Recurrent Affine Transformation for Text-to-Image Synthesis

Senmao Ye, Huan Wang, Mingkui Tan, Member, IEEE, and Fei Liu

Abstract—Text-to-image synthesis aims to generate realistic images conditioned on text descriptions. Recently, conditional affine transformations (CATs), such as conditional batch normalization and instance normalization, have been applied to different layers to control the contents in synthesized images. However, the isolated CAT blocks predict the batch statistics of neighboring layers independently. What’s more, CATs are simple multilayer perceptions that are hard to optimize. To address above issues, we propose a recurrent affine transformation (RAT) that connects all the CAT blocks with a recurrent neural network for modeling the long-term dependency between CAT blocks. To verify the effectiveness of RAT, we conduct both microscopic and macroscopic analyses of RAT, which not only demonstrates the effectiveness of RAT but also turns out to be a useful perspective to analyze how GANs fuse conditional information. In addition, we apply a spatial attention mechanism to the discriminator, which helps the text description to supervise the generator to synthesize more relevant image contents. Extensive experiments on the CUB, Oxford-102, and COCO datasets demonstrate the proposed model’s superiority in comparison to state-of-the-art models.

Index Terms—Text-to-image, generative adversarial network, recurrent neural network, spatial attention.

I. INTRODUCTION

SYNTHESIZING images conditioned on descriptive sentences is a popular research topic in the cross-field of vision and language processing. Due to its various potential applications, such as photoediting, computer-aided design, and virtual scene generation, many methods have been proposed to address this problem, such as generative adversarial networks (GANs) [1], [2], [3] and variational autoencoders [4], [5]. Recently, GAN-based methods have achieved promising results on this task [6], [7], [8], [9], [10].

GANs usually adaptively fuse suitable text information into the synthesis process with multiple isolated fusion blocks, such as conditional batch normalization (CBN) and conditional instance normalization (CIN). CIN [13] is first proposed for style transfer. Afterward, BigGAN [14] and Style-GAN [15] synthesize natural images with impressive visual quality based on CBN and CIN, respectively. Recently, DF-GAN [7], DT-GAN [12], and SSGAN [8] used CIN and CBN to fuse text information into synthesized images. Despite their popularity, CIN and CBN suffer from a severe drawback in that they are isolated in different layers, which ignores the global assignment of text information fused in different layers. Furthermore, isolated fusion blocks are difficult to optimize since they do not interact with each other.

In this paper, we propose a recurrent affine transformation to control all the fusion blocks consistently. As depicted in Fig. 1, RAT expresses different layers’ outputs with standard context vectors of the same shape to achieve unified control of different layers. The context vectors are then connected using a recurrent neural network (RNN) to model long-term dependencies. With the skip connections and weight sharing of RNN, batch statistics among fusion blocks are not only forced to be consistent between neighboring blocks but also reduce training difficulty.

In addition, to improve semantic consistency between texts and synthesized images, we incorporate a spatial attention model.

Fig. 1. Comparison of different fusion strategies. DC-GAN represents early text-to-image models [1], [9], [11] that directly inject text information at the beginning. Style-GAN represents recent text-to-image models [7], [8], [12] that separately inject text information into each layer with a CAT block. RAT-GAN proposes to model the text injection process as a temporal sequence.

Digital Object Identifier 10.1109/TMM.2023.3266607

Authorized license use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
in the discriminator. Hence, text descriptions supervise the generator to synthesize more relevant image contents. Nevertheless, based on extensive experiments, we find that vanilla spatial attention with the softmax function leads to model collapse (the generated pixels are all zero) since softmax suppresses most probabilities to be near zero, leading to increased instability of the GAN. To overcome this, we use a soft threshold function in place of the softmax function, which prevents small attention probabilities from being close to zero. With spatial attention, our discriminator can focus on image regions that are pertinent to the text description, allowing it to supervise the generator more effectively.

The contributions of this paper are as follows:

- We propose a recurrent affine transformation that connects all the fusion blocks to allow global assignment of text information in the synthesis process.
- We incorporate spatial attention into the discriminator to focus on relevant image regions so that the generated images are more relevant to the text description.
- We conduct extensive comparisons and visualizations to demonstrate that the proposed model improves the visual quality and evaluation metrics on the CUB, Oxford-102, and COCO datasets.

II. RELATED WORK

a) Text-to-image Synthesis: Text-to-image synthesis is one of the tasks within conditional image synthesis [16], [17], [18], [19]. With the advent of GANs, conditional GANs have achieved excellent performance on this task. Conditional GAN [20] first proposed the conditional version of GAN, which directly concatenates noise vectors and conditional feature vectors. However, text-to-image models based on this work fuse text information roughly by concatenating the text feature with a noise vector. Vincent et al [13] proposed a more advanced fusion method (i.e., CIN) that uses adaptive mean and variance to control the image style. CIN and its variants have been commonly used in recent works. For example, BigGAN [14] and Style-GAN [15] successively use CBN and CIN to achieve impressive visual quality on ImageNet.

Recent developments have utilized CBN and CIN for integrating text information into synthesis. The semantic-spatial awareness version of CBN was proposed by SSGAN [8] to make CBN aware of spatial regions relevant to text description. As SSGAN uses an attention mask to mask the affine parameters at different locations, relevant image regions are more sensitive to text guidance than irrelevant regions. DTGAN [12] adopts CIN to fuse text information in their model. To better fuse text information into the synthesis process, DF-GAN [7] proposed a deep fusion method that has multiple affine layers in a single block. Different from previous work, DF-GAN discards the normalization operation without a decrease in performance, which reduces computational occupation and limitations from a large batch size. Similar to DF-GAN, RAT-GAN also dumps normalization operations since normalization has no influence on performance.

