Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 9922017, 13 pages
https://doi.org/10.1155/2021/9922017

Research Article

Separating Chinese Character from Noisy Background Using GAN

Bin Huang[1], **Jiaqi Lin**[1], **Jinming Liu**[1], **Jie Chen**[1], **Jiemin Zhang**[1], **Yendo Hu**[1], **Erkang Chen**[1] and **Jingwen Yan**[2]

[1]*Computing Engineering College, Jimei University, Xiamen 361021, China*
[2]*College of Engineering, Shantou University, Shantou 515063, China*

Correspondence should be addressed to Erkang Chen; ekchen@jmu.edu.cn

Received 17 March 2021; Revised 7 April 2021; Accepted 20 April 2021; Published 3 May 2021

Academic Editor: Philippe Fournier-Viger

Separating printed or handwritten characters from a noisy background is valuable for many applications including test paper autoscoring. The complex structure of Chinese characters makes it difficult to obtain the goal because of easy loss of fine details and overall structure in reconstructed characters. This paper proposes a method for separating Chinese characters based on generative adversarial network (GAN). We used ESRGAN as the basic network structure and applied dilated convolution and a novel loss function that improve the quality of reconstructed characters. Four popular Chinese fonts (Hei, Song, Kai, and Imitation Song) on real data collection were tested, and the proposed design was compared with other semantic segmentation approaches. The experimental results showed that the proposed method effectively separates Chinese characters from noisy background. In particular, our methods achieve better results in terms of Intersection over Union (IoU) and optical character recognition (OCR) accuracy.

1. Introduction

Converting paper documents into electronic documents and then recognizing them using optical character recognition (OCR) technology have been widely used in daily life. In recent years, with the development of machine learning technology, the recognition accuracy of OCR has been greatly improved [1–3]. We can now process a document with both machine-printed text and handwritten text and then recognize them separately [4, 5]. Similar applications can be found in the archiving and processing of historical documents [6, 7]. In the field of education, related technologies for examination paper autoscoring have emerged, which greatly reduce burden for teachers and students. Taking Figure 1 as an example, an examination paper with students' answers can first be processed by OCR, and then the recognized answers can be evaluated and scored automatically by the machine. Under certain circumstance, since the test paper template cannot be easily obtained, it is also necessary to directly identify the printed test paper template.

In order to achieve examination paper autoscoring, one of the technical challenges to be solved is handling overlapping characters. This may happen when an elementary school student did not master writing well or put annotation on the test paper. The current OCR technology cannot handle the mixed situation of printed text and handwritten text in the same image. Generally, only a single type of text can be recognized by OCR technology [8]. Our early experiments showed that when recognizing printed text, the OCR accuracy was greatly reduced if there were handwritten strokes or handwritten characters around the printed text. Even worse was that the machine was not able to find the text area needed to be recognized. Therefore, it is desirable to separate the handwritten characters from the printed characters on the examination paper and then process different text types accordingly. Furthermore, for Chinese characters, the separation of handwriting and printing becomes more difficult because the font structure is far more complicated than Western fonts [9, 10]. A slight loss or increase of strokes may change the meaning of the characters completely, which

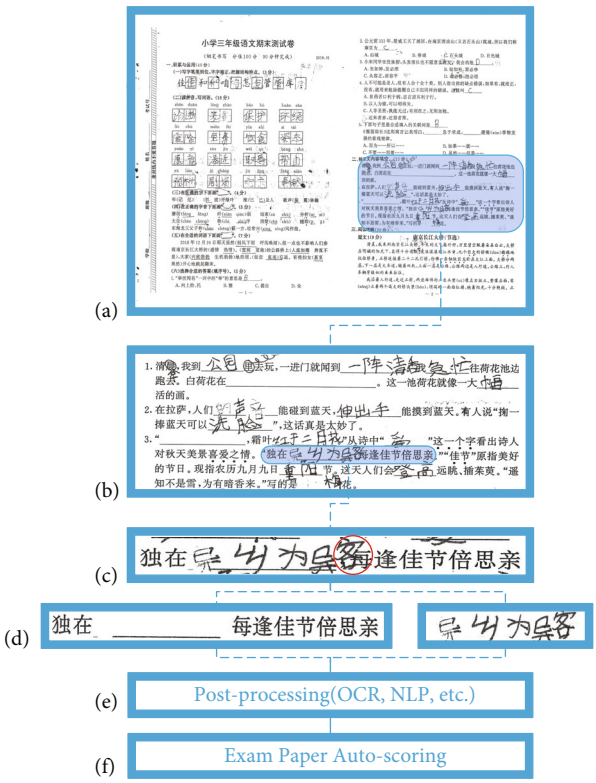

Figure 1: Basic process of examination paper autoscoring. (a) A sample examination paper which consists of both handwriting and printed text. (b) A subquestion with answers to be scored. (c) Handwriting touches or even overlaps with printed text. The red circle shows an example of overlapping characters. (d) The proposed method targets separation of overlapping Chinese characters into printed text (left rectangle) and handwriting (right rectangle). (e, f) After successful separation is made, postprocessing and autoscoring become feasible.

makes it difficult to separate effectively when handwriting fonts and printed fonts are highly overlapped.

Separating Chinese characters from noisy background (particularly with overlappings) can be considered an image semantic segmentation problem. Previous deep learning methods [11–13] have shown success in other applications. However, these methods have poor performance due to the complex structure of Chinese characters. To distinguish Chinese characters from similar fonts, we adopted a GAN-based approach [14–19]. A network, called DESRGAN, was developed to denoise background and reconstruct both the stroke structure and fine details of targeted Chinese characters. Our method used ESRGAN [19] as the basic network structure and applied dilated convolution to residual-in-residual dense blocks. A new loss function that can measure the integrity of the font skeleton was imposed. Then, the generator of the trained GAN model was used to separate targeted characters.

Our main contributions include the following: (a) we proposed a new network structure and a loss function that achieves the goal of Chinese character separation from noisy background, especially when characters are highly overlapped; (b) the proposed method achieved the best results in both IoU and OCR accuracy; and (c) our dataset (upon request) for further research is provided.

