Intelligent Automation & Soft Computing
DOI: 10.32604/iasc.2022.027146
Article

Tech Science Press

End-to-end Handwritten Chinese Paragraph Text Recognition Using Residual Attention Networks

Yintong Wang[1,2,*], Yingjie Yang[2], Haiyan Chen[3], Hao Zheng[1] and Heyou Chang[1]

[1]School of Information Engineering, Nanjing Xiaozhuang University, Nanjing, 211171, China
[2]Institute of Artificial Intelligence, De Montfort University, Leicester, LE1 9BH, United Kingdom
[3]College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, 210016, China
*Corresponding Author: Yintong Wang. Email: wangyintong@nuaa.edu.cn

Received: 11 January 2022; Accepted: 24 February 2022

Abstract: Handwritten Chinese recognition which involves variant writing style, thousands of character categories and monotonous data mark process is a long-term focus in the field of pattern recognition research. The existing methods are facing huge challenges including the complex structure of character/line-touching, the discriminate ability of similar characters and the labeling of training datasets. To deal with these challenges, an end-to-end residual attention handwritten Chinese paragraph text recognition method is proposed, which uses fully convolutional neural networks as the main structure of feature extraction and employs connectionist temporal classification as a loss function. The novel residual attention gate block is more helpful in extracting essential features and making the training of deep convolutional neural networks more effective. In addition, we introduce the operations of batch bilinear interpolation which implement the mapping of two dimension text representation to one dimension text line representation without any position information of characters or text lines, and greatly reduce the labeling workload in preparing training datasets. In experimental, the proposed method is verified with two widely adopted handwritten Chinese text datasets, and achieves competitive results to the current state-of-the-art methods. Without using any position information of characters and text line, an accuracy rate of 90.53% is obtained in CASIA-HWDB test set.

Keywords: Handwritten text recognition; residual attention; convolutional neural networks; batch bilinear interpolation; connectionist temporal classification

1 Introduction

Handwritten Chinese text recognition (HCTR) is a challenging issue and has received significant attention from many researchers [1–3]. There can be generally attributed to three important aspects. Firstly, the rapid growth of the HCTR application requirements includes mailing address recognition, office handwriting document recognition and precious historical manuscript processing. Secondly, the inherent long-term complexity of the handwriting recognition that involves thousands of character categories, variant writing style and complex space structure for characters or lines. Thirdly, the available training dataset is often insufficient to cover the handwriting style widely from different writers, which is necessary for obtaining deep neural networks with good performance. In reality, the labeling of handwritten Chinese text data containing the position information of single character or multi-characters line is expensive and error-prone. To this end, the inherent characteristics of handwritten text, such as the significant difference in individual handwriting styles, the extremely similar characters in large number of character categories and complex text structures, make it still an open research problem.

From the initial single character recognition [4,5] to the current mainstream text line recognition [6–8], the field of HCTR has observed tremendous progresses for the past several decades, and text recognition has becoming a development trend reducing explicit segmentation proposal in conducive to increase multi-characters sequence recognition. There are many approaches learn to both simultaneously segment and recognize a handwritten text image representing a sequence of observations [3,9–11]. It is well known that most advanced text recognition methods work on an entire input text line image without any explicit segmentation information for character or word. There is no doubt that this eliminates the requirement to provide word or character position information as part of ground-truth transcription. In addition, we known that the available mainstream methods have their own characteristics, which can be generally divided into convolutional neural networks (CNN) [12], recurrent neural networks (RNNs) [13], Long Short Term Memory networks (LSTM) and CNN-RNNs methods [7]. Text images, especially text lines, can be recognized as a series of character sequences, and RNNs can use their internal state to process variable length sequences of inputs. Undoubtedly, the RNNs-based method can be applied to text recognition, and many similarly advanced methods have obtained good recognition results [14–16].

Although the above mention methods have been successful to some extent, one can spot some shortcomings in these works. First, the text line segmentation is much easier than word or character segmentation, but the former is still faces the risk of error segmentation, which will lead to serious deterioration of HCTR performance. Second, the RNNs-based method struggles to make full use of long sequence relationships in real text recognition. The long range dependencies of pure-visual text recognition is sometimes not significant, only local neighborhoods will affect the final frame or character recognition results [17–19]. Third, the RNNs-based method brings non-trivial latencies and is unfriendly to parallel computing due to its sequential processing nature, and variable saving in iterative operations consumes a lot of computing resources. Therefore, segmentation-free recurrent-free model architectures can better confront the complex text structure and utilize the parallel computing with limited computing resources to achieve efficient HCTR.

To address the efficiency restriction of recurrent architectures and the adverse effect of segmentation on recognition performance, we propose an end-to-end handwritten Chinese paragraph text recognition using residual attention networks. The Chinese paragraph text image is processed by residual attention convolution without any character or line segmentation. The batch bilinear interpolation process is employed to obtain one-dimensional feature representation, and connectionist temporal classifier (CTC) [20] loss is adopted in the training process. The main contributions of this work can be summarized as follows: (1) recurrent-free architecture for handwritten Chinese paragraph text recognition is presented, which not only can effectively avoid the large delay problem due to the recurrent iterative operations, but also can make full use of the parallelization capabilities in training. (2) Residual attention gate block combines the advantages of residual framework and attention mechanism, extracting the representative features can alleviate the problems of gradient disappearance or explosion for deeper convolutional neural networks. (3) Batch bilinear interpolation is utilized to encourage mapping each character of the input 2D image to the distinct part of the output 1D text line without losing information, and then converts the text recognition from single-line multi-characters recognition to multi-lines multi-characters recognition.

The rest of the paper is organized as follows: Section 2 reviews related work. In Section 3 introduces the proposed end-to-end handwritten Chinese paragraph text recognition. In Section 4 presents the experimental results and analysis. Finally, we give the conclusions and future works in Section 5.

2 Related Works

Handwritten Chinese text recognition, as a challenging issue in the field of pattern recognition faces variant writing style, thousands of character categories and text-line touching, has received widespread attention. In this section, we briefly summarize the development process of handwritten recognition covering the single-character method, the single-line multi-characters method and the multi-lines multi-characters method, so as to provide an outline of this research field as a whole.

The single-character method involves feature extraction and classification of single character images. A typical single-character recognition model mainly focuses on preprocessing, feature extraction and classification [5,21]. As we know, the performance of these methods has reached a bottle neck due to the restricted ability of handcrafted representation features. Modified quadratic discriminant function (MQDF) [22] as the representation of earlier successful method for single-character recognition has been surpassed by neural networks methods [23,24]. The CNN-based methods consist of multiple layers of convolution operation and pooling operation for automatic feature learning and fully connected layers for classification, then learn high-level relevant features through deep hierarchical structure and achieve the recognition accuracies comparable to human discernment. Li et al. [25] inspired by the human cognition process of handwritten characters, and presented a matching network making a connection between handwritten characters and template characters. Li et al. [26] improved GoogLeNet based on deep convolutional generative adversarial network. Overall, these approaches have achieved good recognition results, and even successfully completed the challenging task of handwriting character recognition with superhuman recognize ability. However, incorrect character segmentation exists as the inherent limitation for the single-character recognition method and brings great difficulties to subsequent recognition.