To mimic human visual attention, spatial attention models [21] have long been studied in deep learning [22]. Inspired by the success of spatial attention models in discriminative tasks [23], [24], some previous work [1], [2] showed that extra spatial structure information could lead to a great improvement in image generation. The method in [1] uses key points and bounding boxes to determine what and where to draw in the image. They represented the key points and bounding boxes as a feature map to feed them into the generator. However, labeling key points and bound boxes requires considerable human labor, and key points are very subjective. Although they also tried to generate key points with an extra GAN, key points are still needed during training. Similar to [1], Wang et al. [25] generated images conditioned on surface maps that can reflect rough sketches of the original images. They also use an extra GAN to synthesize surface maps. The above two models learn to generate handcrafted annotations so that they can synthesize images from scratch with an explicit notion of image spatial structure. However, they both need additional manual annotations to train their model.

b) Matching Text and Image Feature: Graves et al. [26] first proposed an attention model in the context of handwriting synthesis. Another attention model was proposed by [27] for machine translation. It also uses a weighted summation of local annotations as a context feature. However, they computed the weights by the softmax function. Following [27], [28] proposed a soft attention model and a hard attention model for image captioning. The soft attention model uses a weighted summation of local region features as a context vector. The hard attention model uses a single local feature as a context feature. The context feature captures visual information relevant to words in the caption. Our attention model aims to find image regions relevant to the entire caption. Unlike [29], our model does not directly use attention to generate images. We only apply attention to the discriminator to facilitate the comparison between images and captions.

Our work reveals that extremely small activations are a significant cause of GAN collapse and successfully solves this problem by normalizing the activations with a soft threshold. Although spatial attention appears straightforward in improving the performance of the discriminator, it does not work for causing GAN collapse. “That is why previous works have only applied attention to generators such as attnGAN and DTGAN since the pioneering work of [1].” Moreover, our spatial attention probabilities are predicted in a way different from the self-attentions widely used in generators.

III. METHOD

In this section, we aim to construct a GAN that can better control image details with the proposed RAT (see Fig. 2). We begin with an introduction to the pretraining of the text encoder in Section III-A. We then discuss how to build a generator with recurrent affine transformations in Section III-B. The final Section III-C describes how to build a discriminator based on spatial attention.

A. Contrastive Text Embedding Pretraining

We use a bidirectional long short-term memory network (LSTM) to encode each text description into a sentence-level
Fig. 2. (a) Generator with the proposed recurrent affine transformation for text-to-image synthesis. The fusion blocks are connected by an RNN to ensure the global assignment of text information. (b) A discriminator with spatial attention can focus on image regions relevant to the text description, which helps to judge the authenticity of the input image.

A feature vector \( s \in \mathbb{R}^d \) and a convolution network to encode each image into an image-level feature vector \( f \in \mathbb{R}^d \). To train these two feature extractors, we adopt a contrastive loss that maximizes the image-text similarity among a batch of training samples. Following AttnGAN [30], we first calculate the similarity matrix for all possible text-image pairs by

\[
M = [s_1, s_2, \ldots, s_n]^T [f_1, f_2, \ldots, f_n],
\]

where \( n \) is the image number in a batch. \( s_i \) and \( f_i \) denote the \( i \)th text feature and image feature, respectively. \( M_{i,j} \) denotes the dot-product similarity between the \( i \)th text feature and \( j \)th image feature. The similarity matrix \( M \) is turned into match probabilities as follows:

\[
\hat{M}_{i,j} = \frac{\exp(M_{i,j})}{\sum_{j=1}^{n} \exp(M_{i,j})}.
\]

To maximize the similarity of text and image features belonging to the same pair, we minimize the contrastive loss:

\[
L = -\sum_{i=1}^{n} \log \hat{M}_{i,i}.
\]

The pretrained text feature extractor maps texts into text feature vectors, which reduces the training difficulty of conditional GANs.

B. Recurrent Affine Transformation

In this section, we aim to use a recurrent affine transformation in the generator to enhance the consistency between fusion blocks of different layers. We begin by introducing the proposed RAT.

RAT first conducts a channelwise scaling operation on an image feature vector \( c \) with the scaling parameter and then applies the channelwise shifting operation on \( c \) with the shifting parameter. This process can be formally expressed as follows:

\[
\text{Affine}(c \mid h_t) = \gamma_t \cdot c + \beta_t,
\]

where \( h_t \) is the hidden state of an RNN and \( \gamma, \beta \) are parameters predicted by two one-hidden-layer MLPs conditioned on \( h_t \).

\[
\gamma_t = \text{MLP}_1(h_t), \quad \beta_t = \text{MLP}_2(h_t).
\]

When applied to an image feature map composed of \( w \times h \) feature vectors, the same affine transformation is repeated for every feature vector. For increasing network depth and nonlinearity, as depicted in Fig. 2, several RNN units, RATs, and convolutions are stacked to form an RAT block.

**a) Recurrent Controller:** We use an RNN to model the temporal structure within the RAT blocks to assign text information on a global basis. Instead of a vanilla RNN, we use the widely used LSTM variant. The initial states of the LSTM are computed from the noise vector \( z \):

\[
h_0 = \text{MLP}_3(z), \quad c_0 = \text{MLP}_4(z).
\]

The update rule of RAT is formulated as follows:

\[
\begin{align*}
\dot{i}_t & = \sigma(s_t), \\
\dot{f}_t & = \sigma(s_t), \\
\dot{o}_t & = \sigma(s_t), \\
\dot{u}_t & = \tanh(c_t), \\
\gamma_t & = \text{MLP}_1(h_t), \quad \beta_t = \text{MLP}_2(h_t),
\end{align*}
\]

where \( i_t, f_t, o_t, u_t \) are the input gate, forget gate, and output gate, respectively. \( T : \mathbb{R}^{D+d} \rightarrow \mathbb{R}^d \) is an affine transformation, where \( d \) is the dimensionality of the text embedding and \( D \) is the quantity of the RNN hidden state units.
Furthermore, two layer-specific MLPs are used to convert each hidden state $h_t$ into $γ_t$ and $β_t$ because each convolution layer has different channel dimensions. Different from conditional instance normalization, conditional batch normalization, and conditional deep fusion [7], our work no longer takes affine transformation as isolated modules. In contrast, we employ RNN to model the long-term dependency between fusion blocks, which not only forces the fusion blocks to be consistent with one another but also reduces the difficulty of training with skip connections.

