Batik-DG: Improved DeblurGAN for Batik Crack Pattern Generation

Yiyang Huang1, Jianxia Su1,*, Jianming Wang2 and Shuaifei Ji3

1School of Information Engineering, Beijing Institution of Fashion Technology, Beijing, China
2School of Materials Design and Engineering, Beijing Institution of Fashion Technology, Beijing, China
3Chinese Fashion Science & Technology Research Institute, Beijing, China

*Corresponding author

Abstract. As a traditional handcraft in China, batik is valuable for both decoration and fashion design due to its unique crack pattern. However, previous researches on batik crack pattern generation remain space in promoting verisimilitude. In recent years, Generative Adversarial Networks (GAN) show great ability in realistic image generation, and DeblurGAN performs well in image deblurring task, showing good potentiality in image generation. In this paper, we improve DeblurGAN for batik crack pattern generation by several methods. First, transposed convolution is replaced with resize-convolution. Second, Hybrid Dilated Convolution (HDC) is added to the network. Third, RaLSGAN is utilized instead of WGAN-GP, and several additional loss functions such as L1, L2 and so forth are tested respectively. Our network is trained on BIFT-Batik dataset and performs well to some degrees both in visual verisimilitude and IQA indexes including PSNR and SSIM. We name our network as Batik-DG (Batik-DeblurGAN).

1. Introduction

Batik is a traditional handcraft and intangible cultural heritage in China. This handcraft covers wax onto the cloth to isolate the dye and thus produce fabulous patterns. Since the wax is fragile, the dye will penetrate through the cracks onto the cloth surface, forming unique crack patterns. Using computer technology to generate batik pattern can help designers to quickly create works with batik style, which is conducive to the spread and development of batik.

DeblurGAN [1] is a kind of generative adversarial networks [2] designed for image deblurring. Basing on DCGAN and pix2pix [3], DeblurGAN uptakes the advantages of many researches including residual blocks, perceptual loss, instance normalization and WGAN-GP, and finally runs 5 times faster than Deep-Deblur, the main counterpart in image deblurring task.
We notice that both image-to-image generation and image deblurring can be regarded as the operation of adding additional information into source images. For this reason, networks for image deblurring may work well in image-to-image generation. However, we found distinct chromatic aberration caused by checkerboard artifacts when applying DeblurGAN to batik crack pattern generation task, and the crack pattern is not long and not clear enough (Figure 2). So, it is necessary to make some targeted and essential improvement on DeblurGAN for batik crack pattern generation.

In this paper, we first introduce related work in section 2, then propose several methods to improve DeblurGAN in section 3 and show the results we get in batik crack pattern generation in section 4, and finally make the conclusion in section 5. The fully improved network is named as Batik-DG (Batik-DeblurGAN).

2. Related Work

2.1. Generative Adversarial Networks
Generative adversarial networks design a min-max game between generator $G$ and discriminator $D$ to help generator learn better than normal neural networks. The discriminator $D$ is trained to distinguish fake data from the real one, and the generator $G$ is trained to fool discriminator by generating better fake data. The training process of GAN can be described as equation (1).

$$\min_{G} \max_{D} V(D, G) = \mathbb{E} [\log D(x)] + \mathbb{E} [\log (1 - D(z))]$$

2.2. DeblurGAN
DeblurGAN has good performance in image deblurring. It is developed from the CNN designed by Johnson et al [4] and adds global skip connection in its generator (see Figure 3, in which $c3s2n256$ means a 3×3 convolution layer with stride $s = 2$ and channel numbers $n = 256$), so the network only need to learn a residual correction to the blurred images. In DeblurGAN, transposed convolution layers are used for upsampling and WGAN-GP instead of Standard GAN is used as the adversarial loss. DeblurGAN also use MSE of feature maps from layer $relu3_3$ in pretrained VGG19 as content loss, which belongs to perceptual loss.
2.3. **Resize-convolution**

Transposed convolution will lead to checkerboard artifacts in upsampling tasks, especially when the kernel size is not divisible by the stride. According to Odena [5], the checkerboard artifacts is avoidable by replacing transposed convolution with resize-convolution, which means resizing the input feature image first and then carrying out the convolution. Interpolation algorithms such as nearest and bilinear can be used in resizing, and Odena’s research indicates that nearest interpolation has better performance.

2.4. **Dilated Convolution and HDC**

Dilated convolution is a useful method to expand the receptive-field of network by separating cells among the kernel in convolution layers. However, dilated convolution may introduce gridding artifacts. One effective solution is HDC (Hybrid Dilated Convolution) [6], which means a group of dilated convolution layers with different dilations.