The single-line multi-characters method realizes handwriting text recognition from a text line image to a character sequence, which eliminates the adverse effects from the decrease in recognition performance caused by incorrect character or word segmentation, while also reduces the labeling work of training samples which is usually time-consuming and expensive. The text line recognition approaches can generally be grouped into segmentation-free approaches [8,27] and over-segmentation approaches [9,28]. The over-segmentation approaches by integrating the character classifier, topological path solving and language context model has achieved the success in offline handwritten text recognition. Wang et al. [29] introduced the over-segmentation method from the Bayesian decision view and convert the classifier outputs to posterior probabilities via confidence transformation. Wang et al. [28] presented a deep network using heterogeneous CNN to obtain hierarchical supervision information from the segmentation candidate lattice. In contrast, the segmentation-free approaches no longer require explicitly segmentation of text lines into single character. The early methods are based on Gaussian mixture model, among which Hidden Markov Model(HMM) is the most representative [27]. HMM based method will also face many parameters in the growth of recognition character length, which leads to the decline of recognition performance. Recently, deep learning has successfully accomplished the challenging task of specific image classification with superhuman accuracy. Messina et al. [30] proposed multidimensional LSTM-RNN using CTC as loss function for end-to-end text line recognition. Wu et al. [15] employed separable multi-dimensional LSTM and RNN with connectionist temporal classifier to replace the traditional LSTM-CTC model. End-to-end handwritten Chinese text line recognition methods [8,31] are proposed. Considering RNN-based methods demand significant computing resource and lack the parallelization capability in the model training phase, it is reasonable to the shift from recurrence neural network to recurrence-free one in recent text sequence recognition modeling works.

The multi-lines multi-characters method, as a method for full page text recognition, transforms the input text image into a character sequence without any position information of characters and text line, which is much cheaper for labeling data and more efficient text image recognition [14,18,32,33]. This position information is indispensable for most current handwritten recognition, and expensive and error-prone in the labeling work of training sample. Traditional methods on full page text recognition consist of multi-modules for text line detection, segmentation and recognition respectively. Moysset et al. [34] presented a character system using RNN and weighted finite-state transducers for text recognition. Wigington et al. [35] proposed a multi-lines text recognition, in which the region proposal network use to get the starting positions of text-line. Tensmeyer et al. [36] proposed a weakly supervised start-follow-recognition method to realize the alignment between predicted character sequence and real character sequence. These approaches consist of several independently pre-trained modules, and hardly achieved the expected output of the whole recognition model. Under this background, the end-to-end full page text recognition is proposed which gradually compresses the text into a whole line of feature maps for recognition. Bluche et al. [14,37] presented an attention-based model for end-to-end English handwriting paragraph recognition transforming the 2D text into a 1D text-line by the collapse CNN layers. Mohamed et al. [18] proposed a unified multi-line text recognition framework, which can transform the CNN-based single text line recognition into multi-lines text recognition. Since the implicit segmentation based approaches can surmount the two main deficiencies of traditional handwritten full page text recognition methods, they have become prevalent recently.

3 Methodology

In this Section, we provide a summary of end-to-end handwritten Chinese paragraph text recognition method. First, Section 3.1 shows the framework of the proposed model architecture. Then, Section 3.2 proposes derivation of the residual attention gating mechanism. After that, Section 3.3 introduces the batch bilinear interpolation mapping 2D text representation into 1D text-line representation without losing information. Finally, Section 3.4 describes our model design.

3.1 Framework

By integrating attention mechanism into residual networks, the proposed method can improve the ability of the text recognition model to learn the meaningful representative features from the paragraph text image, and alleviate the gradient disappearance or explosion for deep neural networks. In addition, our method introduces text-line up-sampling to implement the mapping from 2D input signal to 1D output signal which is a vital processing for end-to-end text recognition. More importantly, our model only contains convolutional operations, and improves the tremendous need for computing time and storage space due to recursive or loop operations during model training. Fig. 1 gives the implementation process of end-to-end handwritten Chinese paragraph text recognition, which contains four important parts: preprocessing, feature extraction, text-line up-sampling and text recognition. The details are as follows:

First, **preprocessing operation** mainly realizes the increase the number of channels without changing the height and width of the input paragraph text image. The image with $h \times w \times 3$ is processed with 1×1 convolution to achieve a new tensor with 29 channels. Then, it is normalized by softmax to promote the convergence speed in training. And then, each channel of the tensor is processed independently by a 13×13 filter using depthwise convolution. Finally, the result is concatenated with the normalized tensor from the input paragraph text image, and the size of the output tensor is $h \times w \times 32$.

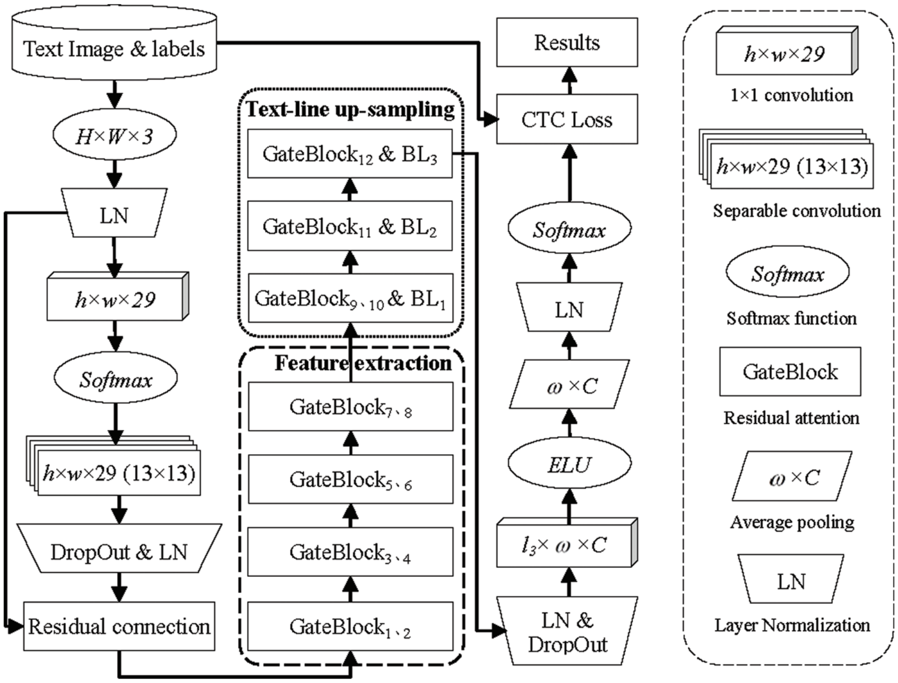


Figure 1: The flowchart of end-to-end handwritten Chinese paragraph text recognition

Second, **feature extraction** refers to the process of high-level representative feature extracting from original text image through stacked residual attention gate blocks. The residual attention gate blocks (GateBlock) as the critical computational block of our model consists of residual mechanism and attention mechanism. In which, we employ separable convolution operation to realize fast representative feature extraction. Referring to the architectural details of our model in Fig. 4, there are four network layers: conv1.x-4.x. Each network layer consists of two GateBlocks and a max pooling. The number of tensor channels is from 32 to 1024, and the width and height of tensor is from $h \times w$ to $h/8 \times w/16$.