Finally, as depicted in Fig. 2(a), we use a one-stage generator consisting of 6 upsample blocks to synthesize fake images. The noise vector $z \sim N(0, 1)$ is sampled from a standard Gaussian distribution and fed into the generator at the beginning. Each upsample block is followed by an RAT block to control the image contents. The last $256 \times 256$ feature map is turned into a fake image of size $256 \times 256 \times 3$ by a hyperbolic tangent function.

C. Matching-Aware Discriminator With Spatial Attention

To improve the semantic consistency between the synthesized image and text description, we incorporate spatial attention into the discriminator. Being aware of matching image regions, text descriptions supervise the generator to synthesize more relevant image contents. As depicted in Fig. 2(b), several downsample blocks are used to encode the image into an image feature map $P$. Combining the information in the image feature map $P$ and sentence vector $s$, spatial attention produces an attention map $α$ that suppresses sentence vectors for irrelevant regions. The precise formulation of this attention model is:

$$x_{w,h} = MLP(P_{w,h}, s),$$  
(11)

$$α_{w,h} = \frac{1}{1+e^{-x_{w,h}}},$$  
(12)

$$S_{w,h} = s \times α_{w,h},$$  
(13)

where $S_{w,h}$ is the feature channel of $S$ at location $\{w, h\}$. $P_{w,h}$ and $s$ are fed into a multilayer perception with one hidden layer to compute an energy value $x_{w,h}$, and then this energy value is converted into attention probabilities $α_{w,h}$. Finally, $S$ is concatenated with $P$ and fed into the latter downsample blocks to produce the global feature presentation. To stabilize GAN training, attention probabilities $α$ are predicted by the soft threshold function [31]:

$$p(x_k) = \frac{1}{1+e^{-x_k}}.$$  
(14)

The soft threshold uses a standard logistic function to squash the negative energy values between $(0, 1)$ before normalization. We do not adopt the popular softmax function since it maximizes the largest probability and suppresses other probabilities to be near 0. The extremely small probabilities hamper the back-propagation of the gradients, which worsens the instability of GAN training. In contrast, the soft threshold function prevents attention probabilities from being close to zero and increases the efficiency of back propagation. The spatial attention model assigns more text features to relevant image regions, which helps the discriminator to determine whether the text image pair matches. In adversarial training, the stronger discriminator forces the generator to synthesize more relevant image content.

D. Objective Functions

Similar to [1], the training objective for the discriminator takes the synthesized image and the mismatched images as negative samples. To keep the gradient smooth, we use hinge loss with MA-GP [7] on real and match text pairs. The precise formulation of the training objective of the discriminator is formulated as follows:

$$L^D_{adv} = \mathbb{E}_{x \sim p_{data}} [\max(0, 1 - D(x, s))] + \frac{1}{2} \mathbb{E}_{x \sim p_G} [\max(0, 1 + D(\hat{x}, s))]$$
$$+ \frac{1}{2} \mathbb{E}_{x \sim p_{data}} [\max(0, 1 + D(x, \hat{s}))],$$  
(15)

where $s$ is the given text description and $\hat{s}$ is a mismatched text description. The corresponding training objective of the generator is:

$$L^G_{adv} = \mathbb{E}_{x \sim p_{data}} [\min(D(x, s))].$$  
(16)

IV. Experiments

a) Datasets: We report results on the popular CUB [32], Oxford-102 [33], and MS COCO datasets [34]. The CUB dataset has 200 different categories with 11,788 images of birds in total. The Oxford-102 dataset contains 102 different categories with 8,189 images of flowers in total. As in [35], [36], we split images into class-disjoint training and test sets. The CUB dataset has 150 train classes and 50 test classes, while the Oxford-102 dataset has 82 train+val and 20 test classes. For both datasets, 10 captions are used per image. The MS COCO dataset consists of 123,287 images with 5 sentence annotations. The official training split of COCO is used for training, and the official validation split of COCO is used for testing. During mini-batch selection for training, a random image view (e.g., crop, flip) is chosen for one of the captions.

b) Training Details: The text encoder is pretrained by the Adam optimizer with a learning rate of 0.002, an output of size 256, and a batch size of 48; then, it is frozen during the training of GAN. The random noise is sampled from a 100-dimensional standard Gaussian distribution. The Adam optimizer is used to optimize the network with base learning rates of 0.0001 for the generator and 0.0004 for the discriminator. We used a mini-batch size of 24 to train the model for 600 epochs on the CUB and Oxford-102 datasets. The training of the CUB dataset takes approximately 3 days on a single NVIDIA RTX 3090 Ti GPU. To accelerate training, we used a mini-batch size of 48 and trained the model for 300 epochs on the COCO dataset. With two RTX 3090 Ti GPUs, the COCO dataset takes approximately two weeks to train.

c) Evaluation Metrics: We adopt the widely used inception score (IS) [37] and Fréchet inception distance (FID) [38] to quantify the quantitative performance. On the MS COCO dataset, an
TABLE I

| Methods         | IS $\uparrow$ (CUB, Oxford) | FID $\downarrow$ (CUB, Oxford, COCO) | Extra Supervision |
|-----------------|-----------------------------|--------------------------------------|-------------------|
| StackGAN++ [2]  | 4.05 ± 0.06, 3.26 ± 0.06    | 15.30, 48.68, 81.59                  | -                 |
| AttnGAN [31]    | 4.36 ± 0.03, -               | 23.98, - 35.49                       | ✓                 |
| CSM-GAN [17]    | 4.62 ± 0.08, -               | - , - 26.77                          | ✓                 |
| KD-GAN [18]     | 4.90 ± 0.06, -               | - , - 23.92                          | ✓                 |
| DM-GAN [9]      | 4.75 ± 0.07, -               | 16.09, - 32.64                       | -                 |
| DTGAN [12]      | 4.88 ± 0.03, -               | 16.35, - 23.61                       | ✓                 |
| DF-GAN [7]      | 5.10 ± -,-, 3.80 ± 0.06      | 14.81, 18.90, 21.42                  | -                 |
| SSAG [8]        | 5.17 ± 0.08, -               | 15.61, - 19.37                       | ✓                 |
| RAT-GAN(Ours)   | 5.36 ± 0.20, 4.09 ± 0.06     | 13.91, 16.04, 14.60                  | -                 |

The results are mostly taken from the original papers, and the best results are highlighted in bold.

