Customizable GAN: customizable image synthesis based on adversarial learning

Zhiqiang Zhang[1], Wenxin Yu[1,*], Jinjia Zhou[2], Xuewen Zhang[1], Siyuan Li[1], Ning Jiang[1], Gang He[1], Gang He[3], and Zhuo Yang[4]

[1] Southwest University of Science and Technology
[2] Hosei University
[3] Xidian University
[4] Guangdong University of Technology
*yuwenxin@swust.edu.cn, star_yuwenxin27@163.com

Abstract. Controllable image synthesis in computer vision is always an essential but challenging task. At present, there are two main methods in this area. One is based on contour synthesis to achieve control of the basic shape of the synthetic object. This method achieves encouraging results, but it cannot control the details of the synthesis. The other method is based on the text description of the synthetic image, which effectively realizes the control of the synthetic content, but it is powerless for the shape control of the object. In this paper, we propose a highly flexible and controllable image synthesis method based on the simple contour and text description. The contour determines the object's basic shape, and the text describes the specific content of the object. The method is verified in the Caltech-UCSD Birds (CUB) and Oxford-102 flower datasets. The experimental results demonstrate its effectiveness and superiority. Simultaneously, our method can synthesize the high-quality image synthesis results based on artificial hand-drawing contour and text description, which demonstrates the high flexibility and customizable of our method further.

Keywords: Computer vision · Deep Learning · Customizable Synthesis · Generative Adversarial Networks.

1 Introduction

In computer vision, image synthesis is always essential but challenging research. In recent years, with the development of deep learning, especially the introduction of generative adversarial networks (GAN) [1], image synthesis has made a significant breakthrough. However, the input of the original GAN is the noise vector of Gaussian distribution or even distribution, resulting in image synthesis that cannot artificially control.

To make the image synthesis process more controllable, it is necessary to provide high-level control information. The current research is mainly from two aspects: one is to control the shape of the synthesis through the contour information, the other is to control the specific content of the synthesis through

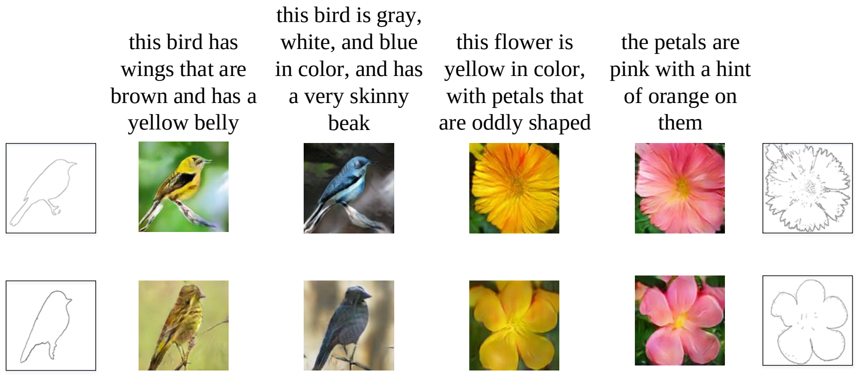
this bird has wings that are brown and has a yellow belly
this bird is gray, white, and blue in color, and has a very skinny beak
this flower is yellow in color, with petals that are oddly shaped
the petals are pink with a hint of orange on them

Fig. 1. The figure display the results of the corresponding birds and flowers under different texts and contours. The contours above are obtained by pre-processing the original dataset. The contours below are drawn by hand.

the text information. Some achievements [2,3] have been made in the aspect of shape control by contour, but the biggest problem in this aspect is that it can only control the shape information, not the specific content. The method of text control starts with image attributes or class labels [4,5], which can control the categories of synthetic content but cannot do anything for more specific details. Furthermore, Reed *et al.* [6] proposed a method of synthesizing images based on the text description, which makes the whole synthesis process more flexible and better control the specific content of the synthesis. However, the synthesis method based on the text description can not control the shape information of the synthesized object.

In order to achieve better control effect and synthesize more realistic results, a customizable synthesis method based on simple contour and text description is proposed. As shown in Fig. 1, the simple contour is used to determine the specific shape information for the object. The text description is then used to generate specific content. Finally, the high-quality images based on hand-drawn contour and artificial text description are obtained by using this method. It not only realizes fine-grained control but also completes the generation of the realistic image.

