International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:13, No:11, 2019

The Layout Analysis of Handwriting Characters and the Fusion of Multi-style Ancient Books' Background

Yaolin Tian, Shanxiong Chen, Fujia Zhao, Xiaoyu Lin, Hailing Xiong

Abstract**—**Ancient books are significant culture inheritors and their background textures convey the potential history information. However, multi-style texture recovery of ancient books has received little attention. Restricted by insufficient ancient textures and complex handling process, the generation of ancient textures confronts with new challenges. For instance, training without sufficient data usually brings about overfitting or mode collapse, so some of the outputs are prone to be fake. Recently, image generation and style transfer based on deep learning are widely applied in computer vision. Breakthroughs within the field make it possible to conduct research upon multi-style texture recovery of ancient books. Under the circumstances, we proposed a network of layout analysis and image fusion system. Firstly, we trained models by using Deep Convolution Generative against Networks (DCGAN) to synthesize multi-style ancient textures; then, we analyzed layouts based on the Position Rearrangement (PR) algorithm that we proposed to adjust the layout structure of foreground content; at last, we realized our goal by fusing rearranged foreground texts and generated background. In experiments, diversified samples such as ancient Yi, Jurchen, Seal were selected as our training sets. Then, the performances of different fine-turning models were gradually improved by adjusting DCGAN model in parameters as well as structures. In order to evaluate the results scientifically, cross entropy loss function and Fréchet Inception Distance (FID) are selected to be our assessment criteria. Eventually, we got model M8 with lowest FID score. Compared with DCGAN model proposed by Radford at el., the FID score of M8 improved by 19.26%, enhancing the quality of the synthetic images profoundly.

Keywords**—**Deep learning, image fusion, image generation, layout analysis, object detection.

I. Introduction

CHINA is an ancient civilization country which has numerous literatures and classics during its long history. According to statistics [22], there are over 30 million ancient books in China, among which over 2.5 million are ancient books. They not only fully demonstrated the wisdom of the Chinese nation before thousands of years, but also are the cultural heritages of all human beings. Therefore, the preservation and digitization of ancient books become historically significant in the aspects of cultural research and cultural inheritance. However, these non-renewable treasures have been disappearing at a rapid rate, and they are now facing unprecedentedly severe challenges.

On the one hand, large number of scattered ancient books and documents has been damaged by mildew, adhesion, mutilation, word elimination and blackening because of poor collection conditions, thus, many important information has disappeared. On the other hand, it is difficult to avoid the influence of uneven lighting affected by the scanning environment and acquisition equipment, which greatly reduces the legibility of digitized images in the digitization process. Therefore, how to scientifically retain the useful information in ancient books has gradually become a focus. With the further development of computer vision, applying state-of-the-art techniques and approaches when digitalizing ancient books is expected to further improve the digitalization level.

Normally, digitalization of ancient books includes several steps such as the collection, arrangement and restoration of ancient books. The digitalization of ancient books is an interdisciplinary protection method, thus, it requires interdisciplinary talents who are proficient in relevant languages and computer technology. In order to analyze the information in collected ancient books, it is necessary to use layout analysis technology to extract the content of the page. As for ancient books restoration, one of our main tasks is to restore the image information and to generate backgrounds of its style.

By research, we draw a conclusion that there have been few studies [1]-[7] upon the combination of layout analysis technology and image synthesis technology. Meanwhile, as an essential part of ancient books digitalization process, multi-style texture recovery would not be carried out without two techniques mentioned above. The main factors hindering the development of this work are as follows: (1) Ancient books are commonly without complete background, which is caused by natural factors and human factors; (2) Image synthesis based on deep learning requires a large number and variety of materials of ancient books to support study. Therefore, multi-style texture recovery of ancient books cannot be carried out smoothly. Under the urgency of the research need, this paper has proposed a practical method to solve it. Through long-term data collection and manual sorting, we got normative data samples in ancient Yi, ancient Chinese (seal of Qin Dynasty), Jurchen and ancient drawings, shown in Fig. 1. We took generative against networks (GANs) and layout analysis as key technologies to fuse the rearranged foreground texts and multi-style background textures. Finally, we achieved the goal of multi-style texture recovery of ancient books.

Yaolin Tian and Hailing Xiong are with the College of Business, Southwest University, Chongqing, 402460 China and with the College of Computer and Information Science, Southwest University, Chongqing 400715 China.

Shanxiong Chen*, Fujia Zhao, and Xiaoyu Lin are with the College of Computer and Information Science, Southwest University, Chongqing 400715 China (*Corresponding author; e-mail: csxpml@163.com).

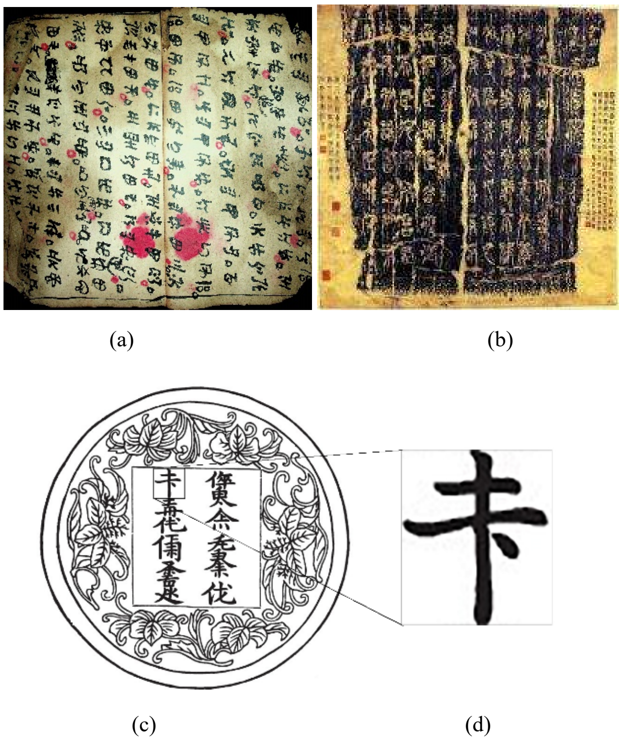


Fig. 1 Datasets samples: We collected several ancient scripts including Yi, ancient Chinese (seal) and Jurchen separately in (a), (b), (c). The segmented Jurchen word in (d) indicates these ancient scripts have a same font feature of quadrate structure. (a) Yi Script; (b) Seal Script; (c) scanned Jurchen Script without background texture; (d) A Jurchen script means "four"

II. Related Work

A. Layout Analysis

Generally, a complete process of page information processing can be divided into three parts, which are layout analysis, layout understanding and layout reconstruction. Layout analysis refers to the process of detecting the type of each area after segmenting a document image into different parts. Therefore, layout analysis technology can be used to separate the picture, table and text in an image, and can be used to detect the region of interest (ROI) with a variety of detection algorithms.