2. Related Work

Many applications in document processing need to solve the problem of separation of handwriting and printed. The Maurdor project created a realistic corpus of annotated documents in French, English, and Arabic to support the efficient development and evaluation of extraction method [20]. DeepErase [21] uses neural networks to erase ink artifacts on scanned documents and only extract text written by a user. The ink artifacts that DeepErase targets mainly include a tabular structure, fill-in-the-blank boxes, and underlines. Guo and Ma [22] used a machine-printed and handwritten annotation discrimination algorithm based on the Hidden Markov Model. Solely focusing on English and other Latin languages, their algorithm can locate the position of the handwritten part in the document in the form of a bounding box. Zagoris et al. [23] proposed a method of recognizing and separating handwritten content from document images mixed with handwritten and printed characters through the bag of visual word model. Their method first computes a descriptor for each block of interest and then classifies the descriptor into handwritten text, machine printed text or noise. However, few research has been focusing on highly overlapped texts, especially Chinese characters that are structurally more complex than English or other Latin languages.

Recent deep learning methods provide new ways for solving the separation of handwriting and printed. Li et al. [5] handles printed/handwritten text separation within a single framework by using conditional random fields. Their algorithm only performs extraction at connected component (CC) level. Each CC is classified into printed and handwritten no matter it is overlapping or not. U-Net [11], which performs well in many segmentation tasks, builds upon only convolution layers and the idea of propagating context information to higher resolution layers during upsampling. Pix2Pix [17] translates an input image into a corresponding output image. With a paired training dataset, it can output sharp and realistic images. Such features make it attractive for solving our character segregation problem. However, a paired training dataset may not be easy to find in real-world applications. CycleGAN [16] is an approach for learning to translate an image from a source domain to a target domain without paired examples. CycleGAN's coding scheme is to hide part of the information about the input image in low-amplitude, high-frequency signal added to the output image [14]. Another way to solve the separation of printed is to treat the image overlapped by handwriting and printed as a low-resolution picture, and the neural network determines which part needs to be enhanced in the process of single-image super-resolution. SRGAN [18] takes advantage of a perceptual loss function which consists of an adversarial loss and a content loss. Based on SRGAN, ESRGAN [19] improves the network structure by introducing the residual-in-residual dense block and computes perceptual

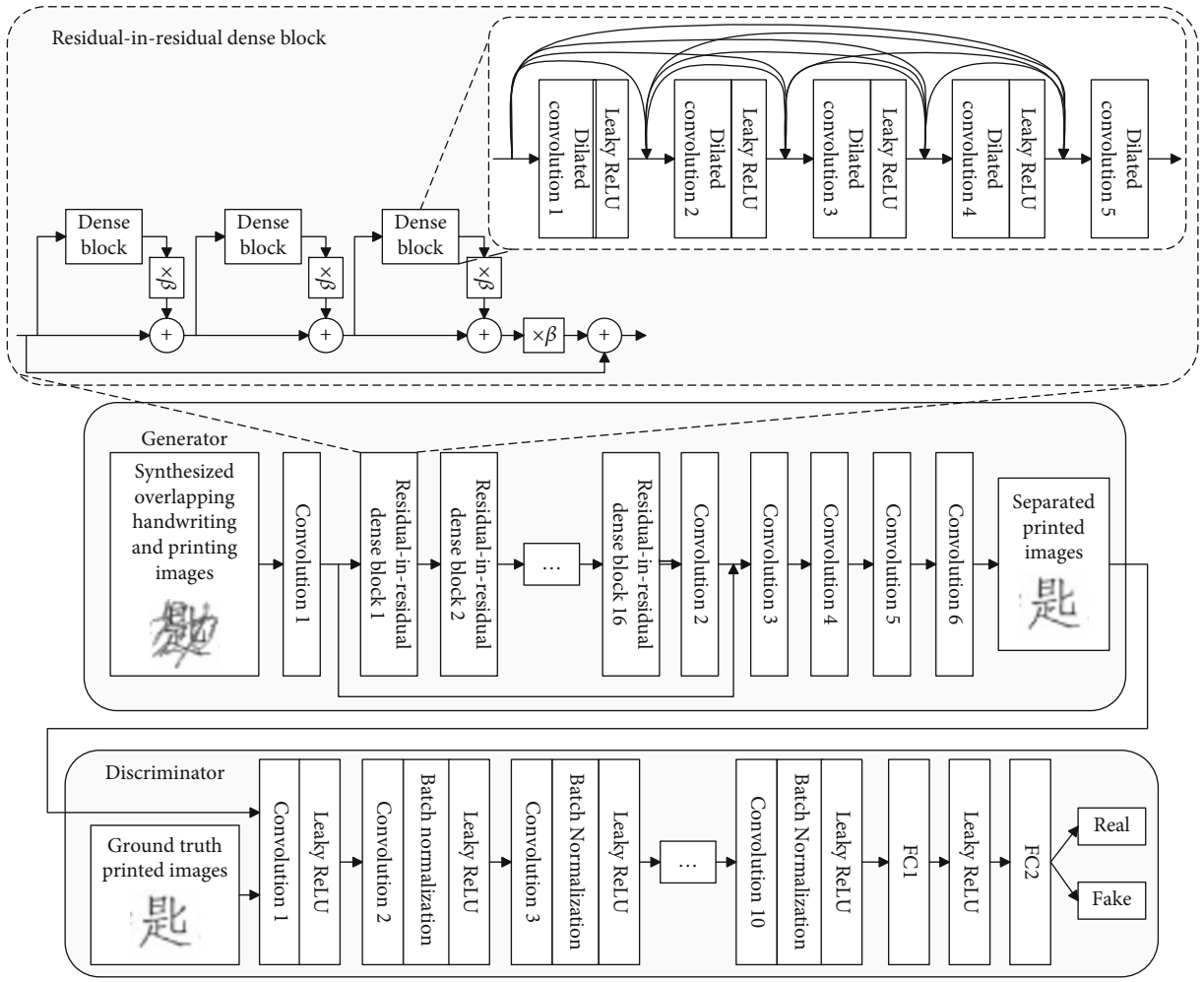

Figure 2: The network structure of the DESRGAN. An image with overlapping handwritten and printed characters is first processed by a series of convolution and RRDB modules. There is an operation of dilated convolution inside each residual-in-residual dense block. The generator outputs separated printed part or handwritten part from the overlapping. The discriminator classifies separated printed or handwritten characters and ground truth into real or fake.

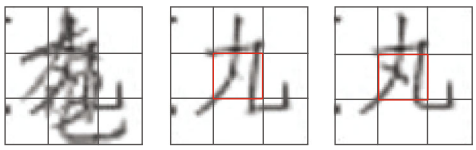

Figure 3: Image gridding (3×3 or 5×5) in the calculation of the integrity loss. From left to right: the overlapping image I^{OL}, the recovered image $G(I^{OL})$, and its ground truth I^{GT}. Note that our integrity loss will focus on the center cells of $G(I^{OL})$ and I^{GT} which are severely inconsistent.

