGAN [33] on both datasets, which is illustrated in Figure 5. It can be observed that the details of birds generated by StackGAN++ and AttnGAN are lost (2^{th}, 3^{th} and 4^{th} column), the shape is strange (1^{th}, 2^{th} and 3^{th} column) and the colors are even wrong (3^{th} column). Furthermore, the samples synthesized by these two approaches lack text-relevant objects (5^{th}, 6^{th} and 7^{th} column), the backgrounds are unclear and inconsistent with the given text descriptions (5^{th} and 7^{th} column), and the colors are rough (8^{th} column) on the challenging COCO dataset. However, our DTGAN generates more clear and visually plausible images than StackGAN++ and AttnGAN, verifying the superiority of our DTGAN. For instance, as shown in the 1^{th} column, owing to the successful application of the visual loss, a long-wingspan bird with vivid shape is produced by the DTGAN, whereas it is too hard for StackGAN++ and AttnGAN to generate this kind of bird. In the meantime, the birds generated by the DTGAN have more details and richer color distributions compared to StackGAN++ and AttnGAN in the 2^{th}, 3^{th} and 4^{th} column, since the DTGAN armed with channel-aware and pixel-aware attention modules is able to generate high-resolution images which semantically align with given descriptions. More importantly, our method also yields high-quality and visually realistic results on the challenging COCO dataset. For example, the number of the skiers and surfers is correct, the backgrounds are reasonable and people in the images are clear in the 5^{th} and 6^{th} column. Moreover, the beach and the sea are very beautiful in the 7^{th} column and the pizza looks delicious in the 8^{th} column. Generally, these qualita-

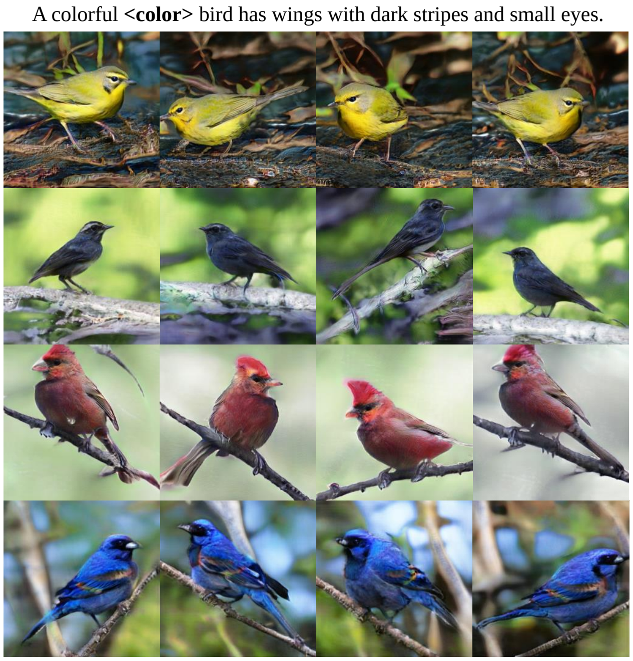

A colorful <**color**> bird has wings with dark stripes and small eyes.

Figure 6. Generated images of the DTGAN by changing the color attribute value in the input text description, for four random draws.

tive results confirm the effectiveness of the DTGAN.

Furthermore, to validate the sensitivity and diversity of our DTGAN, we generate birds by modifying just one word in the given text description. As can be seen in Figure 6, the generated birds are similar but have different poses and shape for the same description. When we change the color attributes in the natural-language descriptions, the proposed DTGAN further produces semantically consistent birds according to the modified text. It means that our approach has the ability to accurately capture the modified part of the text description and to synthesize diverse images for the same natural-language text.

4.5. Component Analysis

In this section, we perform an extensive ablation study on the CUB dataset, so as to evaluate the contributions from different components of our DTGAN. The novel components in our model include a channel-aware attention module (CAM), a pixel-aware attention module (PAM), CAdaILN and a new type of visual loss (VL). We first quantitatively explore the effectiveness of each component by removing the corresponding part in the DTGAN step by step, i.e., 1) DTGAN, 2) DTGAN without the VL, 3) DTGAN without CAdaILN, 4) DTGAN without the PAM, 5) DTGAN without the CAM, 6) DTGAN without the CAM and PAM. All the results are reported in Table 3.