Some progresses have been made on layout analysis technology in Chinese, English and other widely used languages. The research of layout analysis based on traditional optical character recognition (OCR) and deep learning are both in-depth. Compared with printed characters, the style of handwritings is more diversified, thus, layout analysis of handwritings is more complex. Therefore, information segmentation for handwritten documents is still one of the technical breakthroughs in layout analysis. In recent years, the layout analysis of handwritten script has got good results. For traditional methods, there are mainly three advanced research directions: layout analysis based on deformable model, based on clustering and based on structural prediction. Among them, one of the breakthroughs based on the deformable model is the horizontal set theory proposed by [1]. It solves the problem of line detection of curve text and line adhesion segmentation of adjacent text lines. Reference [2] proposed a method based on clustering, which used connected region analysis and minimum spanning tree to determine the text line cluster, and then obtained the text line. Reference [3] used conditional random domain (CRF) to classify connected components, then separating text and non-text regions. As for deep learning area, one of the most famous researches is that [4] put forward a new layout analysis method based on full convolution network (FCN) to study the medieval ancient prose script. After regional segmentation, line contour extraction and baseline detection, text line can be detected exactly. The model has great robustness on different experimental dataset and accuracy can reach more than 90% on average. However, the application of layout analysis technology is not limited to these widely used languages. With the deepening of layout analysis application, the types of languages that can be processed are also enriched. For example, [5] used traditional OCR algorithm to conduct in-depth research on Mongolian image layout analysis. Reference [6] proposed a method for layout analysis of Tibetan historical documents based on convolution noise reduction from encoder, effectively separating different layout elements of Tibetan historical documents. Reference [7] proposed a method for layout analysis of Manchu documents based on Mask R-CNN, which achieved a good detection and segmentation effect. Because the layout analysis technology gradually presents the characteristics of strong extensibility and processing objects in wide range, it is now more and more commonly used in the field of image word processing.

B. Multi-Style Image Synthesis

1) Techniques of Image Generation

GAN which has made great progress in recent years is one of image synthesis technology with great performance. Besides GANs, there are three common generative Models: flow-based Models, autoregressive Models and VAE. The following discussion compares GANs with the other three models from the perspectives of theory and performance.

Theoretically speaking, flow-based models, autoregressive Models and VAE are all based on explicit probability density. The flow model can apply the reversible transformation to the prior samples to calculate the accurate logarithmic likelihood. The autoregression model can decompose the observed distribution into conditional distribution. It processes one observed component at a time, and then calculates the accurate maximum likelihood estimation. Variational auto-coder allows learning probability models with implicit variables, and estimates approximate probability density by Bayesian inference. However, GANs use implicit coding and it can model directly on implicit probability density. Therefore, even if the probability density is not computable, GANs can still be applied. At the same time, the traditional probabilistic generation model generally requires Markov chain sampling and inference, while GANs avoid this process with extremely high computational complexity, thus GAN improves the generation efficiency. In this way, the actual application scenarios of GANs are more extensive.

Odena compared the synthesis performance of GANs and other two models in the "open questions" in [8], results are shown in Table I.

TABLE I
THE COMPARISON OF GANS, FLOW MODELS AND AUTOREGRESSIVE MODELS

	Parallel	Efficient	Reversible
GANs	Yes	Yes	No
Flow Models	Yes	No	Yes
Autoregressive Models	No	Yes	Yes

In terms of efficiency, the computational cost of GANs is higher than that of the flow model, but it has higher training efficiency. From the perspective of parallelism, although autoregressive model (such as Pixel-CNN) can produce clearer images, it still needs to model through the conditional distribution on pixels, so its evaluation speed is slow and its performance is limited in large-scale generation tasks [9]. In addition, from the perspective of the quality of generated images, compared with the VAE, GANs is prone to generate clearer images.

2) Mechanism of GANs

Goodfellow et al. [10] put forward the Generative Adversarial Network (GAN) and it has become one of the most promising methods in recent years in the field of unsupervised learning. The main idea of GANs is to conduct continuous confrontational learning between generator and discriminator, the generator tries to generate more realistic images in training, while the discriminator aims to distinguish the newly generated images from real images [11]. The two processes are carried out alternately, and should be continuously trained and updated until the two processes reach the Nash Equilibrium. Formula (1) can be used to describe the above process:

$$\min_G \max_D E_{x\sim P_{data}} \log[D(x)] + E_{x\sim P_z} \log[1-D(G(z))] \quad (1)$$

x represents the real sample; z represents the noise of the input generator; $E_{x\sim P_{data}(x)}$ represents the distribution of generated samples and formal samples; $E_{x\sim P_z(x)}$ represents noise distribution. In the process of network training, on the one hand, it let D tell the differences between the generated data and the real data ultimately to make $D(G(z))$ as close to 0 as possible. On the other hand, it trains G to generate more real data, so as to make $\log(1 - D(G(z)))$ as small as possible.

3) The Development of GANs

Since GAN was first proposed, it has developed rapidly, and the actual improvement of image synthesis model is almost too fast to keep up. New GANs models, such as [12] and [20], have been proposed in academia to improve the effect of image generation and improve the performance of GANs in terms of quality (such as authenticity, sharpness and size), stability and diversity.

The DCGAN is the first model that uses FCN for data generation since GAN was put forward. Compared with traditional GAN structure, DCGAN has made a breakthrough in performance and greatly improved the learning ability of the model. The DCGAN model is put forward by Radford et al. [12] is the initial exploration of combining GANs with convolutional neural network. It improved GAN model only from the angle of model structure while avoiding putting forward a feasible optimization scheme from the algebraic defects of GAN itself. Thus, the category of outputs is difficult to control and the limitation of the output image with poor quality still exists. In order to solve the problem that the output category is difficult to control, [13] proposed Conditional GAN (CGAN) based on the GAN. By encoding the label information as a vector and connecting it to the input terminal of generator and discriminator, it adjusts the weight of each node of the network and makes the optimization objective change. In order to train model in an unsupervised learning process, [14] proposed InfoGAN which further reduced the cost of manual annotation. The condition GAN makes it possible to control the output of GANs. In order to solve the problem of low quality of output images, [15] proposed a pyramid GAN model (LAPGAN) to improve the resolution of generated images. Based on this, [9] proposed a GANs model with higher performance, which can generate 1024px × 1024px high-resolution face pictures, promoting the capability of GANs to improve further.

Since GAN was put forward, its model has been changed one after another, and it is becoming one of the most powerful tools for synthesizing data. Driven by problems, the academic circles continuously put forward the GANs mechanism that trains more stable and generates images with higher quality.

III. NETWORK: LAYOUT ANALYSIS AND MULTI-STYLE TEXTURES FUSION

A. Network Framework

The overall framework flow of layout analysis and multi-style textures fusion of ancient books proposed in this paper is shown in Fig. 2. The overall framework of the network can be divided into two sub-processes, one is used to synthesize the target image, and the other is used to form the text image with neat layout. Among them, the original images of ancient books get a neat text layout after layout adjustment; the texture data set trains to synthesize texture image and its outputs are used to evaluate the synthesis effect of the model. Finally, the foreground texts and synthesized background are fused to get the results of digitalization.

B. Data Collection and Collation

The texture data of ancient books used in this study mainly come from three sources:

1. By modifying keywords, we crawled data in frequent times in order to get sufficient data, then we performed data cleaning. In addition, we selected appropriate images in texture data set ETH Synthesizability, DTD and KTH-TIPS;
2. We collected well-preserved background textures from Yi

ancient materials provided by Guizhou research institute of Yi studies, then selected the texture images with high resolution as our study material;

3. We made manual screenshots of ancient books' background among the ancient materials displayed in the national digital library.