Inception-v3 network pretrained on Image-net is used to compute the KL-divergence between the conditional class distribution (generated images) and the marginal class distribution (real images). The presence of a large IS indicates that the generated images are of high quality. The FID computes the Fréchet distance between the image feature distribution of the generated and real-world images. The image features are extracted by the same pretrained Inception v3 network. A lower FID implies that the generated images are closer to the real images. To evaluate the IS and FID scores, 30k images are generated by each model from the test dataset as previous works [7], [8], [12], [39] reported that the IS metric completely fails in evaluating the synthesized images. Therefore, we only compare the FID on the COCO dataset. On the CUB and Oxford-102 datasets, pretrained Inception models are fine-tuned on two fine-grained classification tasks [2].

d) Compared Models: We compare RAT-GAN with recent state-of-the-art methods: StackGAN++ [2], DM-GAN [9], DF-GAN [7], DAE-GAN [10], SSACN [8], AttnGAN [30], DTGAN [12].

A. Comparisons With Others

a) Quantitative Results: In this section, we present results on the CUB dataset of bird images, MS COCO dataset of common object images, and Oxford-102 dataset of flower images in Table I. For simplicity, we denote our text-to-image model as RAT-GAN. On the CUB dataset, our model achieves the highest Inception scores (5.36) and lowest FID scores (13.91) compared with other state-of-the-art models. On the Oxford dataset, our model achieves the highest Inception scores (4.09). The results of DF-GAN on the Oxford dataset are based on its publicly available code on GitHub. The superiority of RAT-GAN is more obvious on the COCO dataset, where images are more complicated. According to the quantitative results, the proposed recurrent fusion strategy performs better in complex scenes. Noticeably, our model has no extra supervision, such as DAMSM loss (SSGAN) or cycle consisted loss (MirrorGAN). We also do not fine-tune the text encoder. Extensive results prove the effectiveness of RAT.

b) Qualitative Results: In this section, we present qualitative results on the CUB dataset of bird images and the Oxford-102 dataset of flower images. The visualization results of stackGAN, DF-GAN, and RAT-GAN are displayed in Fig. 3. StackGAN is a classical multistage method, and DF-GAN is a popular one-stage method for text-to-image synthesis. Attn-GAN also uses an attention model to boost performance. On the CUB dataset, we observe that our model performs much better than DF-GAN and stackGAN with clear details such as feathers, eyes, and feet. Moreover, the background is also more reasonable in the results of RAT-GAN. On the Oxford dataset, RAT-GAN has better texture and more relevant colors than the others. With the proposed RAT and the spatial attention model, our model has fewer distorted shapes and more relevant contents than the other two models. In addition, we observe a new collapse mode from stackGAN that suppresses the image pixels to be 0.5/1 in the $128 \times 128$ and $256 \times 256$ branches (the first $64 \times 64$ branch is normal), which leads to vaguer images than the other two models. Hence, we use the results before such collapse at approximately 200 epochs.

As depicted in Fig. 4, compared to DF-GAN and Attn-GAN, the qualitative superiority of RAT-GAN is more obvious on the COCO dataset because of the more complex scenarios and texts. Equipped with recurrent interactions, fusion blocks are treated as a union, which allows RAT to achieve a more complex level of control over synthesized images. When compared with the baseline model DF-GAN, RAT-GAN can synthesize objects with more regular shapes and clear semantic meanings. To better illustrate the image quality on COCO, we visualize more images of people, food, buildings, and vehicles in Fig. 5. For simplicity, the corresponding captions have been omitted. On the COCO dataset, our model outperforms other models that also have no external information, such as additional training data [40] or text encoders pretrained on a large-scale corpus.

B. Ablation Studies

We present quantitative and qualitative evaluation results of model components for the ablation study of the recurrent controller and spatial attention. To reduce the computational burden,
Fig. 3. Qualitative comparison on the CUB and Oxford datasets. The input text descriptions are given in the first row, and the corresponding generated images from different methods are shown in the same column. Best view in color and zoom in.

Fig. 4. Qualitative comparison of our model with other methods on the COCO dataset. Best view in color and zoom in.

Fig. 5. More visualization samples on the COCO dataset.

the models trained in this section have only half the dimensions of the full model.

a) Analysis of RAT Step: In ID 0∼3 of Table II, more affine layers result in better text-to-image performance. In particular, the performance improvement is most obvious in ID 1. When
more RAT blocks are stacked in ID 2 and ID 3. RAT gradually acquires better control over the image contents. However, there is no obvious improvement in ID 4 because there are too many layers, which makes the optimization too difficult.

b) Analysis of Recurrent Connection: In Table II, ID 5 substitutes RAT units with MLPs, which results in a much worse performance than RATs in ID 1, thus indicating that recurrent connection is effective. Moreover, in ID 6, performance worsens with more MLP blocks since stacked MLPs are difficult to optimize. Finally, in ID 7 and 8, we add residual connections among CATs, which turns out to be detrimental to the final results. Because different conditional information is needed for each layer, residual connections copy feature vectors directly from previous layers. As a result, the skip connection results in model collapse. Additionally, as shown in Fig. 7, connecting CAT blocks with RNN (RAT-GAN) or MLP blocks (MLP) leads to faster convergence than the baseline (Baseline) because isolated fusion blocks are hard to optimize. However, stacked MLP blocks are also difficult to optimize. Hence, MLP soon saturates. As a result, RAT-GAN performs better than both MLP and Baseline.

c) Visualization of RNN Activations: To analyze the microscopic characteristics of the RAT, we visualize the RNN activations at each layer in Fig. 8. Specifically, we randomly choose some channels among 256 RNN channels and visualize their input gate $i_t$, output gate $o_t$, and forget gate $f_t$. The variation in RNN activation shows three distinct features: 1) neighboring layers prefer similar semantic information, directly demonstrating that semantic consistency naturally exists among adjacent layers; 2) the forget gate retains information from previous layers with probabilities mostly larger than 0.5, which also demonstrates semantic consistency between neighboring layers; and 3) the front and rear layers have completely opposite semantic preferences, demonstrating that they have different functions.