ID	Components				IS ↑	FID ↓
	CAM	PAM	CAdaILN	VL		
1	✓	✓	✓	✓	**4.88 ± 0.03**	**16.35**
2	✓	✓	✓	-	4.72 ± 0.04	19.23
3	✓	✓	-	✓	2.26 ± 0.02	91.53
4	✓	-	✓	✓	4.71 ± 0.05	21.69
5	-	✓	✓	✓	4.60 ± 0.07	22.95
6	-	-	✓	✓	4.54 ± 0.04	23.72

Table 3. Ablation study of our DTGAN. CAM, PAM and VL represent the channel-aware attention module, the pixel-aware attention module and the visual loss, respectively. The best results are in bold.

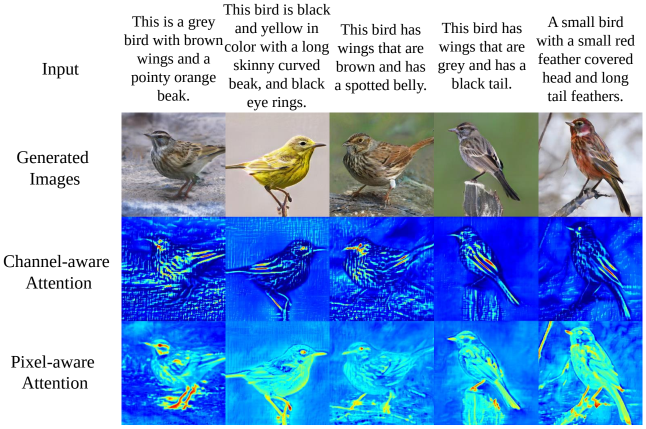

Input
This is a grey bird with brown wings and a pointy orange beak.
This bird is black and yellow in color with a long skinny curved beak, and black eye rings.
This bird has wings that are brown and has a spotted belly.
This bird has wings that are grey and has a black tail.
A small bird with a small red feather covered head and long tail feathers.
Generated Images
Channel-aware Attention
Pixel-aware Attention

Figure 7. Visualization of the channel-aware (detailed features) and pixel-aware (global shape) attention maps.

By comparing Model 1 (DTGAN) with Model 2 (removing the VL), the VL significantly improves the IS from 4.72 to 4.88 and reduces the FID by 2.88 on the CUB dataset, which demonstrates the importance of adopting VL in the DTGAN. By exploiting CAdaILN in our DTGAN, Model 1 performs better than Model 3 (removing CAdaILN) on the IS and FID by 2.62 and 75.18, confirming the effectiveness of the proposed CAdaILN. Both Model 4 (removing the CAM) and Model 5 (removing the PAM) outperform Model 6 (removing the CAM and PAM), indicating that these two new types of attention modules can help the generator produce more realistic images. Furthermore, Model 1 achieves better results than both Model 4 and Model 5, which shows the advantage of combining the CAM and PAM.

To better understand what has been learned by the CAM and PAM during training, we visualize the channel-aware and pixel-aware attention maps for different images in Figure 7. We can see that in the 2^{th} row, the eyes, beaks, legs and wings of birds are highlighted by the channel-aware attention maps. Meanwhile, in the 3^{th} row, the pixel-aware attention maps highlight most important areas of images, including the branches and the whole bodies of birds. This suggests that the CAM helps the generator focus on the crucial parts of birds, while the PAM guides the generator to refine the globally visual contents. Then, the generator can fine-tune the discriminative regions of images obtained by our attention modules.