Third, **text-line up-sampling** refers to the process of mapping from two-dimensional (2D) text representation to one-dimensional (1D) text line representation through GateBlock and batch bilinear interpolation without losing information. Referring to the architectural details of our model in Fig. 4, there are three layers, conv5.x-7.x. In which, the first layer consists of two GateBlocks and a batch bilinear interpolation, and the other layers include a GateBlock and a batch bilinear interpolation. The size of tensor is from $h/8 \times w/16$ to $l_3 \times w/128$, and the number of channels is from 1024 to 256. In the above steps, we use the combination to guide the batch bilinear interpolation to achieve the mapping of sequences of different sizes.

Finally, **text recognition** refers to classify the high-level representative feature sequence obtained by extracting from the input paragraph text. As the top-level operation of our model, CTC can realize the training of neural network recognizer on the free-segmentation paragraph text by considering all possible alignment between two 1D representation sequences. Not only realize the prediction from representative feature sequence to character sequence, but also use its spatial model and strong linear prior knowledge to induce the model to implement text-line up-sampling.

3.2 Residual Attention Gate Block Modeling

The proposed method is to stack multiple residual attention gate blocks as one of the important calculation blocks. The attention mechanism is used to adjust the inter-layers information flow so as to reduce the importance of irrelevant features and increase the importance of meaningful features through weighting. This mechanism has received many interests from visual processing, and combines it with the residual neural networks structure in deep CNN to improve the convergence of the model has become a research focus. We consider rebuilding the gating mechanism on attention mechanism and residual networks [38]. Fig. 2 gives the detailed working principle of the *i*-th GateBlock.

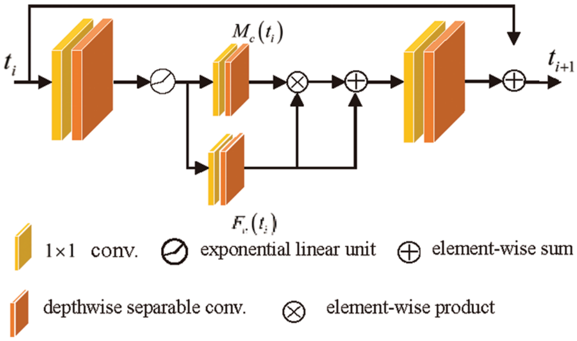


Figure 2: The structure of the *i*-th GateBlock

Let $T = \{t_1, t_2, \cdots, t_m\}$ be the output tensor set for each network layer, where *i*-th tenor is $t_i \in \mathbb{R}^{h' \times w' \times 2^k}$, h', w' and k are the height, width and channels of the output tensor, respectively. Mapping function $H(\cdot)$ implements the input tensor of one network layer to the output tensor. That is, for *i*-th network layer, the input tensor and the output tensor are t_i and t_{i+1}=$H_c(t_i)$ respectively. In order to implement the original goal of the residual attention gate block, we design a novel mapping function that integrates the advantages of both the attention mechanism and the residual neural networks. This formula is as follows:

$$H_c(t_i) = [M_c(t_i)+1] \times F_c(t_i)+t_i \tag{1}$$

where, $H_c(t_i)$ indicates the c channel of the output tensor mapped from the input tensor, $M(\cdot)$ and $F(\cdot)$ represent the mask branch and the trunk branch function, respectively. Doing an identity transformation on the Eq. (1), we can obtain the following the expression:

$$H_c(t_i)=M_c(t_i) \times F_c(t_i)+F_c(t_i)+t_i \tag{2}$$

In Eq. (2), let $H'_c(t_i)$=$M_c(t_i) \times F_c(t_i)$ and $H''_c(t_i)$=$F_c(t_i)+t_i$, and we get:

$$H_c(t_i) = H'_c(t_i)+H''_c(t_i) \tag{3}$$

The Eq. (3) consists of two key functions, where $H'_c(t_i)$ indicates the attention module of the GateBlock. For the *i*-th layer, the gate block from the input tensor t_i to the output tensor t_{i+1} includes two mapping functions, $M_c(t_i)$ indicates the mask mapping function of the attention module, and $F_c(t_i)$ indicates the main mapping function of the attention module. In the process of feature extraction, they work together to favor the meaningful feature. Besides, $H''_c(t_i)$ represents the residual neural network structure, which also consists of two mapping functions, $F_c(t_i)$ and t_i, respectively. Where $F_c(t_i)$ represents the output tensor obtained by the input tensor t_i and its mapping function $F_c(\cdot)$.

For making effective utilize highway gate stacking for deep neural network, we must to regard the dimensionality problem of different network layers to implement the residual connection. From Eq. (1), we know that $M_c(t_i)$ is firstly plus 1, and then multiplied by the mapping function $F_c(t_i)$, finally plus the input tensor t_i. Referencing the dual transformation mappings in Ref. [6], P_1 be a negative transformation mapping $x \in \mathbb{R}^{H \times W \times C}$ to $x' \in \mathbb{R}^{H' \times W' \times C'}$, and P_2 be a positive transformation mapping $x' \in \mathbb{R}^{H' \times W' \times C'}$ to $x \in \mathbb{R}^{H \times W \times C}$. Then Eq. (1) can be rewritten as:

$$\begin{cases} t'_i = P_1(t_i) \\ t_{i+1} = P_2([M_c(t'_i) + 1] \times F_c(t'_i)) + t_i \end{cases} \tag{4}$$

The negative transformation mapping P_1 can ensure that the model maintains the optimization benefit of the residual structure while calculating the residual attention for different dimensional tensor t_i. It is noteworthy that the depthwise separable convolution as the mainly operation in P_1 and P_2 mapping functions, and Exponential Linear Unit(ELU) as an activation function of neural network. Besides, three parameters of tensor, $C' = \alpha C$, $H' = H$ and $W' = W$, mean that up-sampling or down-sampling of tensor transformation is only on the tensor channels. The expansion parameter α is an exponential value with base 2, that is $\alpha = 2^{k'}$, $k' \in [-3, -2, -1, 0, 1]$. Different values of k will produce different GateBlock performances, $k' < 0$ indicates that number of tensor channels is reduce, and then GataBlock uses less memory and executes faster; $k' = 0$ means that number of tensor channels is constant; $k' > 0$ indicates that number of tensor channels is increase, and then more feature information participate in convolution operation in GataBlock to obtain better feature representation ability.

3.3 Text-line Up-sampling Modeling

Text-line up-sampling is used to solve settle the problem of multiple character recognition in a vertical direction in paragraph text recognition. It transforms the 2D text representation into the 1D text line representation which is sufficient to accommodate all characters. For the original text image containing L characters, the real sequence length of the two-dimensional space obtained by representative feature extraction is L, and the sequence length of the one-dimensional space is $L'(L \leq L')$. In other words, the one-dimensional space should be large enough, so that the sequence on the two-dimensional space and the sequence on the one-dimensional space can present a one-to-many relationship, and maintain the neighbor relationship between the original sequences [14,18,39]. Therefore, batch bilinear interpolation is introduced in the text-line up-sampling, and the mapping from two-dimensional text representation to one-dimensional text line representation is progressively realized through three tensors with lengths l_1, l_2 and l_3 respectively.