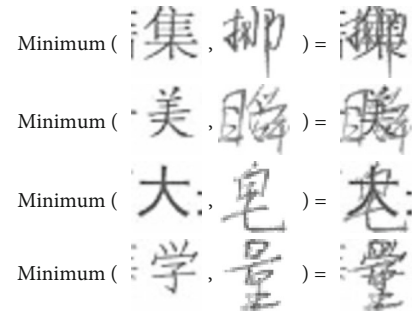

Figure 4: A simple example of image synthesis of overlapping of handwritten and printed characters. The three columns from left to right are printed Chinese, Chinese handwriting, and synthesized overlapping character.

loss by using features before activation instead of after activation. These techniques significantly improve the overall visual quality of reconstruction. Due to its versatility, GAN-based super-resolution techniques can potentially improve poor quality of document images, which is attributed to low scanning quality and resolution. Lat and Jawahar [24] super-resolve the low resolution document images before passing them to the OCR engine and greatly improve OCR accuracy on test images. However, we found that existing approaches could not provide satisfactory segregation results.

Besides, there are research efforts toward handwriting synthesis. Graves [25] utilizes Long Short-term Memory recurrent neural networks to generate highly realistic cursive

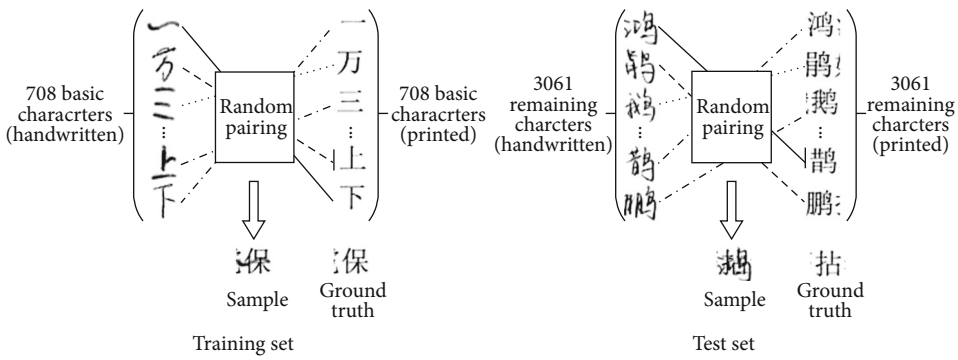

Figure 5: Data synthesis method. For each font type, 708 basic Chinese characters which contain various basic components are chosen to synthesize overlapping scenarios. Handwritten characters and printed characters are then randomly paired and overlapped to constitute data samples in training set. Synthesized in a similar way, the test set contains more Chinese characters and the character structure is more complex. Each data sample has a corresponding ground truth (i.e., original printed or handwritten character) for evaluating performance. The total size of the test set reaches approximately 12,200 unique overlapping characters.

Figure 6: Results of separating printed Chinese characters in Hei font. Columns from left to right: synthesized handwritten/printed overlapped data sample in test set and results of separation of printed characters by CycleGAN, Pix2pix, U-Net, ESRGAN, and DESRGAN. The ground truth and OCR recognition results are also provided. U-Net classifies each pixel into one of two categories and generates a binary image.

handwriting in a wide variety of styles. His algorithm employs an augmentation that allows the network to generate data sequences conditioned on some high-level annotation sequence (e.g., a character string). Lian et al. [10] propose a system to automatically synthesize personal handwriting for all (e.g., Chinese) characters in the font library. Their work showed feasibility of learning style from a small number (as few as 1%) of carefully selected samples handwritten by an ordinary person. Although the handwriting fonts produced by their models have better visual effects, their offline processing flow requires the preparation of the writing trajectory of each stroke for all characters, which requires a lot of manual effort. Zhang et al. [9] use a recurrent neural network as a generative model for drawing Chinese characters. Under their framework, a conditional generative model with character embedding is employed to indicate the RNN of the identity of the character to be generated. The character embedding, which is jointly trained with the generative model, essentially limits the model to search the characters with similar writing trajectory (or similar shape) in the embedded space. Chang et al. [26] formulate the Chinese handwritten character generation as a style learning problem. Technically, they use CycleGAN to learn a mapping from an existing printed font to a personalized handwritten style. Our work referred to these methods to construct a dataset for training and evaluating the proposed DESRGAN.

3. Design

Based on the GAN architecture, our method is shown in Figure 2. Given a Chinese character with noisy background (e.g., overlapping), the generative network separates the targeted character, which can be printed or handwritten, from the input image. Since each stroke in a Chinese character is almost indispensable, extra attention should be paid to maintaining the integrity of the Chinese character structure. We