Visual Loss. To balance the trade-off between image qual-

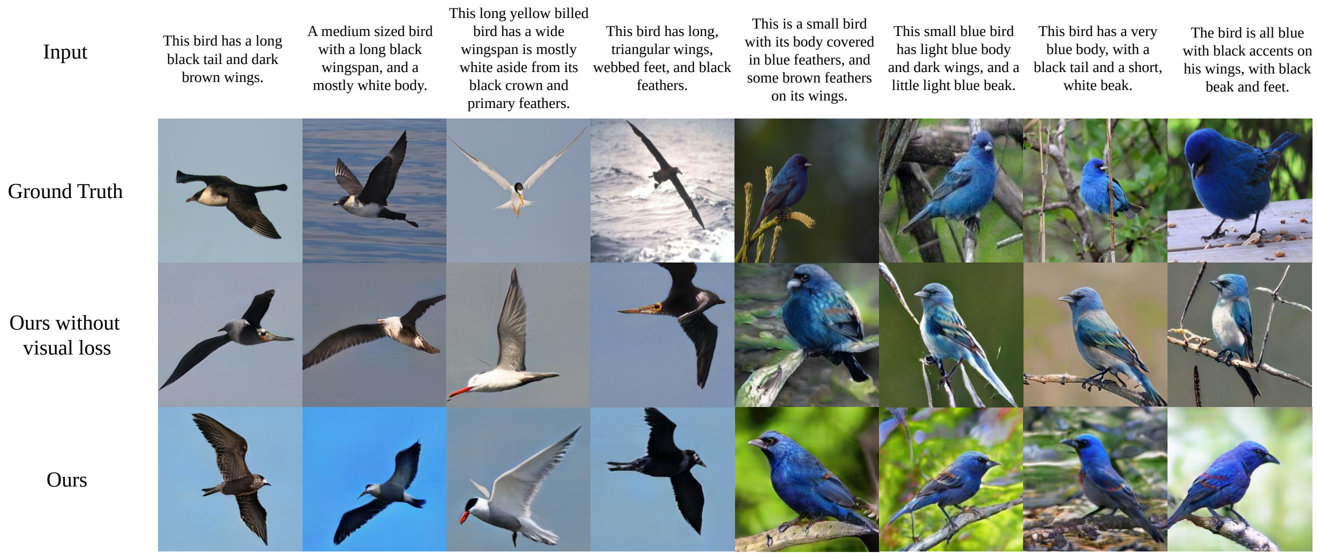

Input
This bird has a long black tail and dark brown wings.
A medium sized bird with a long black wingspan, and a mostly white body.
This long yellow billed bird has a wide wingspan is mostly white aside from its black crown and primary feathers.
This bird has long, triangular wings, webbed feet, and black feathers.
This is a small bird with its body covered in blue feathers, and some brown feathers on its wings.
This small blue bird has light blue body and dark wings, and a little light blue beak.
This bird has a very blue body, with a black tail and a short, white beak.
The bird is all blue with black accents on his wings, with black beak and feet.
Ground Truth
Ours without visual loss
Ours

Figure 8. Visual comparison of the effect of our visual loss (VL) module, yielding more vivid shape and richer color distributions (bottom row).

Parameter	Values	IS ↑	FID ↓
λ_1	0.05	4.74 ± 0.05	18.15
	0.10	$\mathbf{4.88 \pm 0.03}$	**16.35**
	0.15	4.82 ± 0.06	16.75
	0.20	4.59 ± 0.04	20.91
	0.30	4.70 ± 0.06	20.28

Table 4. Evaluation of the DTGAN for different values of λ_1 which is the weight of the visual loss (VL) in the generator. The best result is in bold.