Bilinear interpolator is a linear reconstruction filter with the distance adaptation ability [40,41]. For the polynomial interpolators, only the first degree always estimates an output observation value between minimum and maximum. This property of bilinear interpolation can avoid negative coefficients. One of our works is to propose a batch bilinear interpolation (BL), which uses vector matrix operations to obtain multiple insertion point values at a time. Fig. 3 is an estimation scenario where the red dots represent the original data I, and its coordinates is $X = \{x_1, x_2, \cdots, x_m\}$ and $Y = \{y_1, y_2, \cdots, y_n\}$. The blue dots are the under-estimate points I', and its coordinates is $X' = \{x'_1, x'_2, \cdots, x'_{m-1}\}$ and $Y' = \{y'_1, y'_2, \cdots, y'_{n-1}\}$. The initial image is only given at discrete points in $\left(x'_i, y'_j\right)$. To obtain an interpolated value at an any point demands must to locate the four grid points around the point $\left(x'_i, y'_j\right)$. The new observation point is situated between x_i and x_{i+1} in $X(i < m)$, and between y_j and y_{j+1} in $Y(j < n)$. The values of the input tensor at the four grid points around I' are I_{11},I_{12},I_{22} and I_{21} respectively. Refer to [41], there are four condition parameters and the best interpolation achievable with four grid points is a bilinear form:

$$I' = A + BX' + Y'C + DY'X' \tag{5}$$

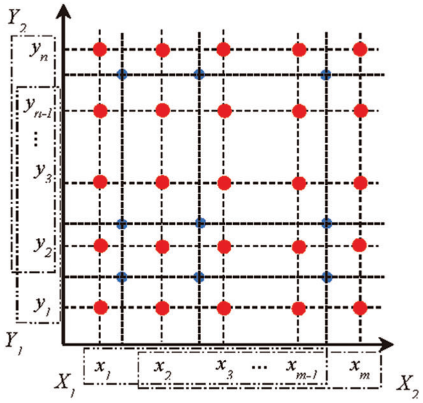


Figure 3: Example of batch bilinear interpolation

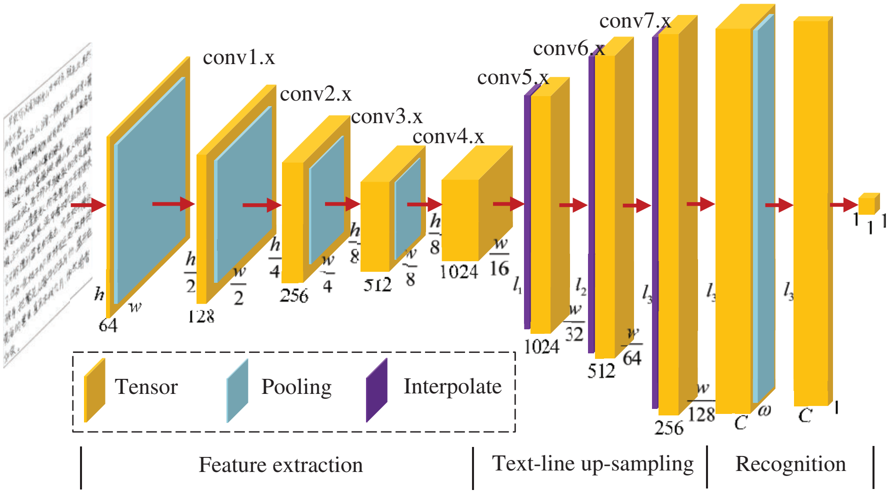


Figure 4: The residual attention convolutional architecture of our method

The A, B, C and D coefficient parameters are decided by the surrounding tensor values as:

$$\begin{cases} I_{11} = A + BX_1 + Y_1C + DY_1X_1 \\ I_{12} = A + BX_1 + Y_2C + DY_2X_1 \\ I_{21} = A + BX_2 + Y_1C + DY_1X_2 \\ I_{22} = A + BX_2 + Y_2C + DY_2X_2 \end{cases} \tag{6}$$

Combining pairs of equations yields:

$$\begin{cases} I_{11} - I_{12} = Y_1C + DY_1X_1 - Y_2C - DY_2X_1 \\ I_{21} - I_{22} = Y_1C + DY_1X_2 - Y_2C - DY_2X_2 \end{cases} \tag{7}$$

The coefficient parameters can be obtained one at a time. Subtracting Eq. (7) yields a value for D:

$$I_{11} - I_{12} - I_{21} + I_{22} = DY_1X_1 - DY_2X_1 - DY_1X_2 + DY_2X_2 = D(Y_1 - Y_2)(X_1 - X_2) \tag{8}$$

From the invertible property of the matrix, $(Y_1 - Y_2)$ and $(X_1 - X_2)$ are meaningful, so the $(Y_1 - Y_2)(X_1 - X_2)$ has an inverse matrix $[(Y_1 - Y_2)(X_1 - X_2)]^{-1}$. And then Eq. (8) can transformed as

$$\begin{cases} D = (I_{11} - I_{12} - I_{21} + I_{22})[(Y_1 - Y_2)(X_1 - X_2)]^{-1} \\ C = (I_{11} - I_{12})(Y_1 - Y_2)^{-1} - DX_1 \end{cases} \tag{9}$$

Similarly, the A and B coefficients can be found.

$$\begin{cases} B = (I_{11} - I_{21})(X_1 - X_2)^{-1} - DY_1 \\ A = I_{11} - BX_1 - CY_1 - DY_1X_1 \end{cases} \tag{10}$$

Now given the values for all four coefficient vectors (A, B, C and D) the values for the New tensor I at all point inside their four grid points can be found from Eq. (5).

According to batch bilinear interpolation, the text-line up-sampling model can be called soft line segmentation convolutional neural networks. The characteristics over the convolutional neural networks trained for line segmentation are that (i) it works on the same features as those used for the written text line recognition; (ii) it is trained to maximize the transcription accuracy, which is more closely related to the object of handwriting full page text recognition.

3.4 Model Design and Implementation

As shown in Fig. 4, our model includes feature extraction, text-line up-sampling, and recognition. Feature extraction uses a residual attention mechanism, which is widely used in deep convolutional neural networks. Text-line up-sampling originated from the problem of super-resolution image processing, and then it was introduced into text handwritten text recognition. We propose three layers of combination GateBlock and batch bilinear interpolation to realize the mapping of the 2D text representation to the 1D text line representation. The initial input tensor is height stretched and width compressed, and it height is l_1, l_2 and l_3 respectively. In recognition, CTC is employed to classify the high-level representative feature sequence to the prediction character sequence. Our model is mainly inspired by residual attention neural network [18], which is widely applied as the backbone neural network and has been proven to have outstanding performance for visual processing tasks. The neural networks consists of successive convolution layers with residual attention gate blocks, they are depth-separable convolution with 3 × 3 kernel size. Each networks layer is followed by layer normalization and batch normalization during training, and ELU as a non-linearity activation function has effectively robustness. The size of text image is uniformly processed as $h \times w$, such as 2100 × 2400.

4 Experimental Results and Analysis

4.1 Experimental Preparation

(1) Datasets

The experiments are on the handwritten Chinese text data from CASIA-HWDB datasets, which represents the CASIA HWDB2.0-2.2 [42]. The dataset is completed by 1019 writers and each one write 5 manuscripts, which includes 5091 manuscripts (excluding 4 lost manuscripts) and 2703 character categories. It consists of training dataset and testing dataset, where training dataset includes 4076 manuscripts, 41781 text lines and 1081508 characters. Testing dataset includes 1015 manuscripts, 10449 text lines and 267906 characters.