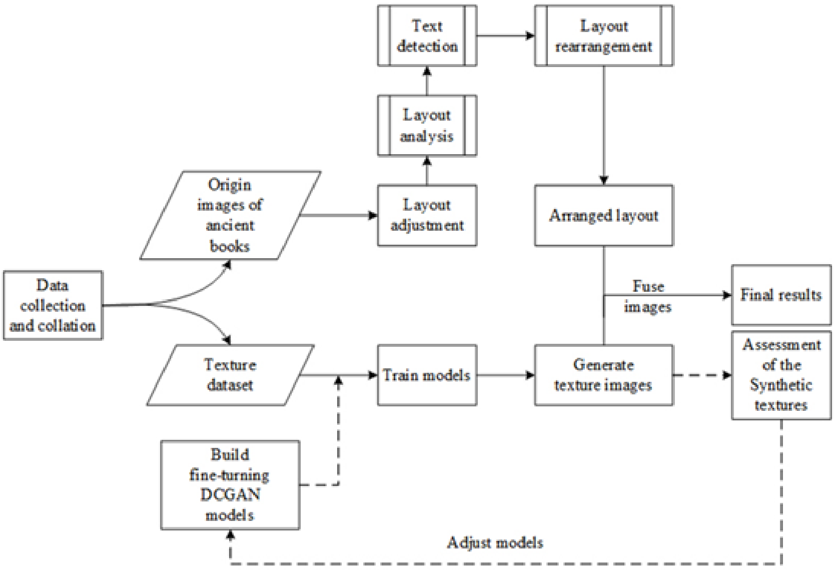
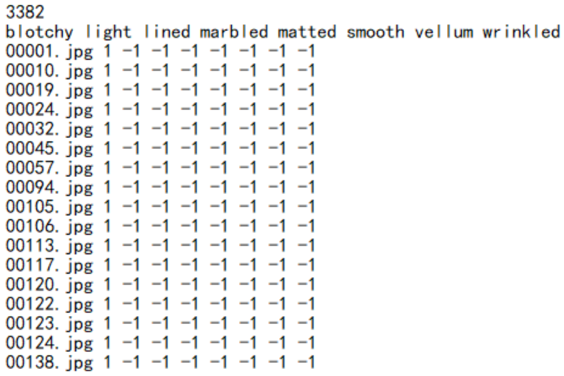


Fig. 2 Overall network framework

After data cleansing, we eventually got 3,382 data that can be used for training and testing, which can be divided into 8 texture types. Among them, 87.2% came from web crawling, 8.9% came from open digital library, and 3.9% came from Yi ancient books collected by our research team. Subsequently, the Dataset was renamed and normalized according to a uniform rule, and arranged in a same folder "Dataset". Finally, all the data in "Dataset" were labeled in the way of CelebA Dataset, as shown in Fig. 3.



```
3382
blotchy light lined marbled matted smooth vellum wrinkled
00001.jpg 1 -1 -1 -1 -1 -1 -1 -1
00010.jpg 1 -1 -1 -1 -1 -1 -1 -1
00019.jpg 1 -1 -1 -1 -1 -1 -1 -1
00024.jpg 1 -1 -1 -1 -1 -1 -1 -1
00032.jpg 1 -1 -1 -1 -1 -1 -1 -1
00045.jpg 1 -1 -1 -1 -1 -1 -1 -1
00057.jpg 1 -1 -1 -1 -1 -1 -1 -1
00094.jpg 1 -1 -1 -1 -1 -1 -1 -1
00105.jpg 1 -1 -1 -1 -1 -1 -1 -1
00106.jpg 1 -1 -1 -1 -1 -1 -1 -1
00113.jpg 1 -1 -1 -1 -1 -1 -1 -1
00117.jpg 1 -1 -1 -1 -1 -1 -1 -1
00120.jpg 1 -1 -1 -1 -1 -1 -1 -1
00122.jpg 1 -1 -1 -1 -1 -1 -1 -1
00123.jpg 1 -1 -1 -1 -1 -1 -1 -1
00124.jpg 1 -1 -1 -1 -1 -1 -1 -1
00138.jpg 1 -1 -1 -1 -1 -1 -1 -1
```

Fig. 3 Texture data labels of our study materials. We collated our textures of ancient materials. The labels of 8 textures are represented as blotchy, light, lined, marbled, matted, smooth, vellum, wrinkled

C. *Layout Analysis*

The layout analysis of texts includes two sub-steps: foreground text segmentation and layout rearrangement of the segmented single text.

1) Text Detection and Segmentation

As for foreground text segmentation, we adopted the top-down projection method for layout analysis. Before detection, we adopted different strategies for image preprocessing in different cases.

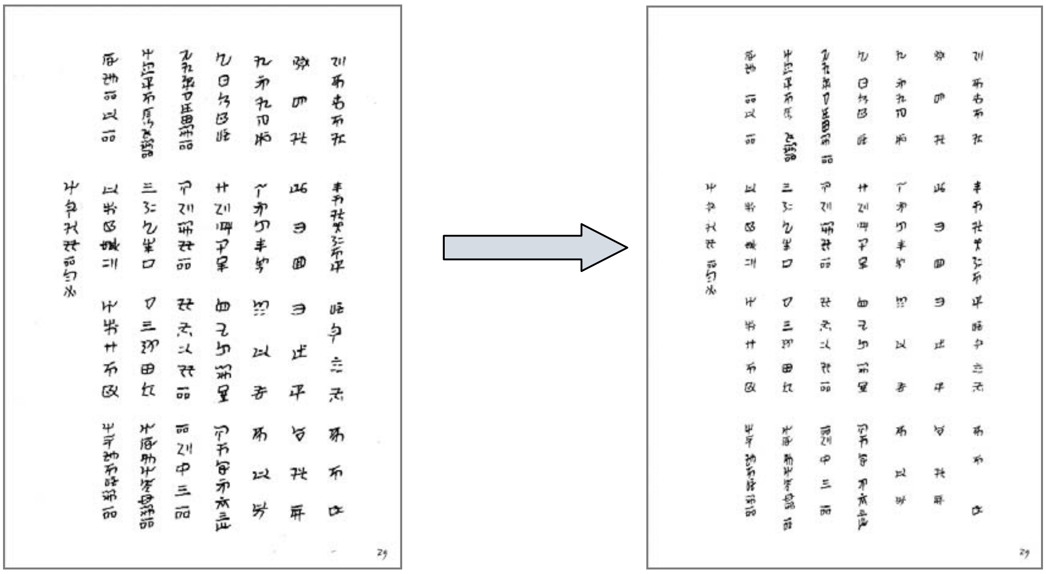
For the image with single background color (the scanned image has a single white color), the non-local average denoising method can be used to remove the noise, and then the binary image is obtained by corrosion expansion. For the script of ancient books with complex background, adaptive thresholding was first adopted according to the uneven distribution of image brightness, then median filter was used to remove salt and pepper noise in the background, and finally the binary image was obtained by corrosion expansion. After preprocessing, the image of ancient books with simple background is detected and segmented by the traditional method based on projection. For the image of ancient books with extremely complex background, we chose MSER detection algorithm combined with non-maximum Suppression (NMS) for text detection and segmentation. Finally, based on the result of detection and segmentation, the position information of a single text on the whole image of ancient books was obtained, and a list of text position information was obtained (list items like (left-top-x, left-top-y, width, height)) and it was named as position.