The main contributions of this paper are as follows: (1) an effective customizable image synthesis method is proposed, and it can achieve fine-grained control and high-quality image generation. (2) the whole process of image synthesis can be controlled manually (drawing the shape and describing content manually), which makes our method most flexible. (3) experiments on the Caltech-UCSD Birds (CUB) [7] and the Oxford-102 flower [8] datasets demonstrate the effectiveness of the method.

The rest of this paper is arranged as follows. The related research works of image synthesis is briefly reviewed in Section 2. Our method details are discussed in Section 3 and validated in Section 4 with promising experimental results. Section 5 concludes our work.

2 Related work

In order to achieve controllable image synthesis, conditional image synthesis is explored. The generation of initial conditions is based on simple image attributes or class labels [4,5], which has achieved good results. However, due to the need for some professional knowledge, it is not suitable for human basic input habits. In addition, using attributes or class labels can not control details. In contrast, there is a way to use simple contour [2,3] to control the shape of the synthetic object, which has better control strength. However, it can only control the basic shape but not the specific content.

At present, the method that accords with people's input habits and has better control power is to use text descriptions to synthesize images. Reed *et al.* [6] first implemented text-to-image synthesis using the end-to-end GAN architecture and achieved encouraging results. Subsequently, many improvement methods [9,10,11,12,13,14,15] have been put forward and achieved amazing results. Although the results of text-to-image synthesis are more real and larger, there is the same problem — based on the same text description, multiple results with different shapes, sizes, and orientations can be synthesized. It means that the input text can only control the generated content, but not the specific shape information.

To solve this problem, Reed *et al.* [16] proposed the Generative Adversarial What-Where Network (GAWWN), and realized the controllable image generation for the first time by combining the location information and text description. Although the method has achieved good control, on the one hand, the results of the method are not satisfactory. On the other hand, the bounding box and key points information used in this method belongs to the rough information, which can not realize the refined control of the object shape. By contrast, our customizable generation method combines simple contour and text description to generate high-quality results, which realizes fine-grained image control generation and effectively solves the problems in GAWWN.

3 Customizable GAN

3.1 Network architecture

The architecture of our approach is shown in figures 2 and 3. Fig. 2 shows the network structure of the generator. In the generator, the simple contour and text description as the input is encoded in different ways and then combined. After that, the corresponding result is synthesized by de-convolution [17].

Specifically, in the generator, the contour is convoluted by a three-layer convolution neural network (CNN), followed by a ReLU activation operation. In

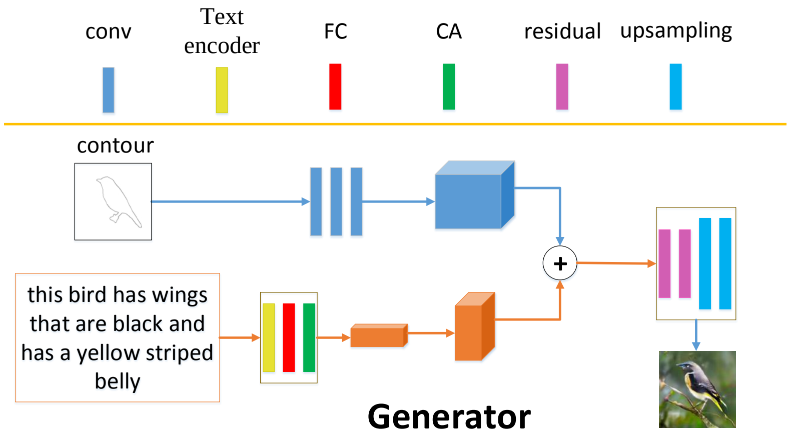
conv
Text encoder
FC
CA
residual
upsampling
contour
+
this bird has wings that are black and has a yellow striped belly
Generator

Fig. 2. The generator structure of the model. The generator synthesizes the corresponding image based on the text description and contour.

addition to the first layer, each ReLU has a Batch Normalization (BN) [18] before it. The text description is first encoded as the text vector by a pre-trained text encoder [19], then expanded dimension by full connection layer. Finally, increase the number of text embeddings by condition augmentation (CA) [9] technology. The specific implementation equation of CA is as follows:

$$D_{KL}(\mathcal{N}(\mu(\varphi_t),\ \sum(\varphi_t))\ ||\ \mathcal{N}(0,\ I)) \tag{1}$$

where $\mathcal{N}$ represents a Gaussian distribution, φ_t represents the encoded text vector, μ and $\sum$ represent the operation of the mean and diagonal covariance matrix, respectively. KL represents the Kullback-Leibler divergence, $\mathcal{N}(0,\ I)$ is a regularization term to prevent over-fitting.