As shown in Tab. 1, the average values of height and width of the text image from CASIA-HWDB are equal to 3488 and 2480 pixels respectively. However, we find that there are blank areas above and below the original text images, and this area is generally large. The average and median values of fine-height of these text images are 1363 and 1312 respectively, which are less than 40% of the original value of image height. Therefore, we continue to analyze and obtain the histogram of the fine-height of text image shown in Fig. 5, where the horizontal axis indicates the height area of the fine text image, and the vertical axis indicates the number of text images in a certain height area. The number of text images with a height range of [1100–1600] pixels reaches 2713 accounting for more than half of the dataset. More importantly, the numbers of text images in the range of (2100–2600] and (2600–3000] are only 160 and 20 respectively, both account for 3.54% of the whole dataset. Considering the above results, we split the text image with a fine-height of more than 2100 pixels into two sub-images, and then the width and height of the image are uniformly set to 2100 × 2400 pixels. Where '1' represents blank background and '0' represents character handwriting. There is use '1' to pad the blank part when the height or width of full page text image is less than the preset values.

Table 1: Basic information statistics of handwritten text image for CASIA-HWDB dataset

	Min	Max	Average	Median
Number of Lines	5	20	11	10
Number of characters	122	391	275	278
Original-Height (pixels)	3377	3593	3488	3484
Original-Width (pixels)	2371	2603	2480	2482
Fine-Height (pixels)	555	2968	1363	1312
Fine-Width (pixels)	1555	2432	2162	2174

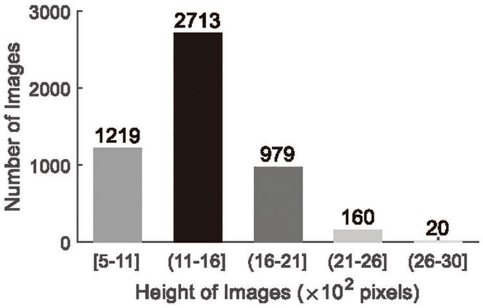


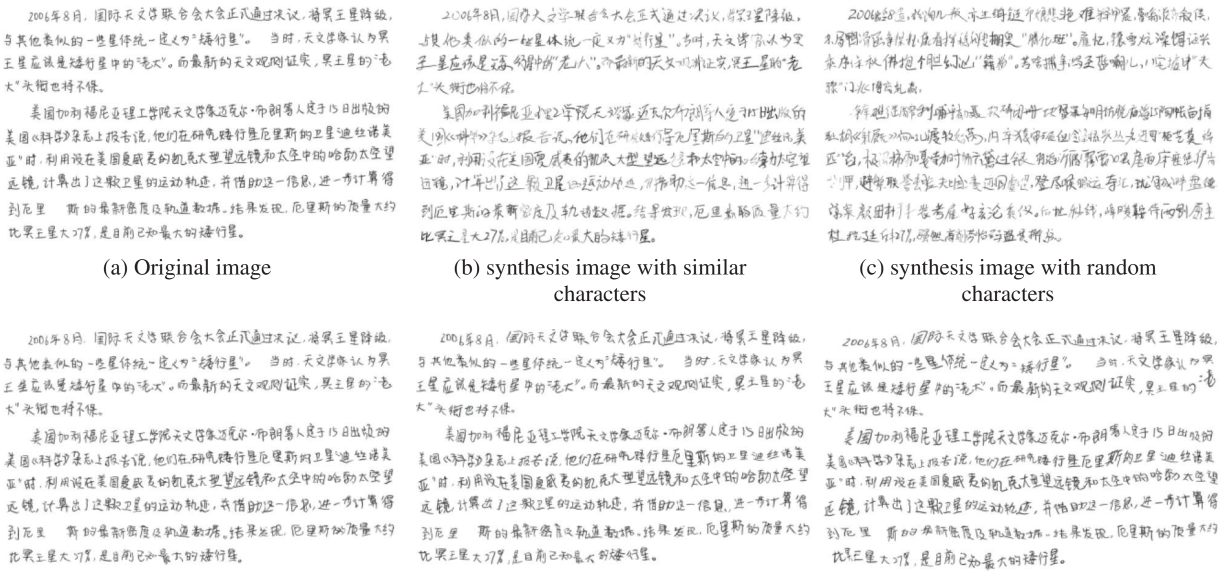
Figure 5: The histogram of image fine-height for CASIA-HWDB dataset

(2) Data Augmentation

While modern neural networks have shown good performance at handwritten recognition, the labeled training dataset is usually insufficient to cover the handwriting style widely from different writers. We here describe two main data augmentation techniques, data synthesis and grid-based distortion augmentation, which can benefit for most handwritten recognition. More than that, the data augmentation methods can be used independently to any handwritten text dataset in training.

(A) Data Synthesis

As shown in Figs. 6b–6c, we propose a synthetic pattern generation method which synthesis handwritten text images from text in CASIA-HWDB2.0-2.2 and isolated characters in CASIA-HWDB1.0-1.2, they are the two handwritten Chinese datasets of CASIA-HWDB [42]. Firstly, gain the character sequence of each line of text image in the former, and randomly replace each character of sequence with isolated characters from the latter. Secondly, put the new character sequence in the area of the original text line, where the height of all character does not exceed 90% of the text line height, the spacing between the characters is normally distributed of the difference value between the width of the text line minus the sum of the width of the characters. Finally, merge these text lines into a new synthesis handwriting text image. In the process of forming a text line by a set of characters, we introduce the 3-Sigma rule [43], which states that for a unified modal distribution nearly all conditions (about 99.73%) locate in three standard deviations from mean. This theory can be used to control the spacing between characters, or the offset between the center of the character and the vertical center of the text line. It is important to note that the mean value and standard deviation of the normal distribution should be controlled to allow a small amount of overlapping characters.



(a) Original image

(b) synthesis image with similar characters

(c) synthesis image with random characters

(d) grid 40*40 and std. 10 pixels

(e) grid 80*80 and std. 10 pixels

(f) grid 120*120 and std. 10 pixels

Figure 6: Augmented data samples. (a) Denotes the original image. (b–c) Represent the synthesis images. (d–f) Grid-based distortion augmentation with different size of grid interval and standard deviation (std.)

(B) Grid-based distortion augmentation

Grid-based distortion augmentation method uses random distortion on regular grids to augment existing handwritten text images [44]. The method allows the perturbation grid to utilize the warp over handwritten text, minimizing the creation of kinks or increase within a character, and then creating more natural twists. Meanwhile, this method can achieve a certain degree of augmentation of handwritten text image at different levels of character, text line and full page text, which difficult to realize simultaneously with other methods, such as project transform, affine transformation and elastic distortion, etc. The process of grid-based distortion augmentation is as follows. (1) Put key control points on a regular grid to align with baselines. Here, we set 40, 80 and 120 pixels interval (almost from one-third to one of the average baseline height for CASIA-HWDB). (2) Perturb each key control point in the horizontal and vertical direction by randomly sampling from a normal distribution. Here, we set the standard deviation of 5, 10 and 15 pixels, respectively. (3) Distort the image on the basis of the disturbed key control points. In other words, the augmentation method is based on three adjustment parameters, key control points, placement interval and standard deviation. Figs. 6d–1f are the augmented handwriting text images being distorted with different size of grid and standard deviation.