2) Text Rearrangement

To solve the problem of messy text alignment, this paper proposes an algorithm called PR.

In order to beautify the layout of ancient books and increase the readability of ancient books, the text alignment of horizontal and vertical columns is carried out without changing the original paragraph division of ancient books. Specific PR algorithm is shown in Algorithm I.

The display of a single page of an ancient book before and after rearrangement with PR algorithm is shown in Fig. 4.



(a) (b)

Fig. 4 The contrast effect of rearranging a single layout of in ancient Yi before (a) and after (b) applying PR algorithm: (a) is the original picture in ancient Yi before using PR and (b) is the effect picture after using PR algorithm

D. The DCGAN Model

1) Basic Structure of DCGAN

DCGAN is a classical unsupervised training generation model, which is easy to expand, and the output quality of training results is high. This model is combined with the powerful feature extraction capability of convolutional neural network (CNN) to improve the learning effect of GANs generation model. It solves the training problems caused by poor initialization through Batch Normalization, effectively improving the stability of training results. This is because Batch Normalization (BN) proves the fact that the initialization of a generating model is crucial as BN can avoid schema collapse which means all the generated samples converge to a point, in another word, the generated samples are the same [16].

Algorithm 1 Position Rearrangement (PR)

```
Input: list position[($x_i$,$y_i$,$w_i$,$h_i$)], $i \in [0, len(\text{position})]$
       col_start: the starting subscript of column
                  with the most words
       col_end: the end subscript of column
                  with the most words;
       row_dis: maximum space between different columns
       col_dis: minimum space between different columns.
Main Iteration: Iterate through all the elements in position
$i \in [0, len(\text{position})]$
$l \Leftarrow len(\text{position})$;
  Δcalculate element number of the "position"
while $i \neq l-1$ do
    $i \Leftarrow 0$;
    while $j \neq i$ do
        $j \Leftarrow 0$;
        if abs(position[i][0]-position[j][0])<row_dis then
            position[j][0] ⇐ position[i][0];
  Δrearrange the lengthways element in "position"
        end if
        $j \Leftarrow j+1$
    end while
    $i \Leftarrow i+1$
end while
while $i \neq l-1$ do
    $i \Leftarrow 0$;
    while $j \neq$ col_end do
        $j \Leftarrow$ col_start;
        if abs(position[i][1]-position[j][1])<col_dis then
            position[i][1] ⇐ position[j][1];
  Δrearrange the transverse element of "position"
        end if
        $j \Leftarrow j+1$;
    end while
    $i \Leftarrow i+1$;
end while
Output: The rearranged list named reposition after adjusting input elements
        coordinates.
```

Compared with GAN models in [8]-[11], DCGAN always has high degree of freedom while training and it does not have a strict control over the generated data mode. However, the following characteristics enable DCGAN model to be capable of generating great performance in this work: (1) In the feature extraction layer of generator and discriminator, CNN is adopted to replace the multi-layer perceptron in GANs. Thus, the network structure has great flexibility; (2) The hyper parameter setting in the model is highly flexible, therefore, it has diversified input and output modes with good expansibility; (3) The training is not specific to a specific data set, the model structure is clear and simple; (4) Compared with other unsupervised methods, the image features extracted by DCGAN discriminator are more effective and suitable for image classification tasks.

In order to develop the capability of DCGAN model that is put forward by Radford et al. [12], we conducted multi-mode training by adjusting the type and dimension of the prior noise. Meanwhile, we control the texture style of the output image by adding Softmax classifier to the model output layer. By comparing the FID score, we got the best performed fine-tuning model M8 with the minimal training loss, and the structure of model M8 was shown in Fig. 5.

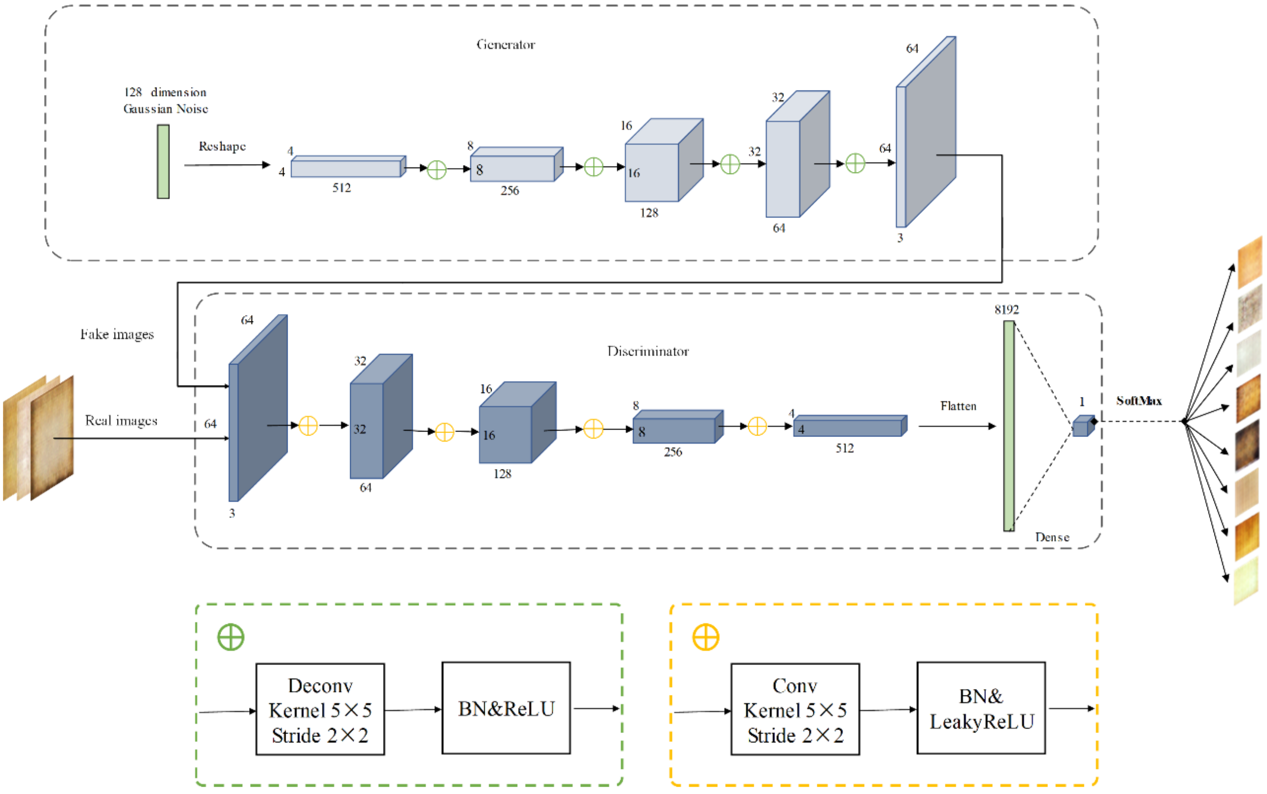


Fig. 5 Structure of Fine-Turning Model M8

TABLE II
FINE-TUNING TRAINING MODEL

Noise type	Models	N-dimension	Kernel	Type-Labels
Random	DCGAN	100	5×5	No
Random	M0	100	3×3	No
	M1	100	7×7	No
	M3	128	3×3	No
	M4	128	7×7	No
Random	M6	100	5×5	Yes
	M2	128	5×5	Yes
	M5	256	5×5	Yes
Gaussian	M7	128	3×3	Yes
	M8	128	5×5	Yes
	M9	256	5×5	Yes

2) Constructing Fine-Tuning Models

In order to further improve the quality of the model output and enhance the generalization ability of the model, we made an in-depth exploration on the basis of DCGAN model and constructed the fine-tuning model M0-M9. The model parameters are shown in Table II. The training output of each fine-turning model is obtained by controlling the training parameters of the model. After training and assessing, the synthesis performance of each model can be evaluated.

