Frame Detection and Text Line Segmentation for Early Japanese Books Understanding

Lyu Bing[1] [a], Hiroyuki Tomiyama[2] [b] and Lin Meng[2] [c]

[1]*Graduate School of Science and Engineering, Ritsumeikan University, 1-1-1 Noji-higashi, Kusatsu, Shiga 525-8577, Japan*

[2]*College of Science and Engineering, Ritsumeikan University, 1-1-1 Noji-higashi, Kusatsu, Shiga 525-8577, Japan*

Keywords: Text Line Segmentation, Early Japanese Books Understanding, Deep Learning, Image Processing.

Abstract: Early Japanese books record a lot of information, and deciphering these pieces of ancient literature is very useful for researching history, politics, and culture. However, there are many early Japanese books that have not been deciphered. In recent years, with the rapid development of artificial intelligence technology, researchers are aiming to recognize characters in the early Japanese books through deep learning in order to decipher the information recorded in the books. However, these ancient literature are written in Kuzushi characters which is difficult to be recognized automatically for the reason for a large number of variation and joined-up style. Furthermore, the frame of article and the text line tilt increase the difficult recognition. This paper introduces a deep learning method for recognizing the characters, and proposal frame deletion and text line segmentation for helping Early Japanese Books understanding.

1 INTRODUCTION

Since early Japanese books have recorded a lot of information as cultural heritage, organizing early Japanese books are useful for research on politics, history, culture, and so on. There are many unregistered pieces of ancient literature in Japan, and there is an urgent need to reorganize the early Japanese books for further researching and understanding of the Japanese culture. However, many early Japanese books are written in Kuzushi characters which is a typeface that can be simplified and written quickly, and is also called Cursive. Currently, most of the characters are not used, and only a few experts can decipher them, so deciphering early Japanese Books is very time consuming and difficult. In recent years, artificial intelligence technology has progressed (A. Krizhevsky, 2012; C. Szegedy, 2015; Simonyan and Zisserman, 2015; K. He, 2016), and researchers are aiming at automatic recognition of Kuzushi characters and other ancient literature using deep learning (L. Meng, 2018). However, in these studies, since each character must be manually cut out before being recognized, automatic recognition of all sentences has not been realized yet. In addition, there are many cases in which text and images are mixed in early Japanese books, and it is necessary to separate the text and images in advance to automatically decode the text. However, images are drawn by hand, just like letters, increasing the difficulty of automatically decoding all sentences. And the difficulty of automatic deciphering of early Japanese books is increasing due to the joined-up style which lets the characters are connected characters, the characters are very blurred, and dirt, worms,

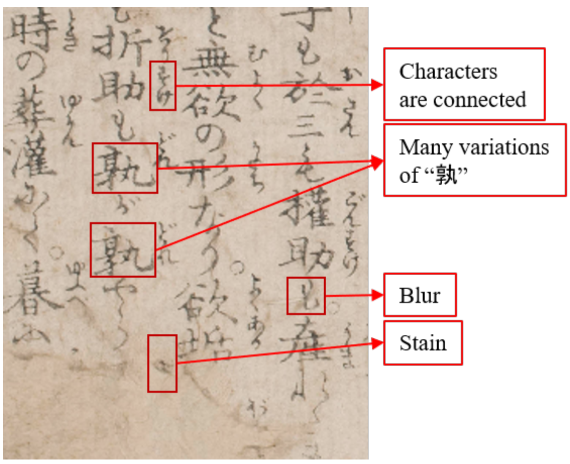


Figure 1: Kuzushi character and a scanned page of the early Japanese book.

[a] https://orcid.org/0000-0000-0000-0000
[b] https://orcid.org/0000-0000-0000-0000
[c] https://orcid.org/0000-0003-4351-6923

Bing, L., Tomiyama, H. and Meng, L.
Frame Detection and Text Line Segmentation for Early Japanese Books Understanding.
DOI: 10.5220/0009179306000606
In *Proceedings of the 9th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2020)*, pages 600-606
ISBN: 978-989-758-397-1; ISSN: 2184-4313

etc.. Furthermore, the frame of article and the text line tilt increase the difficult recognition. Figure 1 shows an example of the early Japanese books register and its problems.

This paper aims to use image processing and deep learning combined method to automatically understanding early Japanese books. At first, the frame deletion method and text line segmentation method are proposed to extract the text line. About the frame deletion, conventional image processing methods are used, and ARU-Net, a deep learning method is applied for line segmentation. Then, the character size is estimated for the segmented sentence line and cut out the character candidates. Finally, AlexNet is applied for character recognition which selected the high reliability from the candidate characters.

Section 2 describes the difference between Kuzushi character documents and current characters documents and introduce some modern ancient documents recognition method including image processing and deep learning. Section 3 goes over the research flow of our proposal. Section 4 shows the frame detection by ARU-Net. Section 5 shows the experimentation. Section 6 concludes the paper with a brief summary and mention of future work.

2 RELATED WORK

2.1 The Difference between Kuzushi Character Documents and Current Character Documents

The kuzushi character in early Japanese books is very different from the current characters.

Figure 2 (a) shows an image of kuzushi character, and Fig. 2 (e) shows a handwritten English image. It can be seen that compared with the alphabet, kuzushi is more complicated and illegible. For the reason of that the alphabet is only 26, and simple to write, Hence the alphabet can be easily recognized.