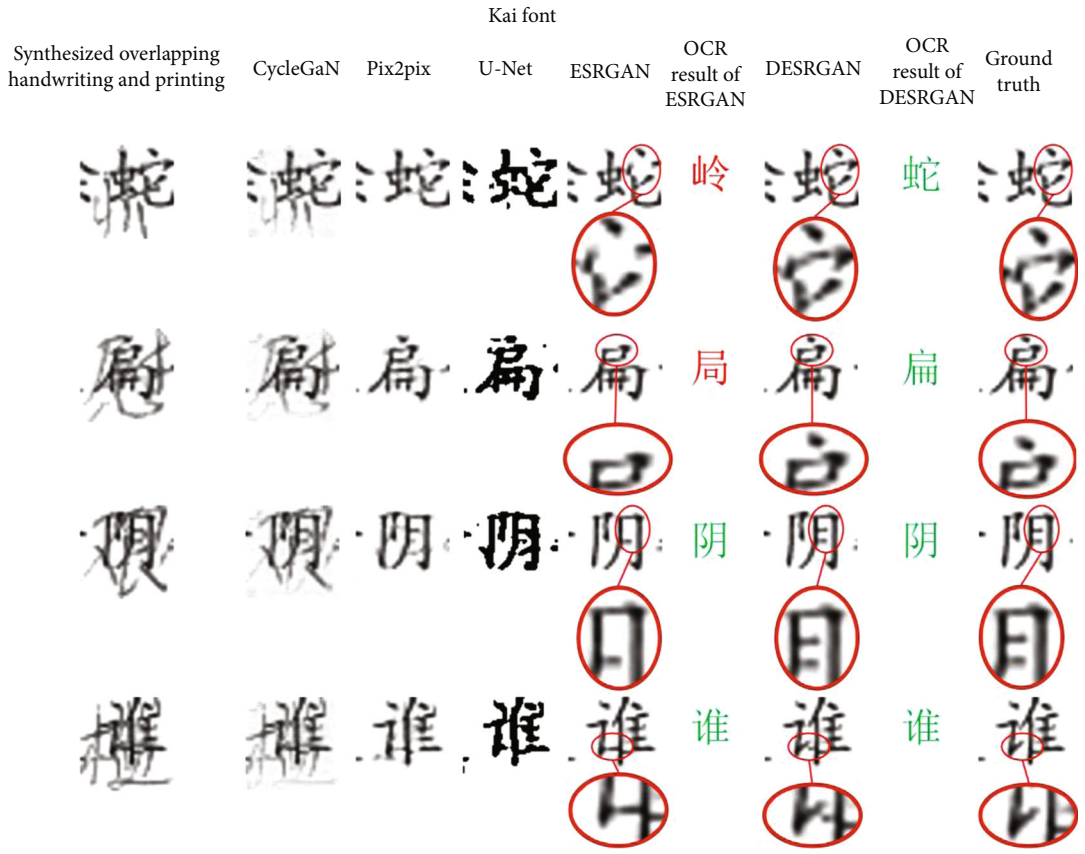

Figure 7: Results of separating printed Chinese characters in Kai font. Critical stokes or fine details of Chinese characters are clipped and enlarged for further examination. Wrong recognition results by OCR tool are colored in red while the correct ones in green.

Figure 8: Results of separating printed Chinese characters in Song font. Critical stokes or fine details of Chinese characters are clipped and enlarged for further examination. Wrong recognition results by OCR tool are colored in red while the correct ones in green.

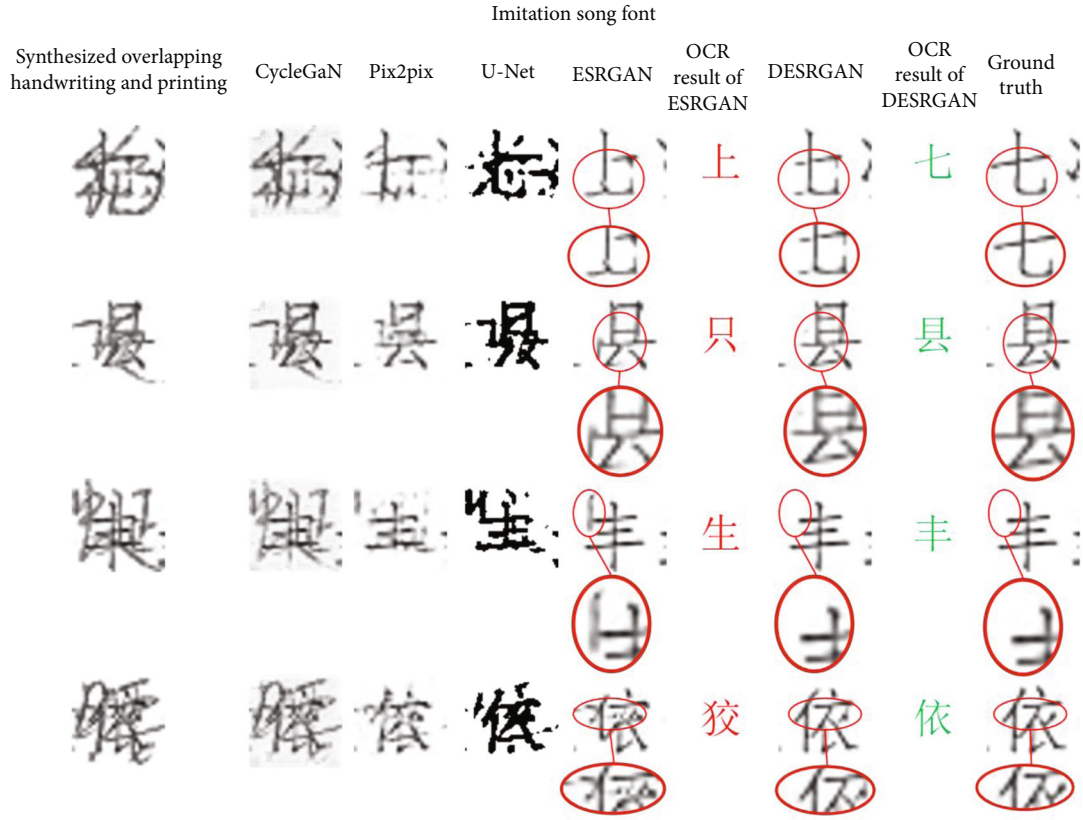

Figure 9: Results of separating printed Chinese characters in Imitation Song font. Critical stokes or fine details of Chinese characters are clipped and enlarged for further examination. Wrong recognition results by OCR tool are colored in red while the correct ones in green.

used a network structure similar to ESRGAN [19] as our backbone network, with major modifications. The dense connection structure of the ESRGAN generator network directly transfers the Chinese character strokes and skeleton information extracted from the intermediate layer to the subsequent layer. The proposed DESRGAN generator network removed the original upsampling layer of the ESRGAN and further replaced original convolution kernels with dilated convolution kernels. VGG19 was used to implement the discriminator which validates whether the image generated by the generative network is real or fake.

In ESRGAN, the loss function is a weighted sum of three components: a perceptual loss L_{percep} which measures the distance between the separated image and the ground truth image features before activation in pretrained VGG19, an adversarial loss L_G^{Ra} based on the probabilities of a relativistic discriminator, and a content loss $L_1 = \mathbb{E}_{x_i}\|G(x_i) - y\|_1$ which evaluates the 1-norm distance between separated printed or handwritten character image $G(x_i)$ and ground truth y.

The perceptual loss L_{percep} is defined as

$$L_{\text{percep}} = \left\|\phi(x_r) - \phi\left(x_f\right)\right\|_1, \tag{1}$$

where $\phi(\cdot)$ represents features before activation in pretrained VGG19, x_r stands for clean printed or handwritten character image, $x_f = G(x_i)$, and x_i stands for the mixed image of handwritten and printed characters.

The adversarial loss for generator is defined as

$$L_G^{\text{Ra}} = -\mathbb{E}_{x_r}\left[\log\left(1 - D_{\text{Ra}}\left(x_r, x_f\right)\right)\right] - \mathbb{E}_{x_f}\left[\log\left(D_{\text{Ra}}\left(x_f, x_r\right)\right)\right], \tag{2}$$

where $D_{\text{Ra}}(x_r, x_f) = \sigma(C(x_r) - \mathbb{E}[C(x_f)])$, σ is sigmoid function, $C(x)$ is the nontransformed discriminator output, and $\mathbb{E}_{x_f}[\cdot]$ represents the operation of taking average for all fake data in the mini-batch.