2.5. **Relativistic GAN**

| Type       | Loss Function                                                                                     |
|------------|--------------------------------------------------------------------------------------------------|
| RSGAN      | \[ L_D = -\mathbb{E} \left[ \log \left( \text{sigmoid} \left( C(x_r) - C(x_f) \right) \right) \right] \]  
|            | \[ L_G = -\mathbb{E} \left[ \log \left( \text{sigmoid} \left( C(x_f) - C(x_r) \right) \right) \right] \]  |
| RaSGAN     | \[ L_D = -\mathbb{E} \left[ \log \left( \text{sigmoid} \left( C(x_r) - \mathbb{E} C(x_f) \right) \right) \right] - \mathbb{E} \left[ \log \left( 1 - \text{sigmoid} \left( C(x_f) - \mathbb{E} C(x_r) \right) \right) \right] \]  
|            | \[ L_G = -\mathbb{E} \left[ \log \left( \text{sigmoid} \left( C(x_f) - \mathbb{E} C(x_r) \right) \right) \right] - \mathbb{E} \left[ \log \left( 1 - \text{sigmoid} \left( C(x_r) - \mathbb{E} C(x_f) \right) \right) \right] \]  |
| RaLSGAN    | \[ L_D = \mathbb{E} \left[ \left( C(x_r) - \mathbb{E} C(x_f) - 1 \right)^2 \right] + \mathbb{E} \left[ \left( C(x_f) - \mathbb{E} C(x_r) + 1 \right)^2 \right] \]  
|            | \[ L_G = \mathbb{E} \left[ \left( C(x_f) - \mathbb{E} C(x_r) - 1 \right)^2 \right] + \mathbb{E} \left[ \left( C(x_r) - \mathbb{E} C(x_f) + 1 \right)^2 \right] \]  |

Note: \( D(x) = \text{sigmoid}(C(x)) \), \( x_r \) refers to real data, \( x_f \) refers to fake data

In Standard GAN, the generator \( G \) is determined to fool the discriminator \( D \) by increase the probability that fake data is real. However, researches from Jolicoeur-Martineau [7] indicate that when training the generator \( G \), simultaneously decreasing the probability that real data is real can promote the performance of the network. This idea resulted in RGAN (Relativistic GAN) for original data and RaGAN (Relativistic average GAN) for averaged data. Furthermore, the relativistic method can also be applied to LSGAN. **Table 1** shows the loss functions of the relativistic GANs mentioned above.

3. **Our Method**

3.1. **Generator**

Our generator is developed from DeblurGAN with two modifications: replacing transposed convolution with resize-convolution and adding HDC(1,2,5) in front of residual blocks. The resize-convolution is aimed at eliminating checkerboard artifacts. And as for HDC(1,2,5), this method can enhance the perception ability of generator by providing \( 17 \times 17 \) receptive-field (**Figure 4**), which is equivalent to 8 layers of \( 3 \times 3 \) convolution. The generator’s architecture is shown in **Figure 5**.
3.2. Loss Function
According to Kupyn et al [1], L1 and L2 loss may introduce blurry artifacts into output images. However, Isola et al [3] pointed out that L1 loss can make results clear, and further research on DeblurGAN-v2 from Kupyn et al [8] shows that combining perceptual loss with L2 loss can both correct the chromatic aberration and improve the image quality. Meanwhile, loss functions such as smoothL1, SSIM and MS-SSIM-L1[9] may also work in optimizing the results. In consequence, we decide to set loss function as equation (2), in which $L_p$ refers to perceptual loss and $L_x$ refers to the loss function mentioned above (L1, L2, smoothL1, SSIM and MS-SSIM-L1).

We also substitute adversarial loss from WGAN-GP to RaLSGAN so as to get better results.

$$L = L_{RaLSGAN} + \lambda_p L_p + \lambda_x L_x$$  \hspace{1cm} (2)

4. Experiments

4.1. BIFT-Batik Dataset
We build our own dataset for this paper which is named as BIFT-Batik. This dataset contains 240 pairs of images of size $512 \times 512$, including both batik images and annotation images. In the annotation images, dark blue (RGB #202040) refers to dyed area and white color (RGB #FFFFFF) refers to pattern area. In experiments, we crop the images randomly to size $256 \times 256$ so as to facilitate processing.
Table 2. Detail information about BIFT-Batik dataset.

| Set    | Number | Patterns                      |
|--------|--------|-------------------------------|
| train  | 200    | flower, bird, fish, copper drum|
| test   | 40     | worm, butterfly, pomegranate, vortex |

4.2. Training Details
We train the improved editions of DeblurGAN with perceptual loss index $\lambda_p = 0.6$, additional loss index $\lambda_x = 50$, dropout ratio $p = 0.5$, leaning rate $\alpha = 10^{-4}$. We use Adam as the optimizer and train each model for 300 epoches. We fix the learning rate for the first 150 epochs and linearly decay the rate to zero.
All our experiments are carried out in PyTorch on one Nvidia Tesla P40.

