Motion Image Deblurring using AS-Cycle Generative Adversarial Network

Xiaoming Zhu
Hunan Tobacco Company,
Changsha Hunan, China

Lijun Yao
Changzhutan Tobacco Logistics Co. Ltd. of Hunan Province,
ChangSha Hunan

Fan Luo
Hunan Tobacco Company,
Changsha Hunan, China

Kejun Wang
Changzhutan Tobacco Logistics Co. Ltd. of Hunan Province,
ChangSha Hunan

Zhou Che
Hunan Tobacco Company,
Changsha Hunan, China

Jing Yan
Hunan Tobacco Company,
Changsha Hunan, China

Min Zhou
Hunan Tobacco Company,
Changsha Hunan, China

Yongchang Cai
Hunan Tobacco Company,
Changsha Hunan, China

Lingling Wang
Hunan Tobacco Workers Training Center,
Hunan Xiangtan

Zelong Cao
Changzhutan Tobacco Logistics Co. Ltd. of Hunan Province,
ChangSha Hunan

Lan Peng
Changzhutan Tobacco Logistics Co. Ltd. of Hunan Province,
ChangSha Hunan

Fengqing Bai
Changzhutan Tobacco Logistics Co. Ltd. of Hunan Province,
ChangSha Hunan

Zifang You
Changzhutan Tobacco Logistics Co., Ltd. of Hunan Province,
ChangSha Hunan

Hongqiu Xiao
Changzhutan Tobacco Logistics Co. Ltd. of Hunan Province,
ChangSha Hunan

Haocheng Qi
College of Information Science and Engineering,
Hunan Normal University,
ChangSha Hunan, China

ABSTRACT
To improve the problem of poor generalization ability of image deblurring model in real scenes, this paper proposes a model named AS-CycleGAN (Cycle Generative Adversarial Network based on Asymmetric Samples). The model trains on unpaired images by using two "dual form" Conditional Generation Adversarial Networks, adopting global residual connection and ResNetv2 residual module. To enhance the texture effect, the SFT layer is integrated. The experimental results on the data set of Gopro show that the SSIM and PSNR values of our algorithm are 15.97% and 0.75% higher than those of the benchmark model CycleGAN, respectively. By improving the residual structure and adding the SFT layer, the effect is even better. AS-CycleGAN provides a powerful help to solve the motion blur problem in the actual scene.

General Terms
Image Processing, Deep Learning

Keywords
Motion image deblurring, cycle generative adversarial networks, unpaired data sets, residual network

1. INTRODUCTION
Due to many factors, such as light, shooting angle and object movement, moving images are more easily blurred. To recover high-quality clear images, image deblurring technology is needed. Whyte et al. [1] proposed a non-uniform blind deblurring algorithm. It is based on a geometrically consistent model of rotation speed during camera exposure to deal with non-uniform image blur. Yet, the effect is not excellent in the actual scene because of the excessive assumptions made in the modeling.

With the development of deep learning, many methods of image deblurring based on neural networks have been proposed. Chakrabarti et al. [2] proposed a method of using CNN to predict the Fourier coefficients of a deconvolution filter and deblurring in the frequency domain. Kupyn et al. [3] proposed the DeblurGAN model based on residual networks, it is able to deal with blurred images generated by objects moving at high speeds. Subsequently, Kupyn et al. [4] also proposed DeblurGANv2 which uses downsampling and Features Pyramid Network (FPN) model to deal with general image restoration. Tencent Youtu [5] proposed Scale-recurrent Network (SRN) based on Recurrent Neural Network (RNN). Dai et al. [6] proposed Deep Multi-Patch Hierarchical Network (DMPHN), this model uses residual learning method

to train more data and enhance migration ability. The above methods all use synthetic paired data sets for training, but synthetic paired data will not be available in the real scene for many tasks.

Mirza et al. [7] proposed Conditional Generative Adversarial Networks (CGAN), the core idea is to allow images to be transformed between different domains. Isola et al. [8] proposed pix2pix framework to provide a general framework for processing different image domain conversions. Zhu et al. [9] proposed CycleGAN to solve the problem of image translation on unpaired data. Such methods can use unpaired data sets for training to enhance the generalization ability of the model.