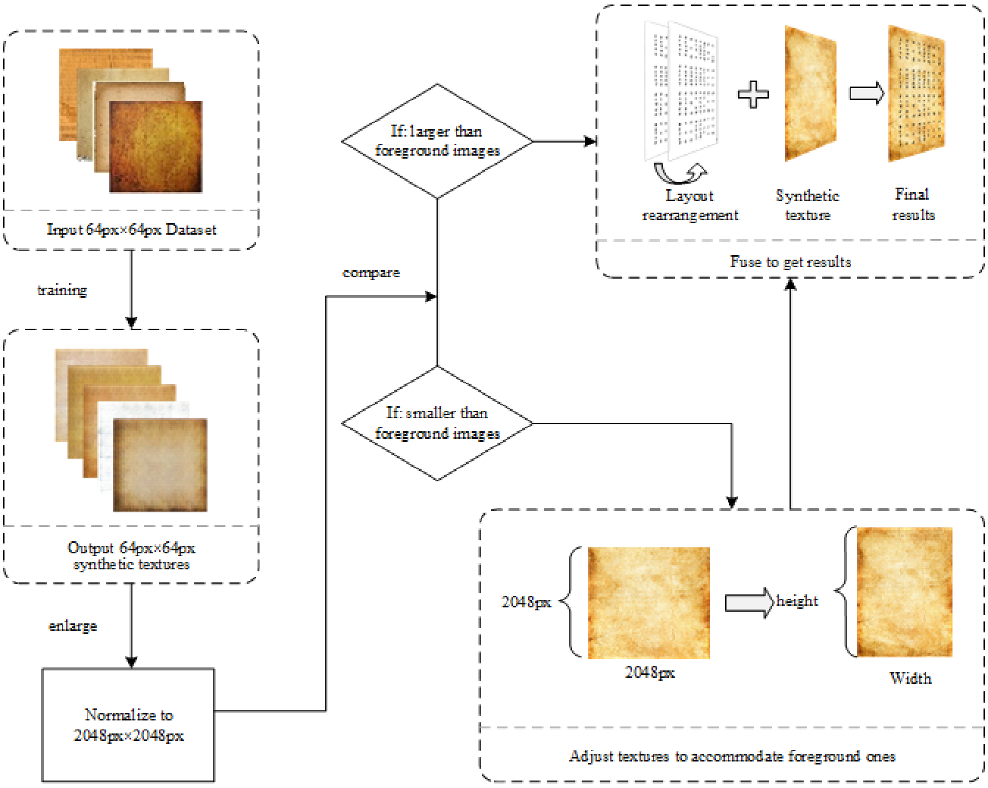

Input 64px×64px Dataset
training
Output 64px×64px synthetic textures
enlarge
Normalize to 2048px×2048px
compare
If: larger than foreground images
If: smaller than foreground images
Layout rearrangement
Synthetic texture
Final results
Fuse to get results
2048px
2048px
height
Width
Adjust textures to accommodate foreground ones

Fig. 6 Comprehensive image fusion strategy

E. *Image Fusion*

In the data processing stage, the texture images of ancient books of various styles are normalized to the size of $64\text{px} \times 64\text{px}$ as the network inputs. In order to improve the training efficiency, the size of images that directly output by GANs should not be too large. Therefore, the image size synthesized by this experiment is $64\text{px} \times 64\text{px}$ .By adopting interpolation method, we enlarged the images into the size of $2048\text{px} \times 2048\text{px}$ suitable for most pages so as to facilitate subsequent batch processing. Meanwhile, because the size of $2048\text{px} \times 2048\text{px}$ cannot necessarily accommodate all information in each page, we need to compare the size of the ancient books and the synthetic textures in special cases. The comprehensive image fusion strategy process is shown in Fig. 6.

F. *Evaluation of Background Synthesis Results*

1) Loss Function

Taking advantages of cross entropy as the loss function can effectively accelerate the convergence rate of the network [17]; meanwhile, it can effectively evaluate the fitting degree of the network and prevent the model from over-fitting. In this paper, binary cross entropy is adopted as the loss function of the training model.

We selected sigmoid as the activation function of the last layer in the model. The definition of sigmoid activation function is shown in (2) [18], where x is the output of the convolutional network of the previous layer.

$$\sigma(x) = \frac{1}{1+e^{-x}} \qquad (2)$$

The mathematical expression of cross entropy is shown in (3), where y is the label of the training sample and a is the actual output of the network.

$$C = -\frac{1}{n}\sum_{x}[y\ln a + (1-y)\ln(1-a)] \qquad (3)$$

Obviously, $a = \sigma(x)$ is workable for (2) and (3). In GANs, the discriminator determines that the label from the real data sample is 1, while the label from the generated sample is 0, which is a dichotomous problem, that is, $n = 2$ is a condition of in a dichotomous cross entropy. Therefore, the optimization process is similar to the dichotomy of sigmoid, that is, the cross entropy of sigmoid. Therefore, the loss function is shown in (4):

$$Loss(x,y) = -\frac{1}{2}[y\ln(\sigma(x)) + (1-y)\ln(1-\sigma(x))] \qquad (4)$$

It can be observed that the closer the predicted output $\sigma(x)$ is to 1, the smaller the value of loss function will be; the closer the predicted output $\sigma(x)$ is to 0, the larger the value of loss function will be. At the same time, the larger the difference between the predicted output $\sigma(x)$ and the sample label y, the larger the loss function, that is, the larger the "penalty" for the current model, which is determined by the characteristics of the logarithmic function itself. Therefore, the model will tend to make the predicted output closer to the real sample label.

2) FID Assessment

In order to evaluate the composite results of the GANs model, large number of images need to be compared, so some automated methods can provide us much benefit to calculate indicative measures in large image sets. In general, there are two commonly used methods for image evaluation. One is the structural similarity index (SSIM). For instance, [19] adopted SSIM as their evaluation standard. The other is image diversity assessment indicator Inception Score, which is to apply a pre-trained neural network to the generated image and calculate its output or statistics for a specific hidden layer.

Existing methods used to evaluate the image similarity, such as MS-SSIM that put forward by [20], can reliably found large-scale collapse mode. However, Karras et al. [9] pointed out that existing methods to evaluate image quality, such as MS-SSIM, could not respond to minor influences, such as color or texture changes, and they were not used to directly evaluate the similarity between image quality and training set. In addition, Inception Score also has limitations. When assessed with Inception Score, its training set can only be ImageNet, which determines that the generation model should also be trained on ImageNet.