In order to effectively combine text embeddings with contour features, spatial replication is performed to expand the dimension of text embeddings. After dimension expansion, the dimension of the contour feature is $16 \times 16 \times 512$, and the dimension of text embeddings is $16 \times 16 \times 128$. After the contour feature and text embeddings are fused, it will pass through two residual units, which are composed of residual blocks [20]. After that, the corresponding image results are synthesized by two up-sampling operations.

There are two parts of the content in the discriminator. One is to judge whether the input image is true or false; the other is to judge whether the input image and text match. Fig. 3 shows the network structure of the discriminator. In the discriminator, the corresponding feature vector of the input image is obtained through the down-sampling operation. The down-sampling operation is divided into two types: one is to get the corresponding feature through two convolution layers and use it to judge whether the image is true or false; the other is to obtain the input image feature through five convolutions and then combine the extended dimension text vector to judge whether the image and the text

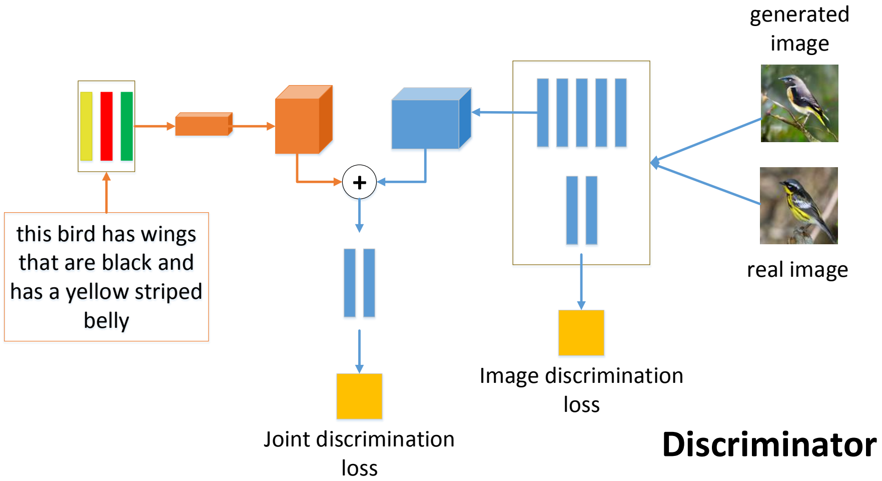
generated image
this bird has wings that are black and has a yellow striped belly
real image
Image discrimination loss
Joint discrimination loss
Discriminator

Fig. 3. The discriminator structure of the model. The discriminator judges whether the received image itself is true or fake and the matching degree between the image and text.

match. In the image discrimination loss, the first convolution layer is followed by BN and leaky-ReLU [21]; the sigmoid function directly follows the second layer. In the joint discrimination loss, the combined image and text features are used to calculate the loss by two convolution layers. BN and leaky-ReLU operation follow the first convolution layer. In the discriminator, the specific features extraction dimension in the discriminator is 4 × 4 × 512, and the text embeddings dimension is 4 × 4 × 128.

3.2 Adversarial learning

There are three types of text input in the adversarial training process, that is, the matching text T, the mismatching text T_{mis}, and the relevant text T_{rel}. In the specific training, the generator synthesizes the corresponding image results through the simple contour and text description, and then the generated results will be sent to the discriminator. In the discriminator, it needs to distinguish three situations: the real image with the matched text, the fake image with the relevant text, the real image with the mismatched text. In each case, the discriminator will distinguish the authenticity of the image and the consistency between image and text. The specific loss functions are as follows:

$$L_G = \sum\nolimits_{(I,T)\sim p_{data}} \log D_0(I_{fake}, T_{rel}) + \log D_1(I_{fake}, T_{rel}) \tag{2}$$

$$\begin{aligned}L_D = \textstyle\sum_{(I,T)\sim p_{data}} \{\log D_0(I_{real}, T) + [\log(1 - D_0(I_{real}, T_{mis}))\\ + \log(1 - D_0(I_{fake}, T_{rel}))]/2\}\\ +\{\log D_1(I_{real}, T) + [\log D_1(I_{real}, T_{mis})\\ + \log(1 - D_1(I_{fake}, T_{rel}))]/2\}\end{aligned} \tag{3}$$

where D_0 represents the first output of the discriminator, and D_1 represents the second.