This paper reconstructs the image deblurring as a sample conversion between the blurred image domain and the clear image domain. This paper proposes a model named AS-CycleGAN which combines CGAN and image translation based on Generative Adversarial Networks (GAN) [10, 11]. To realize image conversion, AS-CycleGAN adopts WGAN-GP [12, 13] and two symmetric CGAN based on CycleGAN [11]. Meanwhile, the conversion of images between the blurred domain and clear domain by using a deep residual network structure. To improve the training speed and reducing the number of parameters, the optimized ResNetv2 is used as residual module instead of the ResNet [14]. To deal with the edges and textures of the restored image, the SFT layer [15] is integrated. To ensure the diversity of generated samples and the stability of training, the improved PatchGAN structure [16] is used for the discriminator. By improving the correlation structure and using real images to train the network model, a better deblurring effect is achieved in real scenes.

2. MODEL ARCHITECTURE

2.1 General structure of model

When the deblurring algorithm is trained on unpaired data sets, the content loss function cannot be directly calculated. The model cannot be trained because of the lack of blur-clear image pairs. Therefore, this paper adda a reverse GAN based on the CycleGAN [9] framework to remap the generated target domain images to the source domain. The computation of content loss is performed indirectly by constraining the distance between the reconstructed source domain images and the input source domain images. The overall framework of the model consists of two generative confrontation networks, as shown in Figure1.

To constrain the content of the generated samples, the training process consists of forward and reverse loops: the forward loop is blurry→clear→blurry, and the reverse loop is clear→blurry→clear, the two loop systems work together to achieve the constraint purpose. Specifically, a blurred image I_B passes through the generator G_A to generate a reconstructed clear image $I_{S\_restored}$, and then passes through G_B to get the reconstructed blurred image $I_{B\_restored}$. To constrain the contents of I_B and $I_{S\_restored}$ to be consistent by constraining I_B and $I_{B\_restored}$ as close as possible. Similarly, for the reverse loop, this paper indirectly constrains the I_S and $I_{B\_restored}$ by constraining the clear image I_S and the reconstructed clear image $I_{S\_restored}$ to be as close as possible to calculate the content loss under unpaired data. The loss function is usually called cyclic consistency loss, the structure is shown in the two boxes on the left and right of Figure 1

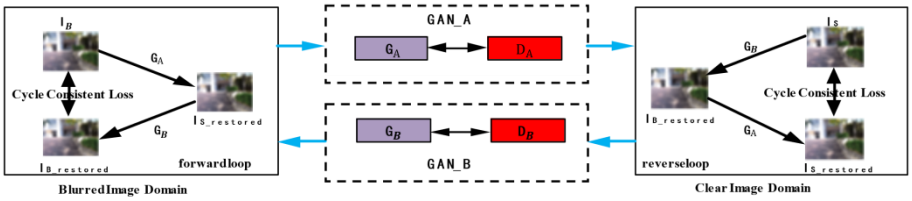

Fig 1: General structure of model

2.2 Generator

This paperincorporates deep residual network and global residual connection on the basis of GAN, connect the input and output of the network. The intermediate network layer only needs to learn the residual between output, input and hidden layers, which reduces the learning volume and training difficulty. The generated network (Figure2) consists of 2 convolutional layers, 9 residual modules and 2 deconvolutional layers. We adopt ReLu activation with the Tanh function at the end. In addition, we add a random deactivation layer (DropOut) and an instance normalization layer (IN). The residual module uses ResNetv2 to realize input and output direct connection. The activation function is placed on the branch for residuals. Each cell is before the affine transformation, so that the information propagates faster in the reverse and forward propagation and prevents the vanishing gradient, as shown in the dashed box below. By using the DropOut layer, the model increases the generated sample diversity, prevents overfitting, and improves the generalization ability. The network structure of generators G_A and G_B are shown in Figure2.

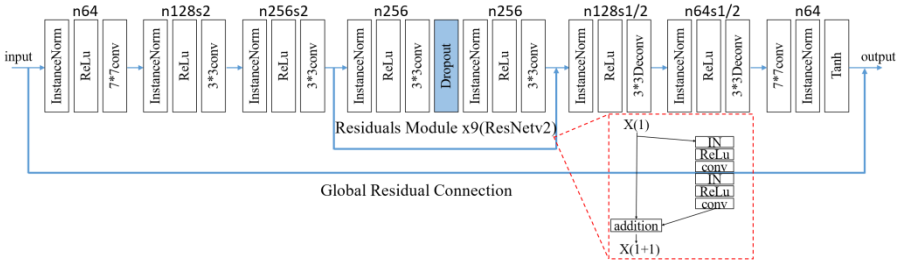

Fig 2: Generators G_A and G_B

2.3 Discriminator

The discriminator is based on PatchGAN [16]. the original Batch Normalization (BN) layer is replaced with an Instance Normalization (IN) layer. Meanwhile, the loss function in the original GAN is replaced with the RaGAN-LS [4] loss function. Accordingly, the Sigmoid function activation layer is removed from the original PatchGAN. The discriminant network contains five convolutional layers with the Leaky Relu function as activation function. The discriminators D_A and D_B are shown in Figure3.