ID	Architecture	IS ↑	FID ↓
1	Baseline	2.26 ± 0.02	91.53
2	+BN-sent	4.67 ± 0.07	19.76
3	+BN-word	4.68 ± 0.04	19.46
4	+CAdaILN	$\mathbf{4.88 \pm 0.03}$	**16.35**
5	+CAdaILN-word	4.71 ± 0.07	19.08

Table 5. Ablation study on CAdaILN. BN-sent indicates Batch Normalization conditioned on the global sentence vector, BN-word indicates Batch Normalization conditioned on the word vectors and CAdaILN-word indicates the CAdaILN function based on the word vectors.

ity and semantic consistency, we investigate the hyper-parameter λ_1 by changing its value in the objective function. We test the value of λ_1 among 0.05. 0.10, 0.15, 0.20 and 0.30. The results are listed in Table 4. We can observe that the best performance is achieved on the CUB dataset if λ_1 is set to 0.1. Therefore, we use λ_1 as 0.1 in the experiments.

In addition, we conduct an ablation study to validate the effectiveness of the VL. The visual comparison between the DTGAN and our model without the VL is shown in Figure 8. We can see that, in the first two columns, the DT-GAN without the VL fails to generate long-wingspan birds with reasonable shape and vivid wings. In the meantime, the proposed model without the VL synthesizes the blue birds which have rough color distributions and lack colorful details in the last two columns. However, the DTGAN produces realistic long-wingspan birds which have semantically consistent shape and colors, while also yielding blue birds with more vivid details and richer color distributions. This indicates that the VL has the ability to potentially ensure the quality of the generated image, including the shape and color distributions of objects in an image.

CAdaILN. To further verify the benefits of CAdaILN, we conduct an ablation study for normalization functions. We first design a baseline model by removing CAdaILN from the DTGAN. Then we compare the variants of normalization layers. Note that BN conditioned on the global sentence vector (BN-sent) and BN conditioned on the word vectors (BN-word) are based on the conditional normalization methods in SDGAN [34], and CAdaILN based on the word vectors (CAdaILN-word) is revised on the basis of CAdaILN according to the word-level normalization method in SDGAN. The results of the ablation study are shown in Table 5. It can be observed that by comparing Model 2 with Model 4 and Model 3 with Model 5, CAdaILN significantly outperforms the BN layer whether using the sentence-level cues or the word-level cues. Moreover, by comparing Model 4 with Model 5, CAdaILN with the global sentence vector performs better than CAdaILN-word by improving the IS from 4.71 to 4.88 and reducing the FID from 19.08 to 16.35 on the CUB dataset, since sentence-level features are easier to be trained in our generator network than word-level features. The above analysis demonstrates the effectiveness of our designed CAdaILN.

5. Conclusion

In this paper, we propose the Dual Attention Generative Adversarial Network (DTGAN), a novel framework for text-to-image generation, to generate high-quality realistic images which semantically align with given text descriptions, only employing a single generator/discriminator pair. DTGAN exploits two new types of attention modules: a channel-aware attention module and a pixel-aware attention module, to guide the generator to focus more on the text-relevant channels and pixels. In addition, to flexibly control the amount of change in shape and texture, Conditional Adaptive Instance-Layer Normalization (CAdaILN) is adopted as a complement to the attention modules. To further enhance the quality of generated images, we design a new type of visual loss which computes the L1 loss between the features of generated images and real images. DTGAN surpasses state-of-the-art results on both CUB and COCO datasets, which confirms the superiority of our proposed method. However, the improved visual quality comes with an apparent reduction in variation of generated images. Future work will be directed at mitigating this phenomenon by using larger training sets.

References

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6077–6086, 2018. 2

[2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 2

[3] Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. Meshed-memory transformer for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10578–10587, 2020. 2

[4] Zhe Gan, Yu Cheng, Ahmed El Kholy, Linjie Li, Jingjing Liu, and Jianfeng Gao. Multi-step reasoning via recurrent dual attention for visual dialog. *arXiv preprint arXiv:1902.00579*, 2019. 2

[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. 5

[6] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in neural information processing systems*, pages 6626–6637, 2017. 6

[7] Seunghoon Hong, Dingdong Yang, Jongwook Choi, and Honglak Lee. Inferring semantic layout for hierarchical text-to-image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7986–7994, 2018. 1