3.3 Training details

In the training process, the initial learning rate is set to 0.0002, and it decays to half of the original every 100 epochs. Adam optimization [22] with a momentum of 0.5 is used to optimize and update parameters. A total of 600 epochs are trained iteratively in the network, of which the batch size is 64.

4 Experiments

4.1 Dataset and data preprocessing

We validated our method on the CUB and the Oxford-102 flower datasets. Ten text descriptions are collected for each image. The CUB dataset contains 11,788 images with 200 classes. The Oxford-102 dataset contains 8,189 images with 102 classes. Following Reed *et al.* [6], we split CUB dataset to 150 train classes and 50 test classes as well as Oxford-102 to 82 train classes and 20 test classes.

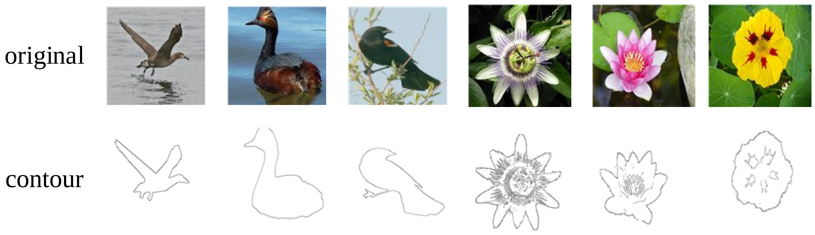

Fig. 4. Some processed contour results. As a result, the foreground in the original image is well drawn in the form of curves. The results of flowers not only contain the outline information of the outermost part but also include the interior information.

In order to experiment with customizable synthesis, it is necessary to preprocess the contour. For the processing of the bird dataset, we first download the corresponding binary image on its official website, then turn the black part of the background into white and retain the outermost contour lines. For the contour map of the flower dataset, we use the Canny operator to process the flower foreground map, the official website provides the foreground map of the blue background, and pure foreground map can be obtained by turning the blue to white. The related results are shown in Figure 4.

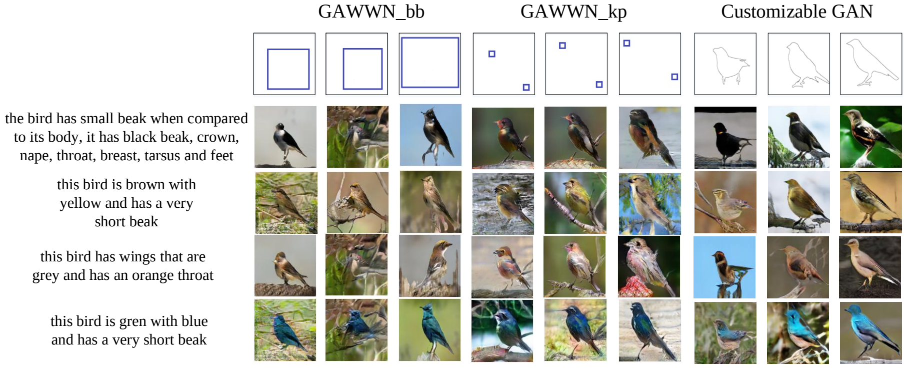
GAWWN_bb
GAWWN_kp
Customizable GAN
the bird has small beak when compared to its body, it has black beak, crown, nape, throat, breast, tarsus and feet
this bird is brown with yellow and has a very short beak
this bird has wings that are grey and has an orange throat
this bird is gren with blue and has a very short beak

Fig. 5. The comparison between our method and GAWWN (including two results based on bounding box and key points).

4.2 Qualitative results

Compare our method with the existing controllable image synthesis based on text and annotations (GAWWN), as shown in Figure 5. There are two kinds of comments in GAWWN [20], the bounding box, and the key points. In the figure, GAWWN_bb represents the GAWWN result based on the bounding box. GAWWN_kp represents the corresponding result based on the key points. The synthesis results based on the bounding box, and key points generally have poor authenticity. By contrast, the results synthesized by our method have better authenticity as a whole. In detail processing, such as smoothness and texture, our results are also better than GAWWN. Besides, the resulting shape of GAWWN is rough, and the generated shape cannot be controlled accurately. Our method can control the specific shape precisely because it inputs contour information.