In view of the limitations of existing assessment methods, [21] proposed a new assessment method FID in 2018, which used multivariate Gaussian statistics (such as mean and covariance) to measure the Fréchet Distance between the generated samples and the real samples. It solved the problem of single training set and made it possible for the synthetic sample data to compare with the realm sample data. Therefore, this paper adopts a more appropriate similarity assessment strategy FID to evaluate the results of background synthesis. The formula is as follows:

$$d^2((m,C),(m_w,C_w)) = \|m-m_w\|_2^2 + \mathrm{Tr}(C+C_w-2(CC_w)^{1/2}) \quad (5)$$

m represents mean and C represents covariance. (m,C) represents the Gaussian statistics of a randomly generated image, and (m_w,C_w) represents the Gaussian statistics of a randomly generated real image. The smaller the FID value, the better the performance of the generated model and the more realistic the synthesized image.

IV. Experiment and Analysis

A. Experimental Environment and Tools

Under the condition of GPU 1060 and 8GB memory hardware, our work was implemented through the PyCharm programming environment and TensorFlow framework, and the training output and fusion results were visualized with the help of the visualization tools such as Tensorboard and matplotlib.

B. Determining Training Parameters

The DCGAN model adopts Stochastic Gradient Descent (SGD) for training. So, when training the neural network, it is necessary to set parameters to control the updating speed of learning rate. If the learning rate is too small, the convergence speed will be reduced greatly and the training time will be increased; if the learning rate is too high, the parameters may oscillate back and forth on both sides of the optimal solution. It is obvious that if the learning rate is too high or too low, the training instability will be profoundly increased. Therefore, the learning rate corresponding to the curve with stable fluctuation of the loss function in the training process can be regarded as the best learning rate of the current model. When the training step is 2560, we measured the change of loss function and set learning rate as 0.0001, 0.0002 and 0.0003 respectively. Results are shown in Fig. 7.

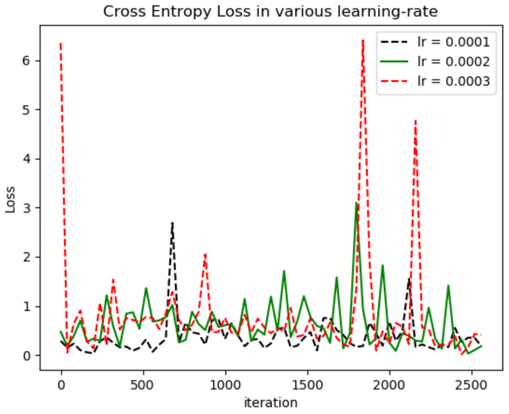


Fig. 7 Curve of loss function under different learning rates

As shown in Fig. 7, when the learning rate is set at 0.0002 and 0.0001, the training is relatively stable, while when the learning rate is set at 0.0003, the loss function fluctuates greatly, which is extremely unstable and the loss function is large, so 0.0003 is not a good option. When the learning rate is 0.0002, the value of the loss function fluctuates around 1, while when the learning rate is 0.0001, the value of the loss function fluctuates around 0.5 and its predicted output is closer to 0 compared with 0.0002 case. Thus, the corresponding loss function is larger when the learning rate is 0.0001. Therefore, it is appropriate to choose 0.0002 as the learning rate through experimental verification.

In the training process of GANs model, we adopted momentum algorithm to accelerate the training, then, we used Adam optimizer and momentum optimizer to optimize the model. The value of momentum optimizer should not be too high, for which will lead to training oscillation and instability. Through experimental verification, 0.5 is appropriate. In the experiment, the training step was set as 28000, 5200 and 2560 respectively, and finally the training result of 2560 steps was adopted as the final synthesis result. Relevant parameters of training are determined through experiments, and the settings of the above parameters are the same as the learning rate, which will not be repeated.

C. *Training the Fine-Turning DCGAN Models*

1) Training 28000 Steps

When the number of training is set to 28000, the model would output an image of (8×64px)×(8×64px) in each training interval. The synthetic textures generated by partial trainings are shown in Fig. 8.

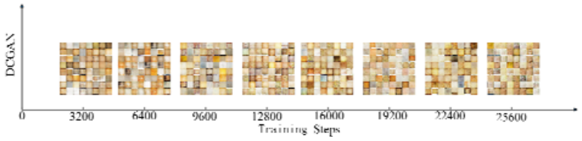
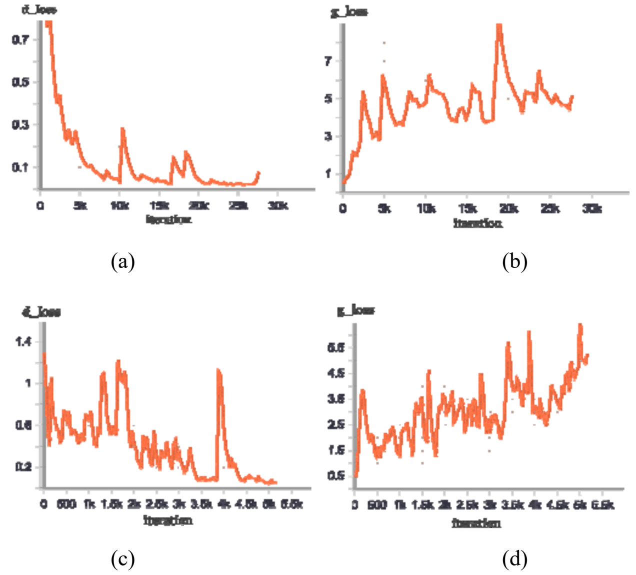


Fig. 8 Training generation diagram without Label-Type



(a)

(b)

(c)

(d)

Fig. 9 The loss functions curve of discriminator and generator for 28000 and 5200 steps. Among these four images, the curve of loss function with 28000 steps are (a), (b); the curve of loss function with 5200 steps are (c), (d)

During this training, epoch was set to be 2000 and we outputted a composite image in each training interval of 400. As shown in Fig. 8, in the first several thousand steps, the pixel filling degree of the composite image increased, and the authenticity of synthetic images rose when the training steps increasing. However, when the training step reached 10000, the sharpness of the image was profoundly enhanced, thus, textures gradually seemed unreal. This phenomenon resulted in model overfitting. When the step of training process was too big, the model would be unstable. The visualized result of loss function curve of the generator and discriminator is shown in Fig. 9, d represents discriminator (same below), g represents generator (same below), the value of d_loss_28000 curve represents the predicted output of discriminator when training 28000 steps and the value of g_loss_28000 curve represents the predicted output of generator in the same process.

It can be observed from Fig. 9 (a) that the predicted output of discriminator infinitely approached 0 with the training steps increasing. Meanwhile, the predicted output of generator was always high and far greater than 1. In another words, the discriminator judged that the synthesized image was fake when training steps reached 28000.