(3) Model evaluation

In order to emphasize the generality of the experiment, we compare the proposed method with some of stat-of-the-art methods that have achieved remarkable performance in handwriting recognition. Bluche et al. [37], ResNet-26 [38], OrigamiNet-12 [18] and Wu et al. [33], whose basic building details can be found in their respective research works. In each epoch, training samples are random sampled from the training dataset without any replacement. There are 90% samples of the training dataset from CASIA-HWDB and augmentation dataset for training the proposed model, the other is employed to verify the confidence of model parameter. Our model is implemented by TensorFlow [45] deep learning framework cooperating with Adam optimizer. The initial learning rate of 1×10^2 be used in all experiments for our experimental, which is exponentially decayed such that reaches 1×10^3 after 1×10^6 batches; the minimum batch allowed by our model training is not less than 2, and the 2×2 max pool is used successively for some networks layers. The detailed model parameters are shown in Fig. 4. In addition, the condition for stopping model training is that the loss function value does not decrease for 50 consecutive iterations or the max number of training times is 1×10^6 iterations.

Levenstein edit distance is often used to evaluate the performance of the handwritten recognition models at the character level, and normalize them by the total length of character sequence. In our experimental, referred on the existing works [7,9,15,29], the Accurate Rate (AR) and Correct Rate (CR) are used to estimate the proposed method and other comparative methods, whose formal can be express as: $AR = (N_t - D_e - I_e - S_e)/N_t$, and $CR = (N_t - D_e - S_e)/N_t$, where N_t indicates the total length of character sequence corresponding to the original text image. S_e, D_e and I_e indicate the length of the substitution errors, the deletion errors and the insertion errors, respectively.

4.2 Experimental Results

4.2.1 Comparison with Different Value of Text-line Up-sampling Length

The length of text-line up-sampling determines the mapping ability from the 2D text to 1D text-line which affects the efficiency of handwritten text recognition. To verify this problem, we designed the experiment of the accurate rate analysis with different lengths of text-line up-sampling. For CASIA-HWDB dataset, the final length of extraction features should be at least 392 due to the longest paragraph containing 391 characters. Therefore, we set five values for final up-sampling length l_3, which are [800,1200,1600,2000,2400]. Let Mode 1 and Mode 2 are the two relationships of the 3-layers text-line up-sample length. For Mode 1, we set the length of the former in two adjacent layers as twice that of the latter and round to the nearest integer multiples of one hundred greater than or equal to that element. For example, final text-line up-sampling $l_3 = 2400$, the lengths of the other two layers are $l_2 = \lceil l_3/2 \rceil = 1200$, $l_1 = \lceil l_2/2 \rceil = 600$ respectively. Similarly, we set the length of the former in two adjacent layers as two thirds times that of the latter in the Mode 2.

As shown in Tab. 2, we known that the optimal accuracy of the CASIA-HWDB dataset is 90.53%, and the corresponding three text-line up-sampling lengths are 2400, 1200 and 600 respectively. In both modes of the 3-layers text-line up-sample length relationship, the accuracy rate increases with the final up-sampling length l_3, which also shows that a long enough text-line up-sampling length is helpful to improve the accuracy of text recognitions. Fig. 7 shows the trends of accurate rate with two modes of final text-line up-sampling lengths. As the final text-line up-sampling length approaches 2400, the growth of accurate rate tends to be flat in the two modes which indicates that too small final text-line up-sampling length will adversely affect accurate rate of our method. Fig. 8 gives the trend of loss function value with final text-line up-sampling length of 2400 in AR of Mode 1. The loss function value of the first feedback is the largest, 1837, which shows a rapid decrease as the batches number increases due to the error rate of the model training at the initial stage is relatively high. As the error rate of the model training decreases, the decrease rate of the loss function value becomes smaller and tends to be flat.

Table 2: The accurate rate on CASIA-HWDB datasets with different up-sampling length (%)

Length relationship	Final text-line up-sampling length l_3				
	800	1200	1600	2000	2400
AR of Mode 1	80.66	84.05	87.59	89.72	90.53
AR of Mode 2	81.11	84.97	87.40	89.75	90.48

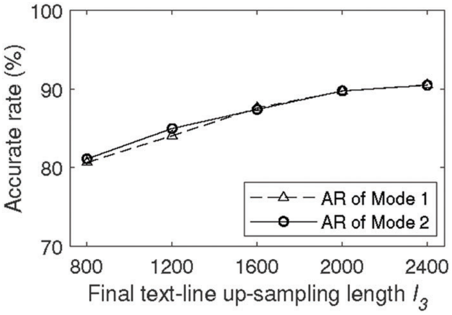


Figure 7: The trends of accurate rate with two modes of final text-line up-sampling length

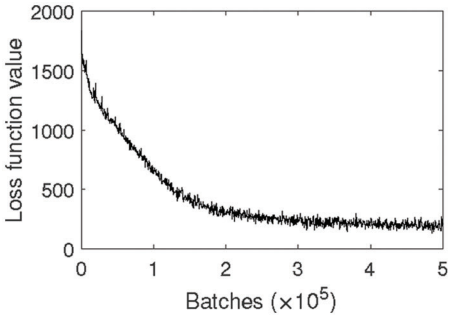


Figure 8: The trend of loss function value with final text-line up-sampling length of 2400 in Mode 1

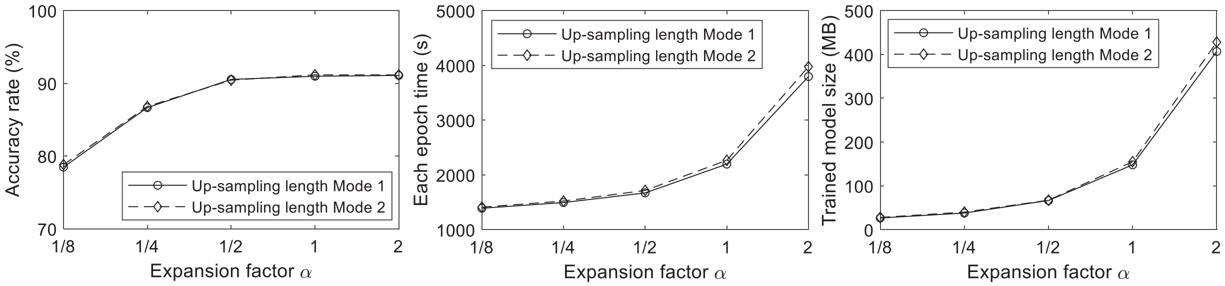
4.2.2 Comparison with Different Value of Expansion Factor

The expansion factor plays a vital role in the channels number in the GateBlock calculation, which up-sampling or down-sampling the input tensor into high-dimensional or low-dimensional representation, and then apply the lightweight depthwise convolution on it, finally down-sampling or up-sampling the representation into the size of the original tensor and achieve the output tensor. Tab. 3 shows the accurate rate, epoch time and model size of different expansion factor α on CASIA-HWDB. In which, the columns represent the different expansion factors α, and the rows represent the accurate rate, each epoch time and trained model size, respectively.