Fig 3: Discriminators D_A and D_B

2.4 SFTGAN

The goal of Spatial Feature Transform Generative Adversarial Networks (SFTGAN) [15] is to recover natural and realistic textures in super-resolution results. This paper proposes a spatial feature modulation (SFT) layer to combine the semantic category prior to the network. The SFT layer is based on the parameters of affine transformation and translation obtained from the prior, then the affine transformation operation is performed on the intermediate features. The loss function contains perceptual loss and adversarial loss. The two inputs are the low-resolution image and the segmentation semantic map. The segmentation semantic map is passed through the Condition Network to generate the Conditions Feature Map, which is shared by each layer, but the SFT layer is not shared. Every other Conv layer has a SFT layer corresponding to the conditions. The SFTGAN super-resolution reconstruction network [15] is incorporated in the GAN_A and GAN_B to enhance the edge and texture effects of the images.The SFT layer is added to the GAN structure of ResNetv2. The Conditional Normalization (CN) is to use a function learned under certain conditions, which replaces the affine transformation in the original BN. The specific implementation process of the network is shown in Figure4, the SFT layer is shown in the dashed box.

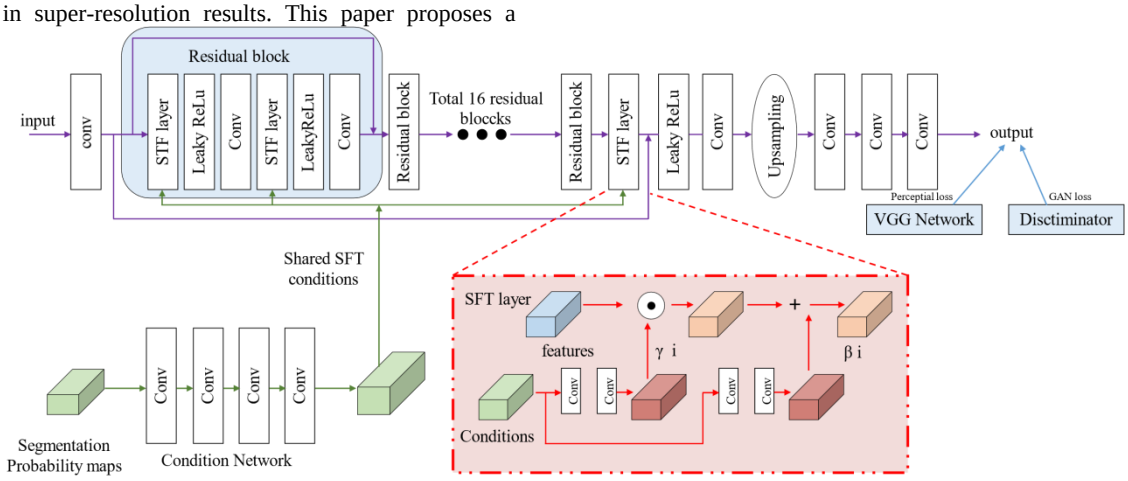

Fig 4: SFTGAN structure

3. LOSS FUNCTION

Overall loss function: The overall loss function includes the adversarial loss and the content loss. In this paper, X denotes the samples in the clear image domain, Y denotes the samples in the blurred image domain, and N denotes the number of samples:

$$L(G_A, G_B, D_A, D_B, X, Y) = L_{GAN}(G_A, D_A, X, Y) + L_{GAN}(G_B, D_B, Y, X) + \lambda_1 L_{cyc_{perception}}(G_A, G_B, X, Y) + \lambda_2 L_{identity}(G_A, G_B, Y, X) \quad (1)$$

In the above equation, λ_1 refers to the weight of the cyclic consistency loss and λ_2 refers to the weight of the same mapping loss. Both generators G_A and G_B need to minimize the above equation, while both discriminators D_A and D_B need to maximize the above equation. The final optimal generators obtained are shown in the following equation:

$$G^*_A, G^*_B = argmin_{G_A,G_B} max_{D_A,D_B} L(G_A, G_B, D_A, D_B, X, Y) \quad (2)$$