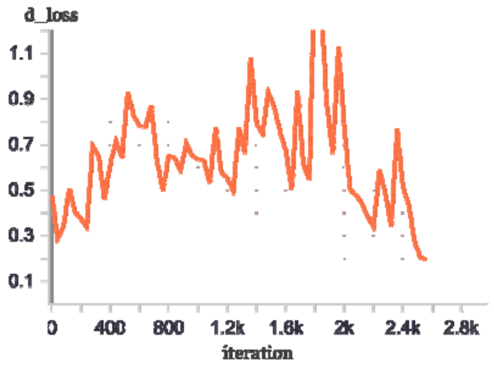
2) Training 5200 Steps

It was reasonable to consider that the interval of the iterations should be reduced by observing Figs. 9 (a), (b). Thus, we set the training steps as 5200 for further observation. At this time, epoch was set to 200. Since the training steps were greatly reduced, the output interval should be correspondingly narrowed down so it was set as 40 this time. The predicted output of generator and discriminator is shown in Figs. 9 (c), (d). By observing Fig. 9 (c), it was clear that when the training steps reached 2000 or above, the d_loss dropped sharply, and the value of the curve dropped from the original state of fluctuation around 1 to 0. Furthermore, when the training step was above 5000, the model was still over-fitting, so the interval needed to be further narrowed down.

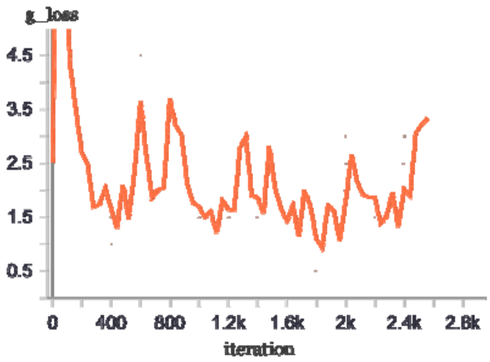
3) Training 2560 Steps

The performances of generator and discriminator were shown in Fig. 10; we could draw a conclusion that the value at the junction of curve changes was about 2600. Thus, we set the training steps as 2560 for further observation. At this time, epoch was set 100 and the training interval was 40. The curve of generator's and discriminator's loss is shown in Fig. 10.

When training steps was 2560, the predicted output of discriminator and generator became steady, fluctuating around 1. As the training steps increased within the scope, the predicted result was close to the sample label 1, which indicated that the loss function was small enough. At this time, synthetic images were closer to real textures and model performed best.



(a)



(b)

Fig. 10 The loss functions curve of discriminator and generator for model training 2560 steps y

D. Evaluation of Background Synthesis Results

In this paper, FID has been used to evaluate the generation model, which is described in detail above. We measured FID between the images which were synthesized in different training models and the real textures in the folder Dataset. When training step was 2560, the d_loss, g_loss and FID results corresponding to different models were shown in Table III (results reserve two decimal fractions).

TABLE III
CRUCIAL PARAMETERS AND FID RESULTS OF TRAINING MODELS

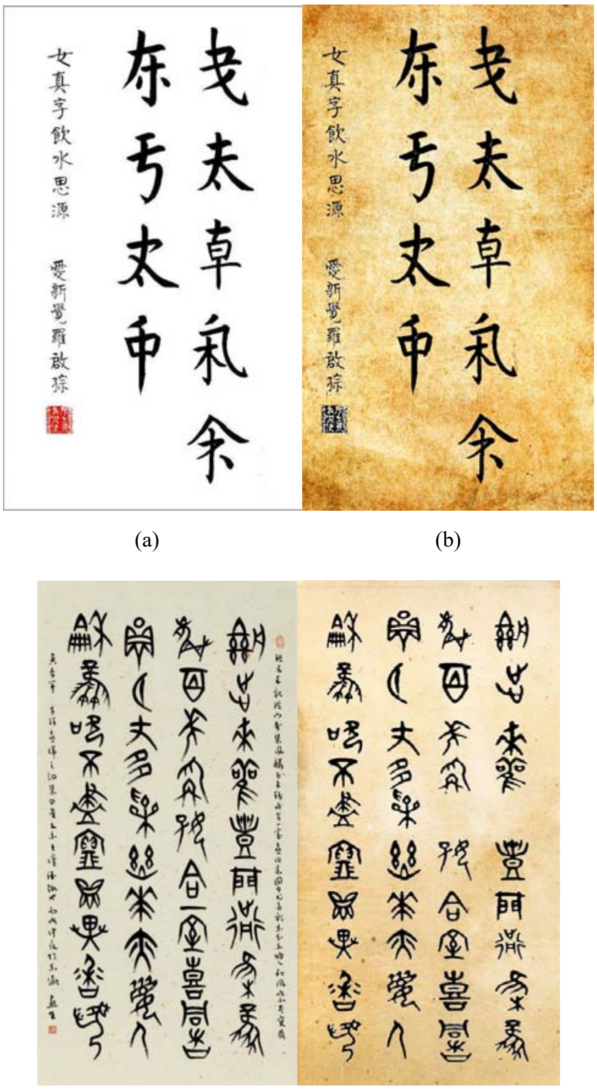
Models	d_loss (2560 steps)	g_loss (2560 steps)	FID Scores
DCGAN [12]	0.1210	5.4141	165.33
M0	0.2100	2.2163	169.74
M1	0.1456	2.7249	207.15
M2	0.5473	1.2655	155.63
M3	0.3809	1.6016	196.61
M4	1.3235	0.4690	165.33
M5	0.5532	1.1082	140.23
M6	0.4859	1.2244	148.87
M7	0.2425	2.8469	157.41
M8	0.4345	1.2367	138.63
M9	0.1969	2.3818	160.79

According to Tables II and III, when the prior noise dimension was same, compared with fine-tuning models with kernel of 3×3 and 7×7, the model with 5×5 kernel had a smaller FID score with better performance; when kernel of the model is 5×5 , the performance of model like M5 was greater than M4 and slightly better than M6. Therefore, we could come about a conclusion that when the noise type was random, M2, M5 and M6 performed better among them all. Subsequently, we further studied these three models by changing the noise type into Gaussian, and found that the M8 had the lowest FID score, thus its model performance was the best and its authenticity of the synthetic textures was improved by 19.26%, when compared with DCGAN.

E. Image Fusion and Visualization

Some visualization outputs are shown in Fig. 11. Our network can synthesize a variety of textures of ancient books so as to recover different kinds of ancient materials. What's more, the proposed PR algorithm can effectively beautify images and has better robustness for layouts with different features.

In order to verify the universality of the network in this paper, we randomly selected test samples from ancient Yi language, Jurchen ancient books, seal of Qin Dynasty ancient paintings and other ancient books with different layout structure according to 18.50%, and obtained relatively ideal results. The average accuracy of text detection based on layouts with multiple ingredients and layouts with single ingredient are shown in Table IV.