For the contour feature extraction, we use three methods. The first is to use the convolution layer directly; the second is to use VGG16 [23] model extraction; the third is to use VGG19 model extraction. We have carried out experiments in all three ways, and the comparison results are shown in Figure 6. It can be seen in the figure that these three methods have achieved better results. Specifically, the results of feature extraction with VGG are better than those without VGG in some details, such as eyes and tail.

Our method has also carried out an extended experiment in the flower dataset, and compared results obtained by the three methods are as shown in Figure 7.

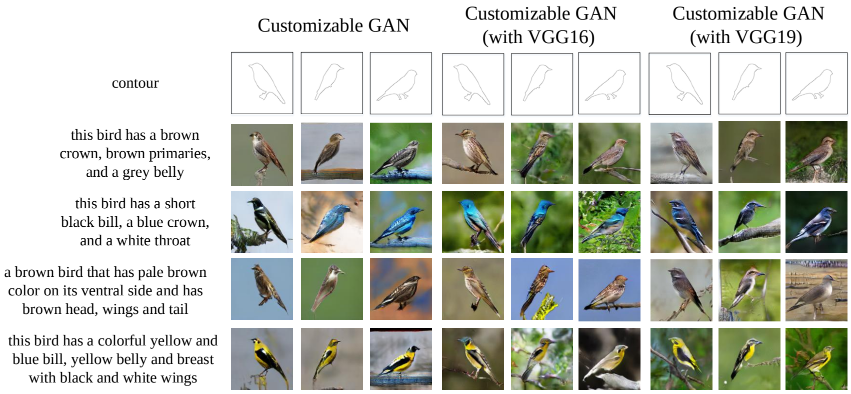
Customizable GAN
Customizable GAN (with VGG16)
Customizable GAN (with VGG19)
contour
this bird has a brown crown, brown primaries, and a grey belly
this bird has a short black bill, a blue crown, and a white throat
a brown bird that has pale brown color on its ventral side and has brown head, wings and tail
this bird has a colorful yellow and blue bill, yellow belly and breast with black and white wings

Fig. 6. The comparison bird results of our method without VGG and with VGG16, with VGG19. It can be seen that the results of using VGG are better in details (such as eyes, pecking).

The results also show that the method using VGG is better than that without VGG in some detail texture processing.

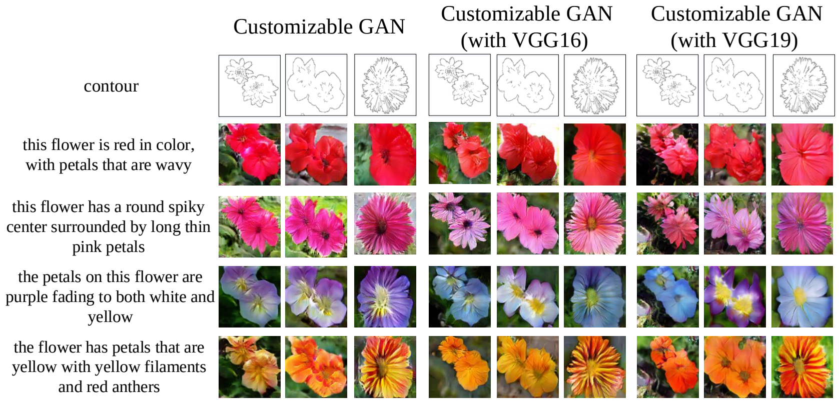
Customizable GAN
Customizable GAN (with VGG16)
Customizable GAN (with VGG19)
contour
this flower is red in color, with petals that are wavy
this flower has a round spiky center surrounded by long thin pink petals
the petals on this flower are purple fading to both white and yellow
the flower has petals that are yellow with yellow filaments and red anthers

Fig. 7. The comparison flower results of our method without VGG and with VGG16, with VGG19.

4.3 Quantitative results

For the evaluation of the generation model, Human Rank (HR) is used to quantify the comparison models. We employed 10 subjects to rank the quality of synthetic images by different methods. The text descriptions and contours corresponding to these results are all from the test set and are divided into 10 groups

Table 1. The quantitative comparison results in CUB dataset.

	GAWWN_bb	GAWWN_kp	ours
consistency	2.78	2.51	1.26
text	2.46	2.28	1.44
authenticity	2.67	2.34	1.42

Table 2. The internal quantitative comparison results of our methods in CUB dataset.

	ours	ours+VGG16	ours+VGG19
consistency	1.269	1.202	1.212
text	1.440	1.402	1.382
authenticity	1.421	1.320	1.243

for use by 10 subjects. The subjects are asked to rank the results in the following ways: consistency, text, and authenticity. “Consistency” indicates whether the result is consistent with control information (the contour or bounding box or key points). “Text” denotes whether the result matches the text description. “Authenticity” represents the level of the authenticity of all results.