To analyze the macroscopic characteristics of the RAT, in Fig. 9, we first reduce the impact of randomness by averaging the probabilities of 256 RNN channels, which makes the activation curves more concentrated than single channels. In subfigure (a), the variations of different texts and different noises are nearly random. In subfigure (b), the noise vector determines the variation of RNN activations, which indicates that the synthesis process is almost controlled by the noise vector. In subfigure (c), the activation variations converge in front layers, which indicates that the design of image styles is well accomplished in front layers. Thereafter, the fused text information is fixed and irrelevant to the noise vector.

To sum up, we conduct both microscopic and macroscopic analyses of RAT, which not only demonstrates the effectiveness of RAT but also provides a useful perspective to analyze how GANs fuse conditional information.

d) Analysis of the Dimensionality, Initialization, and Gradient Exploration of RAT: The hidden state dimension is one of the most important hyperparameters for RNNs. In Table III, RNN is the vanilla recurrent unit, and LSTM is the long-short term unit used in our full model. We further conduct ablation studies on another RNN variant called the gated recurrent unit (GRU). It has been observed that RNNs prefer large channel dimensions due to their simple architecture. When the dimension is set to 512, the LSTM performs optimally.

The hidden state is initialized by the noise vector $z$, which is another input of the RAT. To study the importance of inputting $z$, we replace $z$ with a random vector and a zero vector. In Table III, IDs 6, 7, 12, and 13 perform obviously worse than IDs 5 and 10, which indicates that the noise vector is indispensable for the RAT to control the overall text information fusion.

Gradient exploration usually occurs in recurrent neural networks, and RAT-GAN has 22 recurrent units in a single forward process. To enhance performance, we attempt to incorporate the popular gradient clipping strategy. However, gradient clipping appears to degrade the model’s performance because the smoothness of the gradient is crucial for GAN performance. In other words, clipping the gradient degrades the continuity and stability of GAN training.

### Table II

| ID | Steps | IS ↑ | FID ↓ |
|----|-------|------|------|
| 0  | RAT   | 4.51±0.14 | 23.36 |
| 1  | RAT   | 4.83±0.18 | 20.78 |
| 2  | MLP   | 4.91±0.22 | 16.69 |
| 3  | MLP   | 5.02±0.32 | 15.02 |
| 4  | MLP   | 4.83±0.32 | 16.90 |
| 5  | 1(Res)| 4.60±0.02 | 22.25 |
| 6  | 4(Res)| 4.02±0.20 | 33.68 |
| 7  | 1(Res)| 3.72±0.01 | 45.64 |
| 8  | 4(Res)| Collapse | Collapse |

MLPs means replacing rat blocks with stacked mlp blocks. To reduce the computational burden, we use a base dimension of 32. The best results are highlighted in bold.

### Table III

| ID     | Model  | Init | Clip | IS ↑ | FID ↓ |
|--------|--------|------|------|------|------|
| 0      | RNN-256| Noise| -    | 4.64±0.23 | 22.31 |
| 1      | RNN-512| Noise| -    | 4.70±0.09 | 20.64 |
| 2      | RNN-512| Noise| ✓    | 4.56±0.09 | 24.45 |
| 3      | RNN-1024| Noise| -   | 4.74±0.15 | 19.88 |
| 4      | GRU-256| Noise| -    | 4.76±0.15 | 19.73 |
| 5      | GRU-512| Noise| -    | 4.95±0.09 | 16.56 |
| 6      | GRU-512| Random| - | 4.56±0.13 | 23.46 |
| 7      | GRU-512| Zero  | -    | 4.43±0.04 | 26.78 |
| 8      | GRU-1024| Noise| -    | 4.81±0.18 | 19.98 |
| 9      | LSTM-256| Noise| -    | 4.86±0.04 | 17.75 |
| 10     | LSTM-512| Noise| -    | 5.02±0.32 | 15.02 |
| 11     | LSTM-512| Noise| ✓    | 4.55±0.09 | 26.23 |
| 12     | LSTM-512| Random| - | 4.66±0.13 | 23.56 |
| 13     | LSTM-512| Zero | -    | 4.61±0.12 | 23.66 |
| 14     | LSTM-1024| Noise| -    | 4.70±0.11 | 18.11 |

To reduce the computational burden, we use half the dimension of the full model. The best results are highlighted in bold.
This bird has long primaries and short rectrices with a white belly and black wings.
The bird has brown crown with gray throat, black and white breast, and brown wingbars.
A small bird with a blue head and black patches around its black eyes, the wings are green.
This bird has a white head with yellow bill, its body is gray, and its tail feathers are black.
This flower is white, yellow, and purple in color, with petals that are striped near the center.
This flower has several purple petals with a circular purple pattern in the middle.
This flower has long white petals that grow away from the center and fold back at the edges.
This flower has long white petals that grow away from the center and fold back at the edges.

Fig. 6. Qualitative comparison on the CUB and Oxford datasets. The input text descriptions are given in the first row, and the corresponding generated images from different methods are shown in the same column. Best view in color and zoom in.