4.3. Results

Figure 7. Results of resize-convolution with different interpolation algorithms.
As for resize-convolution layers, three kinds of interpolating algorithm are used, which include the nearest, bilinear and bicubic. We trained networks with these algorithms respectively. As shown in Figure 7, these three methods give similar visual results with no checkerboard artifacts and obvious chromatic aberration, and perform differently in IQA. The nearest interpolation reaches the best among three algorithms, as Table 3 indicates. So, we use resize-convolution with nearest interpolation in subsequent experiments.
The results of our networks with HDC(1,2,5) is shown in Figure 8, which shows that HDC(1,2,5) obviously helps the network to generate better image with longer batik crack pattern. In addition, Figure 9 shows the images of each channel, which indicates that HDC(1,2,5) can effectively avoid gridding artifacts.

Table 3. IQA of resize-convolution with different interpolation algorithms.

| Interpolation Algorithm | $PSNR_{train}$ | $SSIM_{train}$ | $PSNR_{test}$ | $SSIM_{test}$ |
|-------------------------|----------------|----------------|---------------|---------------|
| nearest                 | 20.50          | 0.8603         | 18.61         | 0.7892        |
| bilinear                | 19.51          | 0.8457         | 17.88         | 0.7791        |
| bicubic                 | 19.25          | 0.8402         | 17.63         | 0.7689        |

Figure 8. Results before and after using HDC(1,2,5).

Figure 9. Images of each channel (RGB) with HDC(1,2,5).
Figure 10 reveals the performance of WGAN-GP and RaLSGAN on DeblurGAN with resize-convolution and HDC(1,2,5). RaLSGAN shows quite well performance in the results.

![Input](image1) ![WGAN-GP](image2) ![RaLSGAN](image3) ![Ground Truth](image4)

Figure 10. Results obtained by using WGAN-GP or RaLSGAN.

![None](image5) ![L1](image6) ![L2](image7) ![smoothL1](image8) ![SSIM](image9) ![MS-SSIM-L1](image10)

Figure 11. Results of Batik-DG with different additional loss functions.

We tested L1, L2, smoothL1, SSIM and MS-SSIM-L1 as additional loss functions and the results are shown in Figure 11, which prove that all the additional loss functions can effectively eliminate the chromatic aberration. L1, L2 and SSIM all achieve good visual performance in batik crack pattern generation while SSIM results in longest crack. Meanwhile, L1 ranks as the best in IQA (Table 4).

| Additional Loss Function | $PSNR_{train}$ | $SSIM_{train}$ | $PSNR_{test}$ | $SSIM_{test}$ |
|--------------------------|----------------|----------------|----------------|----------------|
| (None)                   | 18.30          | 0.8240         | 16.53          | 0.7430         |
| L1                       | **22.26**      | 0.8770         | **19.73**      | **0.7993**     |
| L2                       | 21.33          | 0.8634         | 19.18          | 0.7886         |
| smoothL1                 | 20.82          | 0.8515         | 18.65          | 0.7694         |
| SSIM                     | 21.24          | 0.8776         | 18.55          | 0.7697         |
| MS-SSIM-L1               | 19.83          | 0.8299         | 18.35          | 0.7681         |

Table 4. IQA of different additional loss functions.

5. Conclusion
We make targeted improvements on DeblurGAN for batik crack pattern generation and get good results to some degrees. Our work confirms that resize-convolution is capable of preventing checkerboard artifacts, and HDC(1,2,5) can promote the results without causing gridding artifacts. In addition, RaLSGAN performs quite well in batik crack pattern generation. We also find L1 loss and SSIM loss can lead to fine results respectively in IQA and visual effects.

We have to point out that our network, Batik-DG, is slightly unsatisfactory in some aspects such as IQA, and could be further improved in subsequent researches.

Acknowledgement
Our work is supported by Open Research Project of Beijing Key Laboratory of Cloth Material Research & Evaluation, Discipline Construction Projects of Beijing Institute of Fashion Technology (Grant No. 120301990125)

References
[1] Kupyn O et al 2018 Deblurgan: Blind motion deblurring using conditional adversarial networks Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition pp 8183-8192
[2] Goodfellow I et al 2014 Generative adversarial nets Advances in neural information processing systems pp 2672-2680
[3] Isola P et al 2017 Image-to-image translation with conditional adversarial networks Proceedings of the IEEE conference on computer vision and pattern recognition pp 1125-1134
[4] Johnson J et al Perceptual losses for real-time style transfer and super-resolution European conference on computer vision pp 694-711
[5] Odena A et al 2016 Deconvolution and checkerboard artifacts Distill 1(10) e3
[6] Yu F et al 2015 Multi-scale context aggregation by dilated convolutions arXiv preprint arXiv:1511.07122
[7] Jolicoeur-Martineau A 2018 The relativistic discriminator: a key element missing from standard GAN arXiv preprint arXiv:1807.00734.
[8] Kupyn O et al 2019 DeblurGAN-v2: Deblurring (Orders-of-Magnitude) Faster and Better Proceedings of the IEEE International Conference on Computer Vision pp 8878-8887
[9] Zhao H 2016 Loss functions for image restoration with neural networks IEEE Transactions on Computational Imaging 3(1) 47-57