As can be seen from Tab. 3, accurate rate, epoch time and trained model size are increased with the exponential value of expansion factor. Among them, the growth of accurate rate is relatively gentle and tends to a certain value, the minimum and maximum values are 78.47% and 91.08% respectively, and the change range is 16.07%. However, the epoch time increased from 1390s to 3795s, and the trained model size increased 26.50 to 406.70 MB, their growth rate is much greater than the accurate rate. The trends of accurate rate, each epoch time and trained model size with different expansion factor on CASIA-HWDB dataset can be seen from Fig. 9, where the final text-line up-sampling length is 2400, and the solid line represents the results in up-sampling length Mode 1 and the dash line represents the results in up-sampling length Mode 2. Figs. 9a–9c demonstrate two results. First, although the increase of the expansion factor can improve the accuracy rate, it also increases the demand for computing resources required in the model training process and show a rapid increase trend. Second, the accuracy rate, each epoch time and trained model size of up-sampling length mode 2 are to a certain extent greater than that of up-sampling length mode 1, which shows that when the final up-sampling length is fixed, the up-sampling length mode with small interval is more likely to achieve high accuracy rate. To sum up, the exponential value of expansion factor $\alpha = 1/2$ can achieve a tradeoff of among the accurate rate, each epoch time and trained model size.

Table 3: The results with different expansion factors on CASIA-HWDB dataset

Index	Expansion factor α				
	1/8	1/4	1/2	1	2
Accurate rate(%)	78.47	86.65	90.53	90.96	91.08
Epoch time(s)	1390	1494	1670	2196	3795
Model size (MB)	26.50	37.80	66.50	147.90	406.70



(a) Accuracy rate (b) Each epoch time (c) Trained model size

Figure 9: The trends with different expansion factor on CASIA-HWDB dataset

4.2.3 Recognition Accuracy

To verify the proposed method, we evaluate it against the four state-of-art methods which achieve strong performance in the handwritten full page text recognition literature. In which, Bluche et al. [37] and Wu et al. [33] are based on MLSTM which can use context information of the handwritten text image and face the problems of time cost and space occupation caused by recursion operation. On the contrary, ResNet-26 [38], OrigamiNet-12 [18] and our method are based on the CNN focusing on the high-level feature extraction of handwritten text, and transformation of 2D text to 1D text line, so as to realize CTC classification. It has been proved that the CNN-based method lack the use of long-distance context information of character sequences in text recognition, but recursion-free structure can better use parallel computing to obtain higher efficiency.

Tab. 4 shows the text recognition accuracy results of different methods, where '—' indicates that the recognition accuracy of the corresponding method is missing. For CASIA-HWDB dataset, the accurate rates of ResNet-26 and OrigamiNet-12 are 79.25% and 81.72% respectively, and our method achieve the most accurate rate is 90.53%, and acquire an improvement for the first two with 14.23% and 10.78% respectively. For ICDAR-2013 dataset [46], Bluche et al. and Wu et al. proposed methods based on RNNs gain the accurate rates are 68.32% and 80.09% respectively. It can be seen that the accurate rate of latter is improved more than the former, and its improvement reached 17.23%. The accuracy of the other three methods based on CNN are 68.50%, 71.22% and 81.40% respectively. Our method gains the most accurate rate on the ICDAR-2013 dataset for all compared paragraph text recognition methods.

Table 4: The comparison of recognition accuracy (%)

Methods	CASIA-HWDB		ICDAR-2013	
	AR	CR	AR	CR
Bluche et al. [37]	–	–	68.32	–
ResNet-26 [38]	79.25	80.61	68.50	69.80
OrigamiNet-12 [18]	81.72	83.54	71.22	72.98
Wu et al. [33]	–	–	80.09	–
Ours	**90.53**	**92.60**	**81.40**	**82.85**

5 Conclusions

In this paper, we have proposed an end-to-end handwritten Chinese paragraph text recognition method based on residual attention convolutional neural networks and batch bilinear interpolation, which has the following features: segmentation-free, recurrent-free, representation feature enhancement and expansion factor adapt to different computing resource platforms. A novel residual attention gate block has been designed to reduce the importance of irrelevant features and increase the importance of meaningful features through weighting, and effectively alleviate the problems of gradient disappearance and gradient explosion for deeper convolutional neural networks. The batch bilinear interpolation serves as the key to realize paragraph text recognition without segmentation, which does not require any position information of characters/text-lines, and effectively solve the problem of high time costs, laborious and expensive handwritten text labeling works. Our experiments show that the proposed method exhibits superior performance on the CASIA-HWDB and ICDAR-2013 datasets.

In the future, we will focus on the lightweight design of the segmentation-free and recurrent-free network structure of handwritten text recognition, and introduce generative adversarial network to obtain more quality augmented data to achieve sufficient training of the recognition model.

Acknowledgement: Authors thankfully acknowledge all those who contributed this work scientifically, administratively, financially and academically during the whole research process.

Funding Statement: The authors extend their appreciation to the National Natural Science Foundation of China for funding this work through research Grant No. 61976118.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] Z. Wang, J. Du and J. Wang, “Writer-aware CNN for parsimonious HMM-based offline handwritten Chinese text recognition,” *Pattern Recognition*, vol. 100, no. 8, pp. 107102, 2020.

[2] Y. Zhou, J. Liu, Y. Xie and Y. K. Wang, “Morphological feature aware multi-cnn model for multilingual text recognition,” *Intelligent Automation & Soft Computing*, vol. 30, no. 2, pp. 715–733, 2021.

[3] S. S. Singh and S. Karayev, “Full page handwriting recognition via image to sequence extraction,” pp. 1–16, 2021. [Online]. Available: https://arxiv.org/abs/2103.06450.

[4] Y. Xue, Y. Tong, Z. Yuan, S. Su, A. Slowik *et al.,* “Handwritten character recognition based on improved convolutional neural network,” *Intelligent Automation & Soft Computing*, vol. 29, no. 2, pp. 497–509, 2021.

[5] P. Melnyk, Z. You and K. Li, “A high-performance CNN method for offline handwritten Chinese character recognition and visualization,” *Soft Computing*, vol. 24, no. 11, pp. 7977–7987, 2020.

[6] Y. Mohamed, H. Khaled and M. Usama, “Accurate, data-efficient, unconstrained text recognition with convolutional neural networks,” *Pattern Recognition*, vol. 108, no. 11, pp. 107482, 2020.

[7] C. Xie, S. Lai, Q. Liao and L. Jin, “High performance offline handwritten Chinese text recognition with a new data preprocessing and augmentation pipeline,” in *Int. Workshop on Document Analysis Systems*. Cham: Springer, pp. 45–59, 2020.

[8] Y. Wang, Y. Yang, W. Ding and S. Li, “A residual-attention offline handwritten Chinese text recognition based on fully convolutional neural networks,” *IEEE Access*, vol. 9, pp. 132301–132310, 2021.

[9] Y. Wu, F. Yin and C. Liu, “Improving handwritten Chinese text recognition using neural network language models and convolutional neural network shape models,” *Pattern Recognition*, vol. 65, no. 2, pp. 251–264, 2017.

[10] F. Baothman, S. Alssagaff and B. Ashmeel, “Decision support system tool for arabic text recognition,” *Intelligent Automation & Soft Computing*, vol. 27, no. 2, pp. 519–531, 2021.

[11] M. Badry, M. Hassanin, A. Chandio and N. Moustafa, “Quranic script optical text recognition using deep learning in iot systems,” *Computers Materials & Continua*, vol. 68, no. 2, pp. 1847–1858, 2021.

[12] R. Srivastava, K. Greff and J. Schmidhuber, “Training very deep networks,” pp. 1–11, 2015. [Online]. Available: https://arxiv.org/abs/1507.06228.

[13] V. Pham, T. Bluche, C. Kermorvant and J. Louradour, “Dropout improves recurrent neural networks for handwriting recognition,” in *Int. Conf. on Frontiers In Handwriting Recognition*, Crete Island, Greece, pp. 285–290, 2014.