Fig. 7. Variation in Inception scores during training on the CUB dataset. RAT-GAN, MLP, and Baseline are compared.

e) Analysis of Spatial Attention: We conducted qualitative and quantitative experiments to verify the effectiveness of the proposed spatial attention model in Table II and Fig. 6, respectively. By adding spatial attention to ID 3, the performance increases from 4.56 to 4.77. Comparing ID 3 and 4, the $4 \times 4$ attention size seems not large enough to effectively discriminate images; hence, a larger attention size of $8 \times 8$ obtains a slightly better performance. However, applying attention with a larger attention size $16 \times 16$ does not result in better performance since image features from early layers differ greatly from text features. In ID 1 and 2, spatial attention with the softmax function leads to model collapse (the generated pixels are all zero). Therefore, the IS and FID are not available. In contrast, spatial attention with a soft threshold prevents such model collapse by preserving small probabilities, which indicates that stable gradients are crucial to the stability of GANs.

f) The Influence of Test Images: FID is typically computed using generated images and training data. However, current text-to-image models use a test split with totally different species to compute the FID score, which differs slightly from the original definition. To study the effect of test data, we tested our full model on different splits of the data in Table V. According to our results, the test-test pair in ID 1 scores the highest FID score, and the test-all pair in ID 2 scores the lowest FID score since the generated images are closer to the training data and more images decrease the variance. On the COCO dataset, such a difference is nearly negligible. The results of our study indicate that test data are important for a fair comparison, particularly for datasets with large differences between training data and test data.

g) The Influence of Fine-tuned Inception Models: To calculate the FID scores, current models such as DF-GAN and DM-GAN utilize Inception models pretrained on ImageNet datasets. However, the ImageNet dataset is very different from the CUB and Oxford datasets. According to the results of IDs 1 and 4 in Table V, pretrained Inception cannot distinguish well between the different bird species in the training and testing splits. As a result, the pretrained Inception model produces the
Fig. 8. Variation of channel activations in a forward propagation. The probabilities of input gate $i_t$, output gate $o_t$, and forget gate $f_t$ are regarded as the activations of the corresponding channels. Each curve represents a randomly selected channel, and we display 9 input gates, 9 output gates, and 9 forget gates in total.

Fig. 9. Variation in the average input channel activations in different layers. Corresponding texts of (a) and (b) are displayed below them. To reduce the impact of randomness, the average probability of 256 hidden channels is regarded as the activation of the corresponding layer.

**TABLE IV**

| ID | Components      | Att size | IS $\uparrow$       | FID $\downarrow$   |
|----|-----------------|----------|----------------------|---------------------|
| 0  | Soft Threshold  | -        | 4.56 ± 0.06          | 23.34               |
| 2  | -               | 4×4      | Collapse             | Collapse            |
| 3  | √               | 16×16    | Collapse             | Collapse            |
| 4  | √               | 8×8      | 4.87 ± 0.20          | 18.42               |
| 5  | √               | 16×16    | 4.80 ± 0.07          | 19.13               |
| 6  | √               | Mixed    | 4.83 ± 0.16          | 18.91               |

To reduce the computational burden, we use half the dimension of the full model. The best results are highlighted in bold.

**TABLE V**

| ID | Dataset | Text | Image   | Inception Model | FID $\downarrow$ |
|----|---------|------|---------|-----------------|-------------------|
| 0  | CUB     | Test | Train   | ImageNet        | 11.28             |
| 1  | CUB     | Test | Test    | ImageNet        | 15.58             |
| 2  | CUB     | Test | Train+Test | ImageNet    | 10.21             |
| 3  | CUB     | Test | Train   | Fine-tune       | 12.92             |
| 4  | CUB     | Test | Test    | Fine-tune       | 40.59             |
| 5  | CUB     | Test | Train+Test | Fine-tune    | 13.91             |
| 6  | COCO    | Test | Train   | ImageNet        | 11.28             |
| 7  | COCO    | Test | Test    | ImageNet        | 11.38             |

The best results are highlighted in bold.

That the FID score decreases with more species that never appear in the training image.

To make the Inception model aware of the difference between test and training images, we use fine-tuned Inception models [2] to compute the FID score. Therefore, fine-tuned Inception has the lowest FID when tested on a test-train pair since the extra test
image split differs greatly from the train image split. However, it is also important for evaluating the generalization ability of text-to-image models. Hence, we report ID 5 with both test and training images as our final results.

Moreover, the pretrained Inception model prefers models with extra supervision from ImageNet since fake images are forced to be similar to images on ImageNet. Our new FID evaluation code is available on GitHub.

h) Analysis of Truncating the Noise Distribution: A moderate improvement in IS and FID scores can be achieved by truncating the noise distribution, according to the experimental results in Table VI. Truncation enhances image quality by removing images with low probability density. From ID 0 to 8, the truncation ratio $r$ is gradually increased from 0.25 to 5.0. Then, $r$ is multiplied to a standard normal distribution clipped between $[-2,2]$ as DF-GAN [7]. The FID score is degraded slightly by a small $r$, which obviously improves IS scores. In the case of a large $r$, the FID score improves slightly, but the IS scores remain similar to those without truncation.

i) Batch Size Analysis: The batch size has a significant impact on the final performance, as shown in Table VII. Contrary to the results reported by Big-GAN [14], we find that too large batch sizes also decrease performance because RAT has no normalization operation that requires a large batch size to stabilize its statistics. Moreover, small batch sizes with greater randomness may result in better local optima. However, a small batch size may result in inferior performance because the gradients are too unstable.

j) Diversity of Generated Images: To qualitatively demonstrate the diversity of RAT-GAN, we generated multiple images conditioned on the same text description and different noise vectors. In Fig. 10, 12 images are synthesized conditioned on the same text. These images share the same semantic content with different spatial structures, demonstrating that RAT-GAN can effectively control the contents of the images. Moreover, it successfully disentangles the semantic information and spatial information into the text feature and the noise vector, respectively.

k) Visualization of Attention Maps: To verify the effectiveness of our spatial attention model, we visualize the attention map $\alpha$ of size $8 \times 8$. For the convenience of observation, we upsample the attention maps to the image size of $256 \times 256$ with bilinear interpolation. Moreover, most attention probabilities are suppressed by normalization in the soft threshold function; hence, we amplify the attention probabilities according to the equation below:

$$\alpha' = \frac{(\alpha - \min(\alpha))}{(\max(\alpha) - \min(\alpha))}. \quad (17)$$