Perceptual loss L_{percep} plays an important role in computer vision tasks such as super resolution where the richness of the details of the recovered image is critical. It is designed to improve high-frequency details and avoid blurry and unpleasant visual effects. However, the goal we want to achieve here is to separate the printed part from the overlapped handwriting as much as possible and indirectly improve the recognition accuracy of subsequent OCR. With regard to this, we believe that the overall structure of the character and the integrity of the strokes are more important than the high-frequency details for OCR tools. Take the case in Figure 3 for example, due to the lack of a stroke in the center of the recovered image, OCR tools output the character "九" other than the correct one "丸".

FIGURE 10: Results of separating handwritten Chinese characters from superimposed printed characters in Imitation Song font. Columns from left to right: printed characters, synthesized handwritten/printed overlapped data sample in test set, results of ESRGAN and DESRGAN, and ground truth. Stokes or fine details of reconstructed handwriting are clipped and enlarged for further examination.

Therefore, a novel gradient-based loss term that can measure the integrity of the font skeleton was explored. Image gradients are powerful shape features, widely used in computer vision tasks. Given an overlapping image I^{OL}, the recovered image $G(I^{\mathrm{OL}})$, and its ground truth I^{GT}, the gradients of $G(I^{\mathrm{OL}})$ and I^{GT} were calculated, denoted as $\nabla G(I^{\mathrm{OL}})$ and ∇I^{GT}, respectively. Instead of relying on whole image level losses, we build on the ideas of gridding and max-pooling. As seen in Figure 3, the whole image area is divided into a square gird (3×3 or 5×5) of cells $\{C_i\}$, and the integrity loss was defined as the largest mean square error between $\nabla G(I^{\mathrm{OL}})$ and ∇I^{GT} of each cell C_i

$$L_{\mathrm{integrity}} = \max_{C_i} \frac{1}{W_i H_i} \sum_{x,y \in C_i} \left\| \nabla I^{\mathrm{GT}}_{x,y} - \nabla G\left(I^{\mathrm{OL}}\right) x, y \right\|^2, \quad (3)$$

where W_i and H_i are the width and height of cell C_i. With this strategy, the integrity of every cell of the skeleton is evaluated. The integrity loss will locate the cell with severe discrepancy between the recovered strokes and the ground truth strokes.

Therefore, the total loss for the generator is

$$L_G = L_{\mathrm{percep}} + \lambda L_G^{\mathrm{Ra}} + \eta L_1 + \alpha L_{\mathrm{integrity}}, \quad (4)$$

where λ, η, and α are the coefficients to balance different loss terms.

4. Experiment Settings

4.1. Dataset. In this work, we focus on character-level separation techniques. To process a document, some existing related technologies, such as layout analysis [27–29] and connected-component analysis [30], can help locate character positions. Therefore, we assume that there are some front-end modules that can help us roughly segment printed characters from a complete document. For the experiment, a dataset containing only overlapping printed and handwritten characters was created, as described below.

The handwritten character images used for synthesis come from the CASIA HWDB (CASIA Handwritten Database) 1.1 dataset [31], which contains images of 3755 commonly used Chinese character images written by 300 different writers. Specifically, we randomly chose handwritten images from four different writers (writer IDs 1003, 1062, 1187, and 1235) for synthesis.

The printed character images used in the synthesis include images of the same 3,755 commonly used Chinese characters listed in CASIA HWDB 1.1. In addition, basic symbols (plus sign, minus sign, equal sign, and answer box)

Figure 11: Results of separating handwritten characters from superimposed printed characters in Song font.

and Arabic numerals 0 to 9 were also added to the printed image dataset, which contains a total of 3,769 characters. For those 3,769 characters, they were printed on the A4 size paper in four fonts (Song, Hei, Kai, and Imitation Song). The printed paper was scanned and transferred to an image. Then, the scanned image was cropped at the character level to obtain the printed images for synthesis.

Existing researches on the pixel level separation of handwritten characters and printed characters are few. There is currently no publicly available dataset of overlapped handwritten and printed characters with pixel-level annotations. Therefore, this work uses handwritten character images and printed character images to synthesize samples of handwritten characters overlapped with printed characters. As Figure 4 shows, the method of image synthesis is to calculate the minimum value of the gray value of the pixels of the two images at the same position and use this minimum value as the gray value of the corresponding pixel of the composite picture.

The selected handwriting samples are paired with the printed samples by random matching. The final pairing results are as follows: printed Hei is randomly paired with the handwritten from CASIA HWDB writer ID 1003; printed Song is randomly paired with the handwritten from CASIA HWDB writer ID 1062; printed Kai randomly paired with the handwritten from CASIA HWDB writer ID 1187; and printed Imitation Song randomly paired with the handwritten from CASIA HWDB writer ID 1235.

On this basis, we refer to a 708 Chinese character set that contains various basic components of Chinese characters proposed by Lian et al. [10] in the study of handwritten Chinese character synthesis [10] (the collection of Chinese characters in their work contains a total of 775 characters, of which there are 67 unusual characters that are not in the CASIA HWDB dataset). Images containing these 708 Chinese characters also constitute the training dataset during the training phase. We assume that the model only needs to learn the features of these 708 Chinese characters to achieve the separation goal, rather than learning the features of all 3769 characters. Reducing the number of characters in the training set has a beneficial effect on both data collection effort and computation cost. Therefore, shown in Figure 5, 708 corresponding samples from handwriting samples and printed samples were selected for random pairing synthesis as the training set. The remaining samples are also taken as a test set by random pair synthesis. The resulting training set contains 2832 samples (708 samples for each font type), and the test set contains 12244 samples. This dataset is used as the dataset commonly used in all subsequent experiments.

Figure 12: Results of separating handwritten characters from superimposed printed characters in Kai font.

4.2. Evaluation Metrics. Both intersection over Union (IoU) and OCR accuracy were used as our evaluation metrics. The separation of handwritten characters and printed characters can essentially be regarded as the semantic segmentation of a mixed image of handwritten and printed characters. The most commonly used quantitative evaluation metric in image semantic segmentation is IoU. Therefore, IoU was used as one of the quantitative evaluation metrics for evaluating the separation quality of handwriting and printed characters.