Adversarial Loss: This paper adopts the adversarial loss function of the two CGAN as adversarial loss. The role of G_A is to make the generated image as natural and clear as possible, close to the original image. G_B is to make the motion blur in the generated image exist as much as possible in the real scene. Because WGAN [12] has the problems of slow convergence and difficult training in a real experimental environment, the improved WGAN-GP [15, 16] is used in this paper. WGAN-GP satisfies the Lipschitz continuum condition by adding a gradient penalty term, and directly adopts "Weight Clipping" when dealing with the Lipschitz restriction to stabilize the training effect. The RaGAN-LS loss [4] is used as the adversarial loss function. The specific adversarial loss calculation is shown in equation (3)(4) below, and the penalty term is omitted here for the convenience of writing.

$$L_{GAN}(G_A, D_A, X, Y) = \frac{1}{N}\sum_{n=1}^{N}[D_A(Y) - D_A(G_A(X))] \quad (3)$$

$$L_{GAN}(G_B, D_B, Y, X) = \frac{1}{N}\sum_{n=1}^{N}[D_B(X) - D_B(G_B(Y))] \quad (4)$$

Content loss: The content loss guides the generator to generate images with appropriate content. This paper uses cyclic consistency loss and the same mapping loss. The cyclic consistency loss is the core part of unpaired data set training. It mainly calculates the loss value between the value of the sample after forward and reverse loops and the value of the original sample. The content loss includes the following main contents:

a) Cycle Consistent Loss: The circular consistency loss function adopts the form of perceptual loss. Com- pared with other forms of loss functions, perceptual loss is closely related to human senses and is closer to the visual perception of human eyes. By constraining the perceptual loss function, the

generated images are more natural and realistic afterwards. Perceptual loss is generally difficult to calculate directly and requires additional help from a pre-trained VGG network. The feature map output from the seventh convolutional layer, which is the conv3_3 layer in the VGG-19 network, is selected to calculate the perceptual loss. As shown in equation (5), where ∅ denotes the output feature map of the seventh convolutional layer of VGG-19 is taken, while C, H, and W denote the number of channels, height, and width of the feature map.

$$L_{cyc\_perception}(G_A, G_B, X, Y) = \frac{1}{N}\sum_{n=1}^{N}\left[\frac{1}{CHW}\left\|\emptyset(G_B(G_A(X))) - \emptyset(X)\right\|_2^2 + \frac{1}{CHW}\left\|\emptyset(G_A(G_B(Y))) - \emptyset(Y)\right\|_2^2\right] \quad (5)$$

b) Same mapping loss: The same mapping loss [17] complements the cyclic consistency loss by further constraining the content consistency of the generated and input images based on the perceptual loss. The same mapping loss requires that the samples in the target domain do not change after passing through the generator, thus facilitating the full and complete "transformation" of the samples in the source domain by the generator. The same mapping loss is sensitive to color, which can effectively ensure the consistency of color style between the input image and the generated image to avoid color difference. The L1 distance is used to calculate the same mapping loss, and the calculation procedure is shown in the left and right boxes in Figure1 above, where I_S is mapped to $I_{S\_restored}$ through G_A and I_B is mapped to $I_{B\_restored}$ through G_B. The expressions are as follows:

$$L_{identity}(G_A, G_B, Y, X) = \frac{1}{N}\sum_{n=1}^{N}[||G_A(Y) - Y||_1 + ||G_B(X) - X||_1] \quad (6)$$

4. EXPERIMENT ANALYSIS

4.1 Experimental environment configuration and data preprocessing

The experimental environment in this paper uses a 1660Ti6G discrete graphics card, a hexa-core CPU (Intel core i7-9750H) at 2.6GHz, and 16GB RAM. The operating system is Win10 and Ubuntu-Server linux16.04, and the deep learning framework is TensorFlow-GPU 13.1 and PyTorch 1.2.0.