[8] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. 5

[9] Junho Kim, Minjae Kim, Hyeonwoo Kang, and Kwanghee Lee. U-gat-it: unsupervised generative attentional networks with adaptive layer-instance normalization for image-to-image translation. *arXiv preprint arXiv:1907.10830*, 2019. 2, 5

[10] Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. Bilinear attention networks. In *Advances in Neural Information Processing Systems*, pages 1564–1574, 2018. 2

[11] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[12] Bowen Li, Xiaojuan Qi, Thomas Lukasiewicz, and Philip Torr. Controllable text-to-image generation. In *Advances in Neural Information Processing Systems*, pages 2065–2075, 2019. 1, 2, 3, 5, 6, 7

[13] Linjie Li, Zhe Gan, Yu Cheng, and Jingjing Liu. Relation-aware graph attention network for visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 10313–10322, 2019. 2

[14] Wenbo Li, Pengchuan Zhang, Lei Zhang, Qiuyuan Huang, Xiaodong He, Siwei Lyu, and Jianfeng Gao. Object-driven text-to-image synthesis via adversarial training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12174–12182, 2019. 6

[15] Xiangru Lian and Ji Liu. Revisit batch normalization: New understanding and refinement via composition optimization. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3254–3263, 2019. 5

[16] Jae Hyun Lim and Jong Chul Ye. Geometric gan. *arXiv preprint arXiv:1705.02894*, 2017. 6

[17] 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. 2, 6

[18] Shuang Ma, Jianlong Fu, Chang Wen Chen, and Tao Mei. Da-gan: Instance-level image translation by deep attention generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5657–5666, 2018. 1, 3

[19] Yulei Niu, Hanwang Zhang, Manli Zhang, Jianhong Zhang, Zhiwu Lu, and Ji-Rong Wen. Recursive visual attention in visual dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6679–6688, 2019. 2

[20] Yingwei Pan, Ting Yao, Yehao Li, and Tao Mei. X-linear attention networks for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10971–10980, 2020. 2

[21] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, pages 8026–8037, 2019. 6

[22] Tingting Qiao, Jing Zhang, Duanqing Xu, and Dacheng Tao. Mirrorgan: Learning text-to-image generation by redescription. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1505–1514, 2019. 1, 2, 3, 5

[23] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 1, 2

[24] 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. 1, 2, 6, 7

[25] Scott E Reed, Zeynep Akata, Santosh Mohan, Samuel Tenka, Bernt Schiele, and Honglak Lee. Learning what and where to draw. In *Advances in neural information processing systems*, pages 217–225, 2016. 6, 7

[26] 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. 6

[27] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681, 1997. 3

[28] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 6

[29] Ming Tao, Hao Tang, Songsong Wu, Nicu Sebe, Fei Wu, and Xiao-Yuan Jing. Df-gan: Deep fusion generative adversarial networks for text-to-image synthesis. *arXiv preprint arXiv:2008.05865*, 2020. 1, 6

[30] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016. 2

[31] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 4

[32] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 2, 6

[33] 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. 1, 2, 3, 5, 6, 7

[34] Guojun Yin, Bin Liu, Lu Sheng, Nenghai Yu, Xiaogang Wang, and Jing Shao. Semantics disentangling for text-to-image generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2327–2336, 2019. 5, 9

[35] Mingkuan Yuan and Yuxin Peng. Text-to-image synthesis via symmetrical distillation networks. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 1407–1415, 2018. 1, 3

[36] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *International Conference on Machine Learning*, pages 7354–7363. PMLR, 2019. 4, 5, 6

[37] 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. 1, 2, 3, 6

[38] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. Stackgan++: Realistic image synthesis with stacked generative adversarial networks. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1947–1962, 2018. 1, 2, 3, 6, 7

[39] Zizhao Zhang, Yuanpu Xie, and Lin Yang. Photographic text-to-image synthesis with a hierarchically-nested adversarial network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6199–6208, 2018. 1

[40] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016. 4

[41] Minfeng Zhu, Pingbo Pan, Wei Chen, and Yi Yang. Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5802–5810, 2019. 1, 2, 3, 5