Some attention maps and their corresponding images are visualized in Fig. 11. Clearly, spatial attention enables the discriminator to identify regions relevant to the caption, which enables the discriminator to make a more accurate comparison.

l) Failure Cases and Discussion: According to our experiments, RAT-GAN could generate novel bird species by mixing existing species. We display four interesting cases in Fig. 12. In subfigure (a), our model generates a hybrid of a belted kingfisher and a hummingbird. In subfigure (b), RAT-GAN generates two bird heads in a single image, indicating that it is not aware of strict logic. It can be seen in subfigure (c) that the glaucous winged gull has a brown neck that is actually white. In subfigure (d), the mallard’s body is the same green color as its head. Inspired by the above phenomena, it is interesting to incorporate...
This bird has white, light brown, and medium brown small splotches. This bird is blue and white in color with a stubby beak and black eye rings. This flower is rose-shaped with orange overlapping petals and green sepals. The petals of the flower are purple in color and have green leaves.

Images

A small black and white bird with a long, pointed beak. This bird has wings that are green and has an orange belly and blue head. The bird has a large bill that is yellow and curved. This bird has a green crown, brown and blue primaries, and a brown belly.

Fig. 11. Attention maps predicted by the spatial attention model. Best view in color and zoom in.

Fig. 12. New bird species synthesized by RAT-GAN on the CUB dataset.

domain-specific knowledge to judge the authenticity of generated images.

V. CONCLUSION

In this article, we address the text-to-image task by GAN with the proposed recurrent affine transformation. The main difficulty of this task lies in effectively fusing text information into the image synthesis process. Previous models usually use isolated fusion blocks to fuse suitable information adaptively. In contrast, RAT successfully improves image quality by adding interactions among fusion blocks through RNN. The mutual interactions not only ensure consistency between neighboring blocks but also reduce training difficulty. In addition, to improve semantic consistency, we incorporate a spatial attention model into the discriminator. Extensive experiments on different datasets demonstrate that our model obviously improves the image quality. In the future, it will be interesting to apply RAT to other image synthesis tasks.

REFERENCES

[1] S. E. Reed et al., “Generative adversarial text to image synthesis,” in Proc. Int. Conf. Mach. Learn., 2016, pp. 1060–1069.

[2] H. Zhang et al., “StackGAN++: Realistic image synthesis with stacked generative adversarial networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 41, no. 8, pp. 1947–1962, Aug. 2019.

[3] S. Fang, H. Xie, J. Chen, J. Tan, and Y. Zhang, “Learning to draw text in natural images with conditional adversarial networks,” in Proc. Int. Joint Conf. Artif. Intell., 2019, pp. 715–722.

[4] K. Gregor, I. Danihelka, A. Graves, D. J. Rezende, and D. Wierstra, “DRAW: A recurrent neural network for image generation,” in Proc. Int. Conf. Mach. Learn., 2015, pp. 1462–1471.

[5] E. Mansimov, E. Parisotto, L. J. Ba, and R. Salakhutdinov, “Generating images from captions with attention,” in Proc. 4th Int. Conf. Learn. Representations, Y. Bengio and Y. LeCun, Eds. San Juan, PR, USA, May 2–4, 2016.

[6] T. Qiao, J. Zhang, D. Xu, and D. Tao, “Mirrorgan: Learning text-to-image generation by redescription,” in Proc. IEEE/CVF Comput. Vis. Pattern Recognit., 2019, pp. 1505–1514.

[7] M. Tao et al., “DF-GAN: Deep fusion generative adversarial networks for text-to-image synthesis,” in Proc. IEEE/CVF Comput. Vis. Pattern Recognit., 2022, pp. 16494–16504.

[8] W. Liao, K. Hu, M. Y. Yang, and B. Rosenhahn, “Text to image generation with semantic-spatial aware GAN,” in Proc. IEEE/CVF Comput. Vis. Pattern Recognit., New Orleans, LA, USA, Jun. 18–24, 2022, pp. 18166–18175.

[9] M. Zhu, P. Pan, W. Chen, and Y. Yang, “DM-GAN: Dynamic memory generative adversarial networks for text-to-image synthesis,” in Proc. IEEE/CVF Comput. Vis. Pattern Recognit., 2019, pp. 5802–5810.

[10] S. Ruan et al., “DAE-GAN: Dynamic aspect-aware GAN for text-to-image synthesis,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., Montreal, QC, Canada, Oct. 10–17, 2021, pp. 13940–13949.

[11] H. Zhang, T. Xu, and H. Li, “Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks,” in Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 5908–5916.

[12] Z. Zhang and L. Schomaker, “DTGAN: Dual attention generative adversarial networks for text-to-image generation,” in Proc. IEEE Int. Joint Conf. Neural Netw., 2021, pp. 1–8.

[13] V. Dumoulin, J. Shlens, and M. Kudlur, “A learned representation for artistic style,” in Proc. 5th Int. Conf. Learn. Representations, Toulon, France, Apr. 24–26, 2017. [Online]. Available: https://openreview.net/forum?id=Bj0O-BuTf

[14] A. Brock, J. Donahue, and K. Simonyan, “Large scale GAN training for high fidelity natural image synthesis,” in Proc. 7th Int. Conf. Learn. Representations, New Orleans, LA, USA, May 6–9, 2019. [Online]. Available: https://openreview.net/forum?id=BJsxiq09Fm

[15] T. Karras, S. Laine, and T. Aila, “A style-based generator architecture for generative adversarial networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 43, no. 12, pp. 4217–4228, Dec. 2021.

[16] F. Feng, T. Niu, R. Li, and X. Wang, “Modality disentangled discriminator for text-to-image synthesis,” IEEE Trans. Multimedia, vol. 24, pp. 2112–2124, 2022.

[17] H. Tan, X. Liu, B. Yin, and X. Li, “Cross-modal semantic matching generative adversarial networks for text-to-image synthesis,” IEEE Trans. Multimedia, vol. 24, pp. 832–845, 2022.