Table 3. The internal quantitative comparison results of our methods in Oxford-102 flower dataset.

	ours	ours+VGG16	ours+VGG19
consistency	1.245	1.121	1.229
text	1.229	1.154	1.225
authenticity	1.282	1.153	1.169

Table 1 shows the comparison results with GAWWN in the CUB dataset. It is evident from the results that our method is considered to be the best in all respect. Our results have better authenticity and more conform to the text and control information.

We compare the internal quantitative results in birds and flower data sets because our method uses three contour extraction methods. The comparison results are shown in Tables 2 and 3. In Tables 2 and 3, among the birds' results, the overall authenticity of VGG19 is better than that of VGG16, while that of flowers is the opposite. The reason for this is that the proportion of birds in the image is relatively small (generally less than 50%), so the judgment of the authenticity of bird image is more dependent on the generation of bird details. VGG19 performs the best authenticity in generating bird results, which shows that it does best in detail generation. Compared with bird images, the proportion of flowers in the image is generally more than 80%, so its authenticity depends on the overall structure. In the authenticity of flower results, VGG16 is better than VGG19, which indicates that VGG16 performs the best in structural consistency. Although VGG19 can obtain pretty detailed information in flower results, the authenticity of VGG16 results is better because flowers pay more attention to integrity. VGG16 also showed the best structural consistency in birds results, indicating that VGG16 is indeed better than VGG19 in terms of structural consistency.

On the whole, VGG19 is better than VGG16 in detail synthesis, and VGG16 is better than VGG19 in overall structure synthesis. This is reasonable because VGG19 is deeper than VGG16 so that it can extract more detail-oriented feature information. The number of layers of VGG16 is relatively small, so it pays more attention to the overall feature information. VGG is a network structure specially designed for feature extraction, which performs well in classification, segmentation, and other tasks. Therefore, the use of VGG is better than the simple use of convolution operation (without VGG).

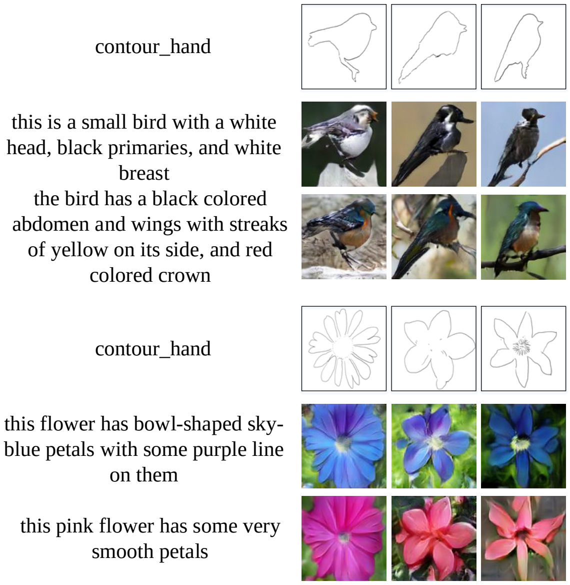
contour_hand
this is a small bird with a white head, black primaries, and white breast
the bird has a black colored abdomen and wings with streaks of yellow on its side, and red colored crown
contour_hand
this flower has bowl-shaped sky-blue petals with some purple line on them
this pink flower has some very smooth petals

Fig. 8. The text descriptions on the left are all artificial descriptions that do not exist in the dataset. The contours are also drawn manually.

4.4 Controllable image synthesis

The most important feature of our work is to realize fine-grained controllable image synthesis based on artificial hand drawing and manual text description. The relevant results are shown in Figure 8. Neither the contour nor the text description in the figure is artificial and does not exist in the dataset. These results reflect well the hand-drawn contour and artificial text description content, but also have a high degree of authenticity. This demonstrates the effectiveness of our method in synthesizing high-quality authentic images and shows the high flexibility and controllability of our method because all inputs can be controlled artificially.