Because this study is a pixel-level segmentation of handwritten printed mixed pictures, IoU is calculated by the number of pixels in the corresponding category (i.e., background and printed). Before calculating IoU, the image is first binarized, and the black area after binarization is regarded as the printed area, and the white area is regarded as the background area. In the process of binarization, the Otsu algorithm is first used to calculate the average binarization threshold of all printed samples in the test set, and this single threshold value is used for binarization of all samples. In our synthesized dataset, the average binarization threshold calculated by the Otsu algorithm for the test set is 184. Finally, we divide the intersection of the separated printed part (or background) and the printed part (or background) in ground truth by their union.

One of applications targeted by this study is the automatic grading of exam papers for primary and middle school students. Therefore, the main purpose of separating handwritten characters from printed characters in this research is to improve the recognition accuracy of the printed text and handwritten text and to prevent the deterioration of recognition of the printed text due to the interference of handwritten strokes or characters on the exam paper. Therefore, the accuracy of OCR for printed characters is used as another quantitative evaluation metric for the separation of handwritten and printed characters.

The OCR tool used to calculate the accuracy of OCR is Chinese_OCR [32], which is open sourced on Github. This model recognizes Chinese characters with high speed and high accuracy. It is very suitable for evaluating the accuracy of OCR of the separated printed samples. When calculating the accuracy of OCR, because Chinese_OCR cannot detect the text of a single character image, we had to first horizontally concatenate every 25 samples into a long image for OCR. Then, the correct number of characters was counted and identified according to the character order in the long image.

4.3. Model Training. The experiment was conducted on a PC with Intel Xeon E5-2603 v3@1.600GHz CPU, NVIDIA Tesla P40 24GB GPU, and 64GB memory. The PC runs the CentOS 7 operating system, and the deep learning framework used is PyTorch 1.2.0.

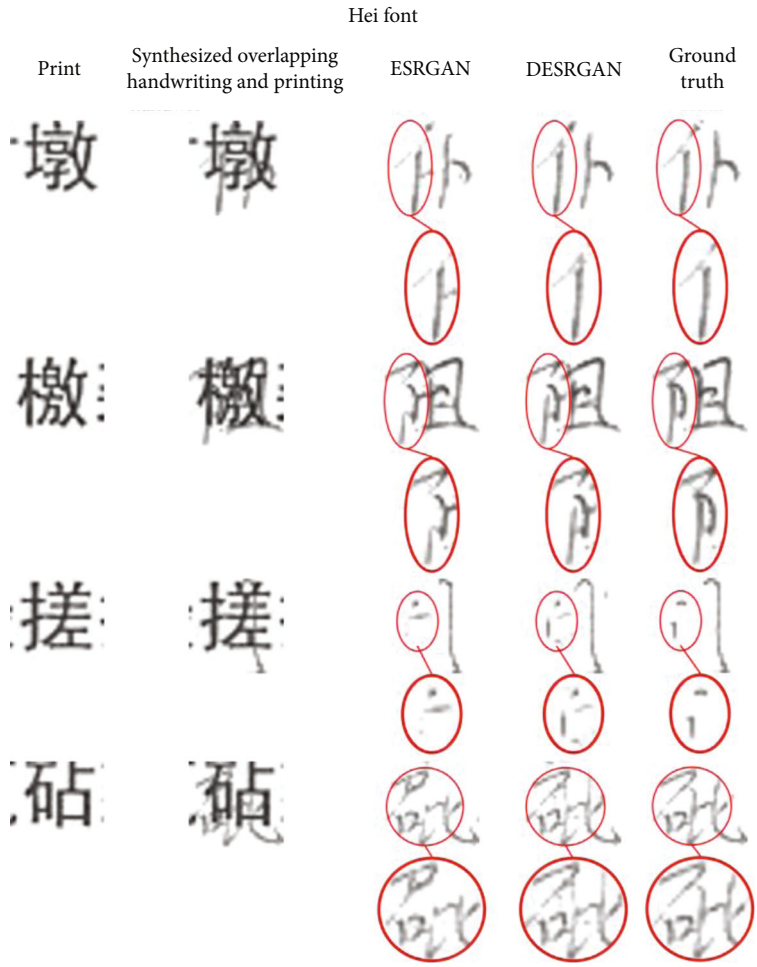

Figure 13: Results of separating handwritten characters from superimposed printed characters in Hei font.

In order to verify the effectiveness of the DESRGAN model, U-Net, Pix2pix, and CycleGAN which are commonly used in image semantic segmentation were selected as a comparison. Both Pix2pix and CycleGAN models were trained for 200 epochs with the batch size set to 1. The learning rate of the first 100 epochs remained at 0.0002, and the learning rate of the last 100 epochs decayed linearly to 0. U-Net was trained for 200 epochs with the batch size set to 100. The learning rate was initially 0.01 and then dropped to tenth of its value after every 50 epochs.

At the same time, we compared with ESRGAN to verify the effectiveness of the proposed modification to ESRGAN. Both ESRGAN and DESRGAN first used L1 loss to train their generators for 2500 epochs separately with the batch size set to 16. The initial learning rate was 0.0002 and then halved after every 200,000 iterations. Then, combined with the discriminator network, the training method of the GAN is used to train 2500 epochs with the batch size set to 16. The initial learning rate was 0.00001 and halved after the 50,000th, 100,000th, 200,000th, and 300,000th iterations. Throughout our work, the coefficient λ of adversarial loss L_G^{Ra} and the coefficient η of content loss L_1 are set to 0.005 and 0.01, respectively.

Table 1: IoU of the separation results of printed characters by different deep learning methods. The IoU of printed and the IoU of background are first calculated for each result against ground truth in the test set and then averaged. The overall IoU is the average of the first two values.

IoU	CycleGAN	Pix2pix	U-Net	ESRGAN	DESRGAN
Printed	0.631	0.755	0.697	0.905	0.911
Background	0.869	0.935	0.910	0.977	0.978
Overall	0.750	0.845	0.803	0.941	0.944

5. Results and Discussion

5.1. Visual Effects of Separation Results. We verified several deep learning methods including our proposed DESRGAN and visually compared the separation results of Chinese characters in the test set. To understand the performance of separating printed Chinese characters from noisy background, we synthesized handwritten/printed overlapped data samples and tested five methods (i.e., CycleGAN, Pix2pix, U-Net, ESRGAN, and DESRGAN). Figures 6–9 show the results of these five methods, along with the separation ground truth.

Table 2: Impact of the proposed loss function on IoU of the separated printed characters.