In this paper, only the blurred images of data set of Gopro [18] is used for training, the corresponding clear images are removed, so this paper uses the aforementioned method on the unpaired dataset for training process. In the testing phase, the quantitative evaluation metrics can be calculated using the information from the paired dataset. It is also convenient to compare the algorithms from other paired datasets, so as to the effect of model restoration can be evaluated objectively. We need to pre-process the image size, bit depth, and channel values according to the experimental requirements. In thispaper, the number of samples N is taken as 60, and the loss functionλ1is taken as 10 andλ2is taken as 0.5. This paper uses peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) as the evaluation criteria. PSNR is widely used for objective evaluation of images, expressed as:

$$f_{MSE} = \frac{\sum_0^M i \sum_0^M j (f_{ij} - g_{ij})}{M \times N} \quad (7)$$

$$f_{PSNR} = 10 \times \lg\left(\frac{(2^{bits}-1)^2}{f_{MSE}}\right) \quad (8)$$

SSIM mainly compares the structural information of different images, expressed as:

$$f_{SSIM}(f_{ij}, g_{ij}) = \frac{(2\mu_f\mu_g + c_1)(2\sigma_{fg} + c_2)}{(\mu_f^2 + \mu_g^2 + c_1)(\sigma_f^2 + \sigma_g^2 + c_2)} \quad (9)$$

4.2 Analysis of experimental results

As can be seen from Figure5, the algorithm achieved good results on the data set of Gopro [18], the image restoration is realistic and natural, basically retaining the structural features of the original image without obvious chromatic aberration and local distortion. The slight shortcomings are the lack of edge and texture effects and poor restoration of parts with serious blurring. According to the box marks in the Figure5, the generated images using the improved ResNetv2 residual module model are more detailed and the texture effect is better than that of the generated images using the ResNet model.

Fig 5: Effect of different network structures on experimental results. Blurred images−first column, AS-CycleGAN (ResNet) −second column, AS-CycleGAN (ResNetv2) -third column, Clear Image −last column.

Our model is trained based on the method of unpaired datasets, as in Figure6, the results of deblurring in the Gopro dataset have been basically close to the model DeblurGANv2 [4] algorithm, especially in images with relatively low blurring degree. By improving the residual module, the deblurring ability has been enhanced. There is no obvious chromatic aberration and distortion, it also improves the unnatural visual sensory effectssuch as over-sharpening, grid effect and edge distortion of the generated images, the reconstructed images are natural and realistic, specifically from the detailed information of the boxes and elliptical boxes in the Figure6 can be more clearly seen the change. After integrating the SFT layer into the original structure, the image details and texture effects are improved, as shown in the third row of Figure6. The composition of the car surface, tree form, stone bench appearance, slogan content and pedestrian posture are more detailed, the details of the edges and textures are handled more effectively and the visual effects are more natural, especially the effect is more obvious when the picture is enlarged, as shown in Figure7.

The results of this paper proposed method and other methods are shown in Table1, where SSIM and PSNR are taken as the mean values of multiple experiments. DeblurGAN(Wild) refers to the results obtained by training the DeblurGAN [3] model on the Gopro dataset, while DeblurGAN(Comb) refers to the results obtained by mixing the synthetic dataset proposed by Kupyn et al. with the Gopro dataset in a 2:1 ratio and training the model on this mixed dataset. In addition, this

paper also selects DeblurGANv2 [4] algorithm model for comparison.

Fig 6: Experimental results of the model in this paper compared with other models. Blurred images−top row, DeblurGAN −second row, AS-CycleGAN(Add STF layer)−third row, Clear Image− bottom row.

Fig 7: Partial effect diagram of incorporating SFTGAN structure. Without SFTGAN−top row, Integration SFTGAN−bottom row.

From Table1, the PSNR and SSIM in the results were significantly improved compared to the benchmark model CycleGAN. The improved ResNetv2 has improved by 18.91% and 0.75%, respectively. The results are even better after incorporating the SFT layer, with 21.01% and 1.08% improvement in the two metrics, respectively. Compared with the well-known DeblurGAN algorithm on paired datasets, the method in this paper achieves experimental results similar to them. The PSNR value exceeds that of DeblurGAN (Wild) and is basically close to other algorithms. The SSIM value only differs from DeblurGAN by about 0.2 and is slightlyhigher than that of DeblurGANv2 algorithm. The ResNetv2 model with the improved residual module and the model with the SFT layer performs even better. They are basically close to or even better than other algorithms on paired datasets. SSIM values are also higher than those of some algorithms on paired datasets