[18] J. Peng et al., “Knowledge-driven generative adversarial network for text-to-image synthesis,” IEEE Trans. Multimedia, vol. 24, pp. 4356–4366, 2022.

[19] X. Hou, X. Zhang, Y. Li, and L. Shen, “TextFace: Text-to-style mapping based face generation and manipulation,” IEEE Trans. Multimedia, early access, Mar. 17, 2022, doi: 10.1109/TMM.2022.3160360.

[20] M. Mirza and S. Osindero, “Conditional generative adversarial nets,” 2014, arXiv:1411.1784.

[21] M. Dong and C. Xu, “On retrospecting human dynamics with attention,” in Proc. 28th Int. Joint Conf. Artif. Intell., Macao, China, 2019, pp. 708–714.

[22] X. Zhu, D. Cheng, Z. Zhang, S. Lin, and J. Dai, “An empirical study of spatial attention mechanisms in deep networks,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., Seoul, South Korea, 2019, pp. 6688–6697.

[23] X. Li, W. Zhou, Y. Zhou, and H. Li, “Relation-guided spatial attention and temporal refinement for video-based person re-identification,” in Proc. AAAI Conf. Artif. Intell., 2020, pp. 11434–11441.

[24] F. Li et al., “Gamma-enhanced spatial attention network for efficient high dynamic range imaging,” in Proc. IEEE/CVF Comput. Vis. Pattern Recognit. Workshops, New Orleans, LA, USA, 2022, pp. 1032–1040.

[25] X. Wang and A. Gupta, “Generative image modeling using style and structure adversarial networks,” in Proc. Eur. Conf. Comput. Vis., 2016, pp. 318–335.

[26] A. Graves, G. Wayne, and I. Danihelka, “Neural turing machines,” 2014arXiv:1410.5401.
[27] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” in Proc. 3rd Int. Conf. Learn. Representations, Y. Bengio and Y. LeCun, Eds. San Diego, CA, USA, May 7–9, 2015. [Online]. Available: http://arxiv.org/abs/1409.0473

[28] K. Xu et al., “Show, attend and tell: Neural image caption generation with visual attention,” in Proc. Int. Conf. Mach. Learn., 2015, pp. 2048–2057.

[29] E. Mansimov, E. Parisotto, L. J. Ba, and R. Salakhutdinov, “Generating images from captions with attention,” in Proc. 4th Int. Conf. Learn. Representations, Y. Bengio and Y. LeCun, Eds. San Juan, PR, USA, May 2–4, 2016. [Online]. Available: http://arxiv.org/abs/1511.02793

[30] T. Xu et al., “AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2018, pp. 1316–1324.

[31] S. Ye, J. Han, and N. Liu, “Attentive linear transformation for image captioning,” IEEE Trans. Image Process., vol. 27, no. 11, pp. 5514–5524, Nov. 2018.

[32] P. Welinder et al., “Caltech-UCSD birds-200,” California Inst. Technol., Pasadena, CA, USA, CNS-TR-2010-001, 2010.

[33] M. Nilsback and A. Zisserman, “Automated flower classification over a large number of classes,” in Proc. IEEE 6th Indian Conf. Comput. Vis., Graph. Image Process., 2008, pp. 722–729.

[34] T. Lin et al., “Microsoft COCO: Common objects in context,” in Proc. Eur. Conf. Comput. Vis., 2014, pp. 740–755.

[35] S. E. Reed et al., “Learning what and where to draw,” in Proc. Adv. Neural Inf. Process. Syst., 2016, pp. 217–225.

[36] S. E. Reed et al., “Generative adversarial text to image synthesis,” in Proc. Int. Conf. Mach. Learn., 2016, pp. 1060–1069.

[37] T. Salimans et al., “Improved techniques for training Gans,” in Proc. Adv. Neural Inf. Process. Syst., 2016, pp. 2226–2234.

[38] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “Gans trained by a two time-scale update rule converge to a local nash equilibrium,” in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 6626–6637.

[39] W. Li et al., “Object-driven text-to-image synthesis via adversarial training,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 12174–12182.

[40] M. D. M. Reddy, M. S. M. Basha, M. M. C. Hari, and M. N. Penchalaiah, “Dall-e: Creating images from text,” UGC Care Group J. J., vol. 8, no. 14, pp. 71–75, 2021.

Senmao Ye received the B.E. and master’s degree from Northwestern Polytechnical University, Xi’an, China. He is currently working toward the doctor’s degree with the School of Software Engineering, South China University of Technology, Guangzhou, China. His research interests include computer vision and natural language processing, especially on image synthesis.

Huan Wang received Ph.D. degree in information and communication engineering from Shenzhen University, Shenzhen, China, in 2019. He is currently a Research Assistant Professor with the Industrial Technology Research Center, Guangdong Institute of Scientific & Technical Information. His research interests include computer vision, image processing, pattern recognition, and scientific information research.

Mingkui Tan (Member, IEEE) received the bachelor’s degree in environmental science and engineering and the master’s degree in control science and engineering from Hunan University, Changsha, China, in 2006 and 2009, respectively, and the Ph.D. degree in computer science from Nanyang Technological University, Singapore, in 2014. From 2014 to 2016, he was a Senior Research Associate in computer vision with the School of Computer Science, The University of Adelaide, Adelaide, SA, Australia. He is currently a Professor with the School of Software Engineering, South China University of Technology, Guangzhou, China. His research interests include machine learning, sparse analysis, deep learning, and large-scale optimization.

Fei Liu received the B.E. degree in automatic testing and control, the M.E. and Ph.D. degrees in control science and engineering from the Harbin Institute of Technology, Harbin, China. He was with the Department of Computer Science, Brandenburg University of Technology, Cottbus, Germany, from 2009 to 2012 and with the Department of Computer, Yamaguchi University, Yamaguchi, Japan, from 2015 to 2016 as a JSPS Scholar. He is currently a Professor with the School of Software Engineering, South China University of Technology, Guangzhou, China. His research interests include modeling and simulation, artificial intelligence, and systems biology.