IoU	DESRGAN without $L_{\text{integrity}}$	DESRGAN + $L_{\text{integrity}}$ ($\alpha = 0.1$)	DESRGAN + $L_{\text{integrity}}$ ($\alpha = 1$)	DESRGAN + $L_{\text{integrity}}$ ($\alpha = 10$)
Printed	0.911	0.913	0.912	0.910
Background	0.978	0.979	0.978	0.978
Overall	0.944	0.946	0.945	0.944

To further understand the impact of a slight loss or increase of strokes in the separation result, the recognition results of OCR tool were additionally placed next to the separation results.

Four popular printed Chinese font types were tested: Hei, Kai, Song, and Imitation Song. For all tested font types, ESRGAN and DESRGAN gave the most visually pleasing results. As shown in Figures 6–9, other methods could not completely eliminate handwriting strokes in the separation result. However, not all fonts are designed equal and some of them (i.e., Song font and Imitation Song font) contain much thinner strokes. As a result, ESRGAN failed to reconstruct some seemly trivial strokes or remove artifacts which are harder to distinguish from the superimposed handwriting. The more complex the structure of Chinese characters or the greater the possibility of similar structures, the easier it is for OCR tools to predict seemly correct but substantially wrong results. Only DESRGAN gave separation results that produced most successful OCR predictions (see Figures S1-S4 in the Supplementary Material for more separation results of printed characters).

Since DESRGAN can effectively separate the printed part from the overlapping image, the next question is whether it is capable of the separating the handwritten part. In essence, this task is more difficult because the handwriting style varies from person to person. For this test, only ESRGAN and DESRGAN were compared because the other three methods produce poor results. We did not report recognition results because no suitable OCR tool for handwriting recognition was found. Figures 10–13 show the visual effect of separating handwritten parts from superimposed printed characters in four different font types. The best separation effect came from the Imitation Song font (Figure 10) and the Song font (Figure 11), while the Hei font gave the worst effect (Figure 13). We speculated that it is because the strokes of printed characters in Hei font are thicker and the colors are darker, which interfered more with handwritten characters. Nonetheless, DESRGAN produced less artifacts and reconstructed better character structure than ESRGAN (see Figures S5-S8 in the Supplementary Material for more separation results of handwriting).

5.2. Quantitative Analysis. As shown in Table 1, the IoU of the separation results of printed characters confirmed the previous visual effect comparison. Both ESRGAN and DESRGAN achieved better results than other deep learning methods, and DESRGAN has a small advantage over ESRGAN. It should be noted that through visual analysis, DESRGAN is better at restoring important details, which only account for a small part of the total pixels.

Table 3: IoU of the separation results of handwritten characters.

	ESRGAN	DESRGAN
IoU of handwriting	0.830	0.834
IoU of background	0.952	0.953
Overall IoU	0.891	0.894

In order to study the impact of the proposed loss function, we conducted experiments under three different settings (i.e., $\alpha = 0.1$, $\alpha = 1$, and $\alpha = 10$). Table 2 shows that the proposed loss function had almost no impact on IoU. This is in line with our expectations, because the main purpose of the new loss function is to improve the overall structure of the characters.

The IoU of separation of handwritten characters by ESRGAN and DESRGAN was also evaluated. Both models received more than 10,000 overlapping Chinese characters from which the handwriting parts were reconstructed individually. The IoU of separated handwriting and the IoU of separated background were first calculated for each result against ground truth in the test set. Table 3 shows that DESRGAN achieved slightly better IoU results.

Table 4 shows that the superimposed handwriting has a great negative impact on the accuracy of OCR. The worst synthesized overlapping in Imitation Song font only achieved zero accuracy. The separation results of ESRGAN and DESRGAN led to higher OCR accuracy than those of CycleGAN, Pix2pix, and U-Net, which proved the advantage of network structure in identifying characters from a noisy background. Furthermore, DESRGAN achieved the highest OCR accuracy in three fonts (i.e., Kai, Song, and Imitation Song) thanks to better preservation of the strokes and basic skeleton of Chinese characters. Compared to ESRGAN, the proposed method improved the OCR accuracy by more than 1% in Song font and Imitation Song font which are more difficult to handle due to their thin strokes and light colors after scanning. Except Imitation Song font, the recognition accuracy rate of the OCR tool for the separation results of DESRGAN has almost reached the level of recognition of ground truth.

The impact of the proposed loss function on OCR accuracy was also measured. As shown in Table 5, the proposed loss component improved the OCR accuracy, especially in the case of Imitation Song font. This result coincides with previous visual analysis, where the OCR results of Imitation Song font are susceptible to trivial loss in characters most.

Table 4: Recognition result of separated printed characters and ground truth by OCR tool.

	Hei font	Kai font	Song font	Imitation Song font	Average
Synthesized overlapping	0.288	0.001	0.020	0.000	0.077
CycleGAN	0.615	0.063	0.129	0.028	0.209
Pix2pix	0.909	0.831	0.507	0.345	0.648
U-Net	0.622	0.587	0.296	0.217	0.431
ESRGAN	0.966	0.962	0.925	0.889	0.936
DESRGAN	0.964	0.971	0944	0.903	0.945
Ground truth	0.968	0.981	0.976	0.981	0.977

Table 5: Impact of the proposed loss function on OCR accuracy of separated printed characters.

	Hei font	Kai font	Song font	Imitation Song font	Average
DESRGAN without $L_{\text{integrity}}$	0.964	0.971	0.944	0.903	0.945
DESRGAN + $L_{\text{integrity}}$ ($\alpha = 0.1$)	0.965	0.972	0.947	0.912	0.949
DESRGAN + $L_{\text{integrity}}$ ($\alpha = 1$)	0.964	0.972	0.936	0.910	0.946
DESRGAN + $L_{\text{integrity}}$ ($\alpha = 10$)	0.962	0.971	0.948	0.912	0.948

6. Conclusions

In summary, a method to separate Chinese characters from noisy background where other characters are likely to overlap was proposed. Our method reconstructed important strokes and retained the overall structure in the complex Chinese characters. The proposed method also allowed the OCR tool to achieve better recognition accuracy. Those findings may have great benefits to scenarios such as test paper autoscoring and advanced document analysis. Our future works include studying how color of handwriting impacts separation process and applying to other writing system.

Data Availability

Data of printed Chinese characters of four common fonts is available upon request. You may contact the corresponding author to request data. We do not own the dataset of handwritten Chinese characters (CASIA HWDB), and you may send data request to the owner.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This research was funded in part by Natural Science Foundation of Fujian Province of China (grant numbers 2019J01712, 2018H0025, and 2019J05099), in part by Xiamen Science and Technology Bureau (grant numbers 3502Z20183035, 3502Z20183037, and 3502Z20183038), in part by Li ShangDa Fund (grant number ZC2015008), and in part by Guandong Provincial Key Laboratory of Digital Signal and Image Processing Technology Open Project (grant number 2017GDDSIPL-02). We would like to thank Kaijin Cui for data collection work. We thank Guorong Cai for their valuable comments.