(a)

(b)

(c)

(d)

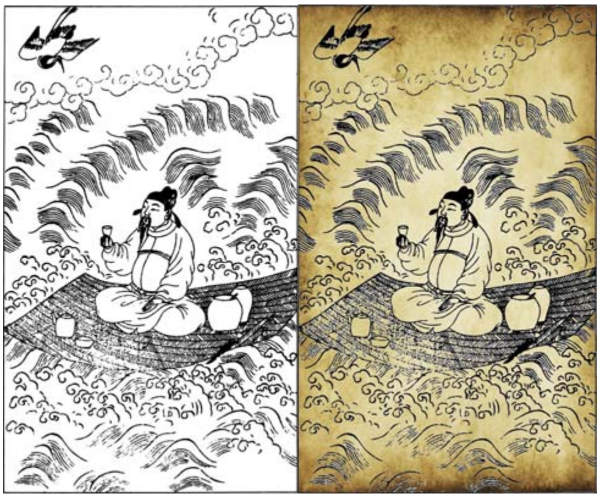


Fig. 11 Some of our visualization results in seal (one of the ancient Chinese), Jurchen scripts and ancient drawings. The original images show in (a), (c), (e) and the results respectively show in (b), (d), (f). Compared with original images without using our network, the (b) is more well-organized after being rearranged, meanwhile, (e) and (d) become more authentic with generated texture. (a) Jurchen Script (raw); (b) Jurchen Script; (c) Seal Script (raw); (d) Seal Script; (e) ancient drawing (raw); (f) ancient drawing

TABLE IV
THE TEXT DETECTION ACCURACY OF DIFFERENT TEST SAMPLES

Datasets	Number of images	Number of test samples	Average-accuracy of samples with multiple ingredient	Average-accuracy of samples with single ingredient
Ancient scripts in Yi	772	142	0.6379	0.9687
Ancient scripts in Jurchen	152	28	0.7916	0.9737
Ancient scripts in seal	579	107	0.8095	0.9268
Ancient drawings	216	40	0.7727	0.9102

It could be inferred from Table IV that the average detection accuracy of layouts containing just a single text element is higher than that of layouts with multiple ingredients, and this difference can greatly be found in the detection of ancient Yi. For different test samples, the average detection accuracy of layout containing a single element could reach at least 91.02%, among which the Jurchen ancient books got the highest score of 97.37%. However, due to the limitations of the traditional detection algorithm, there still existed problems like stroke loss, as shown in Fig. 11 (d). In the future, we plan to expand the data set and study how to further improve the performance of model network by combining with deep learning detection algorithm.

V. CONCLUSION

This paper takes advantages of proposed PR algorithm based on layout analysis and GANs technique to realize multi-style texture recovery so as to beautify a variety of ancient materials. The deep learning technology has been applied to the synthesis and fusion of ancient books' background. What's more, the fine-tuning DCGAN model M8 has been adopted to synthesize the texture of ancient books, which makes contributions to the digitalization process of ancient books. In addition, by using dichotomy cross entropy loss function and FID to evaluate the synthesis results, the convergence efficiency of our model has been greatly improved, and problems such as overfitting and mode-collapse were effectively avoided.

Due to the data of ancient books' texture are still limited, there still exists a big challenge for further research. Since the synthesis performance of our model can still be enhanced on the basis of enlarged data, we will adopt the comprehensive method of VAE and GANs model for further exploration in the future.

REFERENCES

[1] Li Y, Zheng Y F, David D., et al. “Script-independent text line segmentation in freestyle handwritten documents.” *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol 30, no.8, pp. 1313-1329. Aug. 2008.

[2] Yin F, Liu C L. “Handwritten Chinese text line segmentation by clustering with distance metric learning.” *Pattern Recognition,* vol 42, no. 12, pp. 3146-3157. Dec. 2009.

[3] Li X H, Yin F, Liu C L., “Printed/Handwritten Texts and Graphics Separation in Complex Documents Using Conditional Random Fields,” in Proceedings of the IAPR International Workshop on Document Analysis Systems (DAS), Vienna, Austria, 2018, pp. 145-150.

[4] Simistira F., Bouillon M, Seuret M, et al. “ICDAR2017 Competition on Layout Analysis for Challenging Medieval Manuscripts,” in Proceedings of the IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 2017, pp. 1361-1370.

[5] Wang Y W. “Research and implementation of layout analysis and post-processing for Mongolian document images,” Ph.D. dissertation, Dept. Computer Sci.., Inner Mongolia Univ., Inner Mongolia, China, 2017.

[6] Zhang X Q, Ma L L, Duan L J, Liu Y Z, Wu J. “Layout Analysis for Historical Tibetan Documents Based on Convolutional Denoising Autoencoder.” *Journal of Chinese Information Processing,* vol 32, no. 07, pp. 67-73. July. 2018.

[7] Chen X, He J J, Li H J, Wu L X. “Manchu Document Layout Analysis Based on Mask R-CNN.” *Journal of Dalian Minzu university*, vol 21, no. 3, pp. 240-245. Mar. 2019. DOI: 10.13744/j.cnki.cn21-1431/g4.2019.03.010.

[8] Augustus Odena. “Open Questions about Generative Adversarial Networks,” presented at the *Distill,* Apr 9, 2019. DOI: 10.23915/distill.00018

[9] Karras T, Aila T, Laine S, Lehtinen J. (2018) Progressive growing of GANs for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196 (2018).

[10] Goodfellow I, Pouget-Abadie J, Mirza M, et al., “Generative adversarial nets” in Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, USA, 2014, pp. 2672-2680.

[11] Zhan Fang-Neng, Zhu Hong-Yuan, “Spatial Fusion GAN for Image Synthesis,” in Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, USA, 2019, pp. 3648-3657.

[12] Radford A, Metz L, Chintala S, “Unsupervised representation learning with deep convolutional generative adversarial networks,” in Proceedings of the International Conference of Learning Representation (ICLR), San Juan, Puerto Rico, 2016, pp. 2234-2242.

[13] Mirza M, Osindero S. (2014). Conditional Generative Adversarial Nets. arXiv preprint, arXiv:1411.1784(2014).

[14] Chen X, Duan Y, Houthooft R, et al, “InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets,” in Proceedings of the International Conference on Neural Information Processing Systems (NIPS), Barcelona, Spain, 2016, pp. 2172-2180.

[15] Denton E L, Chintala S, Fergus R, "Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks" in Proceedings of the International Conference on Neural Information Processing Systems (NIPS), Montréal, Canada, 2015, pp. 1486-1494.

[16] Chen S X, Wang X L, Han X, Liu Y, Wang M G. "A recognition method of Ancient Yi character based on deep learning." *Journal of Zhejiang University (science edition),* vol 46, no. 3, pp. 261-269. May. 2019. DOI: 10.3785/j.issn.1008-9497.2019.03.001.

[17] Ren J J, Wang N. "Research on Cost Function in Artificial Neural Network." *Journal of Gansu Normal Colleges*, vol 23, no. 2, pp. 61-63. Feb. 2018. DOI: 008-9020(2018)02-061-03.

[18] Zhou F, Li Y, Fan X Y. "Improved Loss Calculation Algorithm for Convolutional Neural Networks in Image Classification Application." *Journal of Chinese Computer System*, vol 40, no. 7, pp. 1532-1537. July. 2019.DOI: 1000-1220(2019)07-1532-06.

[19] Azadi S, Fisher M, Kim V, et al., "Multi-Content GAN for Few-Shot Font Style Transfer," in Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, USA, 2018, pp. 7564-7573.

[20] Wang Z, Simoncelli E. P., Bovik A.C., "Multi-scale Structural Similarity for Image Quality Assessment," in Proceedings of the Asilomar Conference on Signals, Systems and Computers, Pacific Grove, USA, 2003, pp. 9-12.

[21] Heusel M, Ramsauer H, Unterthiner T, et al., "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium," in Proceedings of the International Conference on Neural Information Processing Systems (NIPS), Long Beach, USA, 2017, pp. 6626-6637.

[22] Li Y, Zheng Y F, David D., et al. "Script-independent text line segmentation in freestyle handwritten documents." *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol 30, no.8, pp. 1313-1329. Aug. 2008.