5 Conclusion

In this work, we propose a customizable image synthesis based on contour and text descriptions. The high-quality image synthesis is achieved through adversarial learning. The synthesis results indicate that our method maintains the basic shape of the contour, while also conforms to the text description. Furthermore, we have evaluated the model on the Caltech-UCSD Birds dataset and the Oxford-102 flower dataset. The experimental results demonstrate the effectiveness and robustness of our method. Besides, the high-quality image synthesis results based on hand-drawn contour and artificial descriptions are also illustrated to prove that our method is highly controllable and flexible.

Acknowledges This research is supported by Sichuan Provincial Science and Technology Department(No. 2018GZ0517 2019YFS0146 2019YFS0155), National Natural Science Foundation of China(No. 61907009), Natural Science Foundation of Guangdong Province (No. 2018A030313802).

References

1. I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio.: Generative adversarial nets. In: Neural Information Processing Systems 27, 2672-2680 (2014)
2. P. Isola, J. Zhu, T. Zhou, and A. A. Efros.: Image-to-Image Translation with Conditional Adversarial Networks. In: Computer Vision and Pattern Recognition, 5967-5976 (2017)
3. J. Zhu, T. Park, P. Isola, and A. A. Efros.: Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. In: International Conference on Computer Vision, 2242-2251, (2017)
4. X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel.: InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets. In: Neural Information Processing Systems 29, 2172-2180, (2016)
5. M. Mirza, and S. Osindero.: Conditional generative adversarial nets. arXiv:1411.1784 (2014)

6. S. E. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee.: Generative adversarial text to image synthesis. In: International Conference on Machine Learning, 1060-1069 (2016)
7. C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie: The caltech-UCSD birds-200-2011 dataset. Technical report CNS-TR-2011-001, California Institute of Technology (2011)
8. M.-E. Nilsback, and A. Zisserman.: Automated flower classification over a large number of classes. In: Indian Conference on Computer Vision, Graphics and Image Processing, 722-729 (2008).
9. H. Zhang, T. Xu, H. Li, S. Zhang, X. Huang, X. Wang, and D. N. Metaxas.: StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks. In: International Conference on Computer Vision, 5908-5916 (2017)
10. H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas.: S-tackgan++: Realistic image synthesis with stacked generative adversarial networks. In: IEEE Trans. Pattern Anal. Mach. Intell., **41**(8), 1947–1962, (2019)
11. Z. Zhang, Y. Xie, and L. Yang.: Photographic text-to-image synthesis with a hierarchically-nested adversarial network. In: Computer Vision and Pattern Recognition, 6199-6208 (2018)
12. T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He.: Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In: Computer Vision and Pattern Recognition, 1316-1324 (2018)
13. T. Qiao, J. Zhang, D. Xu, and D. Tao.: Mirrorgan: Learning text-to-image generation by redescription. In: Computer Vision and Pattern Recognition, 1505-1514 (2019)
14. M. Zhu, P. Pan, W. Chen, and Y. Yang.: DM-GAN: dynamic memory generative adversarial networks for text-to-image synthesis. In: Computer Vision and Pattern Recognition, 5802-5810 (2019)
15. T. Qiao, J. Zhang, D. Xu, and D. Tao.: Learn, Imagine and Create: Text-to-Image Generation from Prior Knowledge. In: Neural Information Processing Systems 32, 885-895 (2019)
16. S. E. Reed, Z. Akata, S. Mohan, S. Tenka, B. Schiele, and H. Lee.: Learning what and where to draw. In: Neural Information Processing Systems 29, 885-895 (2016)
17. M. D. Zeiler, and R. Fergus.: Visualizing and understanding convolutional networks. In: European Conference on Compuer Vision, 818-833 (2014)
18. S. Ioffe, and C. Szegedy.: Batch normalization: accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning, 448C456 (2015)
19. S. E. Reed, Z. Akata, H. Lee, and B. Schiele.: Learning deep representations of fine-grained visual descriptions. In: Computer Vision and Pattern Recognition, 49-58 (2016)
20. K. He, X. Zhang, S. Ren, and J. Sun.: Deep residual learning for image recognition. In: Computer Vision and Pattern Recognition, 770-778 (2016)
21. B. Xu, N. Wang, T. Chen, and M. Li: Empirical evaluation of rectified activations in convolutional network. arXiv:1505.00853 (2015)
22. D. P. Kingma, and J. Ba.: Adam: a method for stochastic optimization. In: International Conference on Learning Representations (2015)
23. K. Simonyan, and A. Zisserma.: Very Deep Convolutional Networks for Large-Scale Image Recognition. In: International Conference on Learning Representations (2015)