Supplementary Materials

We provide more separation results in the supplementary material for reference. In the results, we show the visual effect of separating the handwritten part and the printed part, respectively, of four different common Chinese fonts. *(Supplementary Materials)*

References

[1] R. Gomez, B. Shi, L. Gomez et al., "ICDAR2017 robust reading challenge on COCO-Text," in *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1435–1443, Kyoto, Japan, 2017.

[2] N. Nayef, F. Yin, I. Bizid et al., "ICDAR2017 robust reading challenge on multi-lingual scene text detection and script identification - RRC-MLT," in *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1454–1459, Kyoto, Japan, 2017.

[3] R. Zhang, Y. Zhou, Q. Jiang et al., "Icdar 2019 robust reading challenge on reading Chinese text on signboard," in *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1577–1581, Sydney, NSW, Australia, 2019.

[4] J. Jo, J. W. Soh, and N. I. Cho, "Handwritten text segmentation in scribbled document via unsupervised domain adaptation," in *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 784–790, Lanzhou, China, 2019.

[5] X. H. Li, F. Yin, and C. L. Liu, "Printed/handwritten texts and graphics separation in complex documents using conditional random fields," in *2018 13th IAPR International Workshop on Document Analysis Systems (DAS)*, pp. 45–150, Vienna, Austria, 2018.

[6] A. Kölsch, A. Mishra, S. Varshneya, M. Z. Afzal, and M. Liwicki, "Recognizing challenging handwritten annotations with fully convolutional networks," in *2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pp. 25–31, Niagara Falls, NY, USA, 2018.

[7] M. Alberti, L. Vögtlin, V. Pondenkandath, M. Seuret, R. Ingold, and M. Liwicki, "Labeling, cutting, grouping: an efficient text line segmentation method for medieval manuscripts," in *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1200–1206, Sydney, NSW, Australia, 2019.

[8] R. Smith, "An overview of the Tesseract OCR engine," in *Ninth international conference on document analysis and recognition (ICDAR 2007)*, vol. 2, pp. 629–633, Curitiba, Brazil, 2007.

[9] X. Y. Zhang, F. Yin, Y. M. Zhang, C. L. Liu, and Y. Bengio, "Drawing and recognizing Chinese characters with recurrent neural network," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 849–862, 2018.

[10] Z. Lian, B. Zhao, X. Chen, and J. Xiao, "EasyFont: a style learning-based system to easily build your large-scale handwriting fonts," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 1, pp. 1–18, 2018.

[11] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, Cham, 2015.

[12] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3431–3440, Boston, Massachusetts, USA, 2015.

[13] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2018.

[14] C. Chu, A. Zhmoginov, and M. Sandler, "CycleGAN, a master of steganography," 2017, https://arxiv.org/abs/1712.02950.

[15] I. Goodfellow, J. Pouget-Abadie, M. Mirza et al., "Generative adversarial nets," in *Advances in neural information processing systems*, no. article 26722680, 2014ACM, New York, 2014.

[16] J. Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2242–2251, Venice, Italy, 2017.

[17] P. Isola, J. Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976, Honolulu, Hawaii, USA, 2017.

[18] C. Ledig, L. Theis, F. Huszár et al., "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4681–4690, Honolulu, Hawaii, USA, 2017.

[19] X. Wang, K. Yu, S. Wu et al., "Esrgan: enhanced super-resolution generative adversarial networks," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, Munich, Germany, 2018.

[20] S. Brunessaux, P. Giroux, B. Grilhères et al., "The maurdor project: improving automatic processing of digital documents," in *2014 11th IAPR International Workshop on Document Analysis Systems*, pp. 349–354, Tours, France, 2014.

[21] W. R. Huang, Y. Qi, Q. Li, and J. Degange, "DeepErase: weakly supervised ink artifact removal in document text images," 2019, https://arxiv.org/abs/1910.07070.

[22] J. K. Guo and M. Y. Ma, "Separating handwritten material from machine printed text using hidden Markov models," in *Proceedings of sixth international conference on document analysis and recognition*, pp. 439–443, Seattle, WA, USA, 2001.

[23] K. Zagoris, I. Pratikakis, A. Antonacopoulos, B. Gatos, and N. Papamarkos, "Distinction between handwritten and machine-printed text based on the bag of visual words model," *Pattern Recognition*, vol. 47, no. 3, pp. 1051–1062, 2014.

[24] A. Lat and C. V. Jawahar, "Enhancing OCR accuracy with super resolution," in *2018 24th International Conference on Pattern Recognition (ICPR)*, pp. 3162–3167, Beijing, China, 2018.

[25] A. Graves, "Generating sequences with recurrent neural networks," 2013, https://arxiv.org/abs/1308.0850.

[26] B. Chang, Q. Zhang, S. Pan, and L. Meng, "Generating handwritten Chinese characters using CycleGAN," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 199–207, Lake Tahoe, NV, USA, 2018.

[27] Y. Li, Y. Zou, and J. Ma, "Deeplayout: a semantic segmentation approach to page layout analysis," in *International Conference on Intelligent Computing*, pp. 266–277, Cham, 2018.

[28] X. Yang, E. Yumer, P. Asente, M. Kraley, D. Kifer, and C. Lee Giles, "Learning to extract semantic structure from documents using multimodal fully convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5315–5324, Honolulu, Hawaii, USA, 2017.

[29] Y. Xu, F. Yin, Z. Zhang, and C. L. Liu, "Multi-task layout analysis for historical handwritten documents using fully convolutional networks," in *International Joint Conference on Artificial Intelligence*, pp. 1057–1063, Stockholm, Sweden, 2018.

[30] F. Chang, C. J. Chen, and C. J. Lu, "A linear-time component-labeling algorithm using contour tracing technique," *Computer Vision and Image Understanding*, vol. 93, no. 2, pp. 206–220, 2004.

[31] C. L. Liu, F. Yin, D. H. Wang, and Q. F. Wang, "CASIA online and offline Chinese handwriting databases," in *2011 International Conference on Document Analysis and Recognition*, pp. 37–41, Beijing, China, 2011.

[32] YCG09, *Chinese_OCR*October 2019, https://github.com/YCG09/chinese_ocr.