Table 1. Comparison of the effects of the model in this paper and other models

	Model	PSNR	SSIM	Time
Unpaired data sets	CycleGAN	23.8	0.928	1.73s
	ResNet(AS-CycleGAN)	27.6	0.935	0.92s
	ResNetv2（AS-CycleGAN）	28.3	0.935	0.96s
	Add STF layer（AS-CycleGAN）	28.8	0.938	1.04s
Paired data sets	DeblurGAN(Wild)	27.2	0.954	0.85s
	DeblurGAN(Comb)	28.7	0.958	0.87s
	DeblurGANv2	29.5	0.934	0.35s

5. CONCLUSION

This paper proposes an algorithmic model trained with real images on unpaired datasets. This paper adopts the WGAN-GP, in which global residual connectivity is used in the generator, as well as a modified ResNetv2 residual module. In addition, the super-resolution repair network SFTGAN is incorporated into the structure to improve the edge and texture effects. PatchGAN is used as the discriminator, and its parameters were improved. Finally, the loss function is determined to accord to the adversarial loss and content loss to build the model to constrain the whole training process to achieve the comprehensive improvement of the deblurring effect. Experiments show that the results of this paper are significantly better than the benchmark model CycleGAN, with the PSNR and SSIM values improving by 15.97% and 0.75%, respectively, by improving the residual structure and adding of SFT layer, the results are better. This paper achieves experimental results similar to or even superior to other algorithmic models on paired datasets. To a certain extent, the problem of poor generalization ability of the model in real scenes is solved.

6. ACKNOWLEDGMENTS

This work was supported by Research Foundation of Hunan Provincial Tobacco Company’s Science and Technology Project 2020 (20-21D02).

7. REFERENCES

[1] Oliver, et al. "Non-uniform Deblurring for Shaken Images." *International Journal of Computer Vision* vol. 98, no. 2, pp. 168-186, 2012.

[2] Chakrabarti, A., "A Neural Approach to Blind Motion Deblurring." *European Conference on Computer Vision Springer*, Cham, pp. 221-235, 2016.

[3] Kupyn, O., et al. "DeblurGAN: Blind Motion Deblurring Using Conditional Adversarial Networks." 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8183-8192, 2018.

[4] Kupyn, O., et al. "DeblurGAN-v2: Deblurring (Orders-of-Magnitude) Faster and Better." 2019 IEEE/CVF International Conference on Computer Vision (ICCV) IEEE, pp. 00897: 8877-8886, 2019.

[5] Tao, X., et al. "Scale-recurrent Network for Deep Image Deblurring." 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition IEEE, pp. 8174-8182, 2018.

[6] Zhang, H. , et al. "Deep Stacked Hierarchical Multi-patch Network for Image Deblurring." 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) IEEE, pp. 5971-5979, 2019.

[7] Mirza, M., and S. Osindero. "Conditional Generative Adversarial Nets." Computer Science (2014), pp. 2672-2680, 2014.

[8] Isola, P., et al. "Image-to-Image Translation with Conditional Adversarial Networks." IEEE Conference on Computer Vision & Pattern Recognition IEEE, pp. 1125-1134, 2016.

[9] Zhu, J. Y., et al. "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks." IEEE, pp. 2223-2232, 2017

[10] Cao, Y., et al. "Review of computer vision based on generative adversarial networks." Journal of Image and Graphics, pp. 1433-1449, 2018.

[11] Wang, K. F., et al. "Generative Adversarial Networks: The State of the Art and Beyond." Acta Automatica Sinica, pp. 321-332, 2017.

[12] Arjovsky M, Chintala S, Bottou L. "Wasserstein generative adversarial networks.". In: International Conference on Machine Learning, pp. 214-223 2017.

[13] Gulrajani I, et al. "Improved training of Wasserstein gans." In: Advances in Neural Information Processing Systems, pp. 5767-5777, 2017.

[14] He K, et al. "Deep residual learning for image recognition." In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778, 2016.

[15] Wang Xintao, et al. "Recovering Realistic Texture in Image Super-Resolution by Deep Spatial Feature Transform." CVPR, pp. 606-615, 2018.

[16] Zhang Yaliang, et al. "Free-Form Video Inpainting With 3D Gated Convolution and Temporal PatchGAN." ICCV, pp. 9065-9074 2019.

[17] Taigman Y, et al. "Unsupervised cross-domain image Generation." In: International Conference on Learning Representations. 2016.

[18] Nah S, et al. "Deep Multi-scale Convolutional Neural Network for Dynamic Scene Deblurring." CVPR, pp. 257-265 2017.