[14] B. Théodore, “Joint line segmentation and transcription for end-to-end handwritten paragraph recognition,” in *Advances in Neural Information Processing Systems*, Barcelona, Spain, pp. 838–846, 2016.

[15] Y. Wu, F. Yin, Z. Chen and C. Liu, “Handwritten Chinese text recognition using separable multi-dimensional recurrent neural network,” in *Int. Conf. on Document Analysis and Recognition*, Kyoto, Japan, pp. 79–84, 2017.

[16] Z. R. Wang and J. Du, “Joint architecture and knowledge distillation in CNN for Chinese text recognition,” *Pattern Recognition*, vol. 111, no. 4, pp. 107722, 2021.

[17] B. Liu, X. Xu and Y. Zhang, “Offline handwritten Chinese text recognition with convolutional neural networks,” pp. 1–6, 2020. [Online]. Available: https://arxiv.org/abs/2006.15619.

[18] Y. Mohamed and B. Tom, “OrigamiNet: Weakly-supervised, segmentation-free, one-step, full page text recognition by learning to unfold,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, Seattle, WA, USA, pp. 14710–14719, 2020.

[19] Z. Wang, Y. Yu, Y. Wang, H. Long and F. Wang, “Robust end-to-end offline chinese handwriting text page spotter with text kernel,” pp. 1–15, 2021. [Online]. Available: https://arxiv.org/abs/2107.01547.

[20] A. Graves, S. Fernández, F. Gomez and J. Schmidhuber, “Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks,” in *Int. Conf. on Machine Learning*, Orlando, FL, pp. 369–376, 2006.

[21] W. Xiaohua, L. Shujing and L. Yue, “Compact MQDF classifiers using sparse coding for handwritten Chinese character recognition,” *Pattern Recognition*, vol. 76, no. 1, pp. 679–690, 2018.

[22] F. Kimura, K. Takashina, S. Tsuruoka and Y. Miyake, “Modified quadratic discriminant functions and the application to Chinese character recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 1, pp. 149–153, 1987.

[23] Z. Li, N. Teng, M. Jin and H. Lu, “Building efficient CNN architecture for offline handwritten Chinese character recognition,” *International Journal on Document Analysis and Recognition*, vol. 21, no. 4, pp. 233–240, 2018.

[24] X. Xiao, L. Jin, Y. Yang, W. Yang, J. Sun *et al.,* "Building fast and compact convolutional neural networks for offline handwritten Chinese character recognition," *Pattern Recognition*, vol. 72, no. 1, pp. 72–81, 2017.

[25] Z. Li, Q. Wu, Y. Xiao, M. Jin and H. Lu, "Deep matching network for handwritten Chinese character recognition," *Pattern Recognition*, vol. 107, no. 1, pp. 107471, 2020.

[26] J. Li, G. Song and M. Zhang, "Occluded offline handwritten Chinese character recognition using deep convolutional generative adversarial network and improved GoogLeNet," *Neural Computing and Applications*, vol. 32, no. 9, pp. 4805–4819, 2020.

[27] Z. Wang, J. Du, W. Wang, J. Zhai and J. Hu, "A comprehensive study of hybrid neural network hidden Markov model for offline handwritten Chinese text recognition," *International Journal on Document Analysis and Recognition*, vol. 21, no. 4, pp. 241–251, 2018.

[28] S. Wang, L. Chen, L. Xu, W. Fan, J. Sun *et al.,* "Deep knowledge training and heterogeneous CNN for handwritten Chinese text recognition," in *Int. Conf. on Frontiers in Handwriting Recognition*, Shenzhen, China, pp. 84–89, 2016.

[29] Q. Wang, F. Yin and C. Liu, "Handwritten chinese text recognition by integrating multiple contexts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 8, pp. 1469–1481, 2012.

[30] R. Messina and J. Louradour, "Segmentation-free handwritten Chinese text recognition with LSTM-RNN," in *Int. Conf. on Document Analysis and Recognition*, Tunis, Tunisia, pp. 171–175, 2015.

[31] D. Peng, L. Jin, Y. Wu, Z. Wang and M. Cai, "A fast and accurate fully convolutional network for end-to-end handwritten Chinese text segmentation and recognition," in *Int. Conf. on Document Analysis and Recognition*, Sydney, Australia, pp. 25–30, 2019.

[32] B. Moysset, C. Kermorvant and C. Wolf, "Learning to detect, localize and recognize many text objects in document images from few examples," *International Journal on Document Analysis and Recognition*, vol. 21, no. 3, pp. 161–175, 2018.

[33] Y. Wu and X. Hu, "From textline to paragraph: A promising practice for Chinese text recognition," in *Proc. of the Future Technologies Conf.*, San Francisco, CA, USA, pp. 618–633, 2020.

[34] B. Moysset, T. Bluche, M. Knibbe, M. Benzeghiba, R. Messina *et al.,* "The A2IA multi-lingual text recognition system at the second Maurdor evaluation," in *Int. Conf. on Frontiers in Handwriting Recognition*, Crete Island, Greece, pp. 297–302, 2014.

[35] C. Wigington, C. Tensmeyer, B. Davis, W. Barrett, B. Price *et al.,* "Start, follow, read: End-to-end full-page handwriting recognition," in *European Conf. on Computer Vision*, Cham, Springer, pp. 367–383, 2017.

[36] C. Tensmeyer and C. Wigington, "Training full-page handwritten text recognition models without annotated line breaks," in *Int. Conf. on Document Analysis and Recognition*, Sydney, Australia, pp. 1–8, 2019.

[37] T. Bluche, J. Louradour and R. Messina, "Scan, attend and read: End-to-end handwritten paragraph recognition with mdlstm attention," in *Int. Conf. on Document Analysis and Recognition*, Kyoto, Japan, pp. 1050–1055, 2017.

[38] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in *IEEE Conf. on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, pp. 770–778, 2016.

[39] C. Dong, C. C. Loy, K. He and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 295–307, 2016.

[40] M. R. Khosravi and S. Samadi, "BL-ALM: A blind scalable edge-guided reconstruction filter for smart environmental monitoring through green IoMT-UAV networks," *IEEE Transactions on Green Communications and Networking*, vol. 5, no. 2, pp. 727–736, 2021.

[41] E. J. Kirkland, "Bilinear interpolation," in *Advanced Computing in Electron Microscopy.* Springer, Boston, MA, USA, pp. 261–263, 2010.

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

[43] F. Pukelsheim, "The three sigma rule," *The American Statistician*, vol. 48, no. 2, pp. 88–91, 1994.

[44] C. Wigington, S. Stewart, B. Davis, B. Barrett, B. Price *et al.,* "Data augmentation for recognition of handwritten words and lines using a cnn-lstm network," in *Int. Conf. on Document Analysis and Recognition*, Kyoto, Japan, pp. 639–645, 2017.

[45] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis *et al.,* "Tensorflow: A system for large-scale machine learning," in *USENIX Symp. on Operating Systems Design and Implementation*, Savannah, GA, USA, pp. 265–283, 2016.

[46] F. Yin, Q. Wang, X. Zhang and C. Liu, "ICDAR 2013 Chinese handwriting recognition competition," in *Int. Conf. on Document Analysis and Recognition*, Beijing, China, pp. 1464–1470, 2013.