Figure 2 (b) is a printed version of Japanese characters, as you can see Japanese characters include kanji, hiragana, katakana and so on, which makes it difficult to identify the characters in early Japanese books. Figure 2 (c)(d) show hiragana and katakana in Japanese characters. We found the character is not connected with neighboring characters. It is a major between current documents and early Japanese books which were written by Kuzushi characters. Also, the characters in Figure 2 (b)(c)(d) are uniformed, otherwise, Kuzushi characters are un-uniformed and have larger number of variation. These two major problems let the Kuzushi characters recognition is more difficult than kanji, hiragana, katakana.

About Chinese characters (Kan ji) recognition as shown in Fig. 2 (f), these character like Kuzhishi characters which have several variation and ununiformed. However, these character are not connected with neighboring characters. Hence, the Kuzushi characters recognition is more difficult than the Chinese characters recognition.

(a) Example of kuzushi character (b) Example of Japanese print

(c) Example of hiragana (d) Example of katakana

(e) Example of alphabet (f) Example of Chinese character

Figure 2: Character examples.

2.2 Deep Learning and Image Processing based Characters Recognition

Researchers have proposed several method for the characters recognition. The software of OCR (Optical Character Recognition) has been developed which is a very popular research for recognizing the characters from documents in 1980's. The documents are scanned, and the layout of the documents are analyzed which indlucdes the characters segmentation and character segmentation. Then the segmented characters are normalized and searched from the template by image processing method (C.C. Tappert, 1990).

In the detail of recognition method for ancient characters, There are several methods for recognizing OBIs by using template matching and by using the Hough transform (L. Meng, 2016; L. Meng and Oyanagi, 2016; Meng, 2017). However, the template matching in (L. Meng and Oyanagi, 2016) was weak when the original character was tilted, and the tilt was also not properly processed. About (L. Meng and Oyanagi, 2016; Meng, 2017), the Hough transform and clustering combined method is proposed for extracting the extracting of charactes. However, this method is only fit for the feature is clear and the kinds of feature are not so many. However, the Kuzushi characters have a larger number of variation and the feature of characters are not easy to be defined. Furthermore, we found the deep learning method achieved a beter performance than the current image processing method.

Currently, larger number of deep learning models are proposed. LeNet (L. Yann and Haffner, 1998), AlexNet (A. Krizhevsky, 2012), GoogLeNet (C. Szegedy, 2015), VGG (Simonyan and Zisserman, 2015) etc. are the model which only have the function of recogntion. It means the character should be prepared in the pre-processing stage. Sometimes, the data should be cut man-made.

SSD (W. Liu and Berg, 2016), YoLo (Redmon and Farhadi, 2018), CenterNet (Zhou et al., 2019) are model which have the function of character detection and character recognition. For the reason the these model are designed by detecting and recognizing the bigger object. Hence in the case of the smaller characters detection and recognition, the performance can not be realized directly.

In this paper we try to use image processing and deep learning combined method for the kuzushi characters recognition. The deep learning method is applied for characters detection and deep learning method of AlexNet is applied for the characters recogntion.

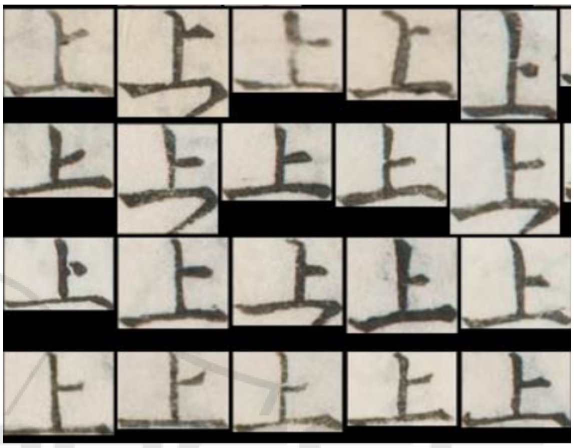
3 OVERVIEW OF CHARACTER RECOGNITION

Figure 4 shows the overview of character recognition for Early Japanese Book understanding.

First of all, the image preprocessing, including the image frame detection and deletion of the book images. Then, ARU-Net (T. Grning and Labahn, 2018)is applied to detect the text area, then the detection results are used to segment the text line in the image. Finally, AlexNet (A. Krizhevsky, 2012) was used for character recognition.

U+306E の 2016	U+3044 い 1514	U+304B か 1503	U+306A な 1336	U+306F は 1330
U+3089 ら 1160	U+307E ま 1079	U+306B に 1073	U+300C 「 1072	U+3068 と 1043
U+3064 つ 952	U+3078 へ 945	U+308B る 912	U+308A り 884	U+3060 だ 869

(a) Type and number of characters



(b) Example of training image

Figure 3: Example of training data.

In term of text line segmentation, the method has been proposed by us in (Bing Lyu and Meng, 2019) Figure 5 shows the segmentation flow and text line segmentation result by using ARU-Net. Figure 5 (a) shows a frame noise removed image, the text line was detected by ARU-Net and the result is shown in Fig. 5 (b). In the same time, Fig. 5 (a) is rotated by 180 degrees and the text line extracted image is obtained by ARU-Net, which is shown in Fig. 5 (c). Then, the text line extracted images of Fig. 5 (b) (c) are binarized, and binarization image of Fig. 5 (e) (f) is obtained

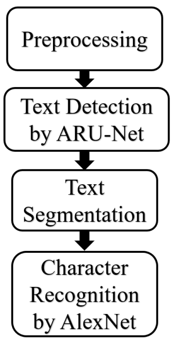


Figure 4: Overview of Character Recognition System.

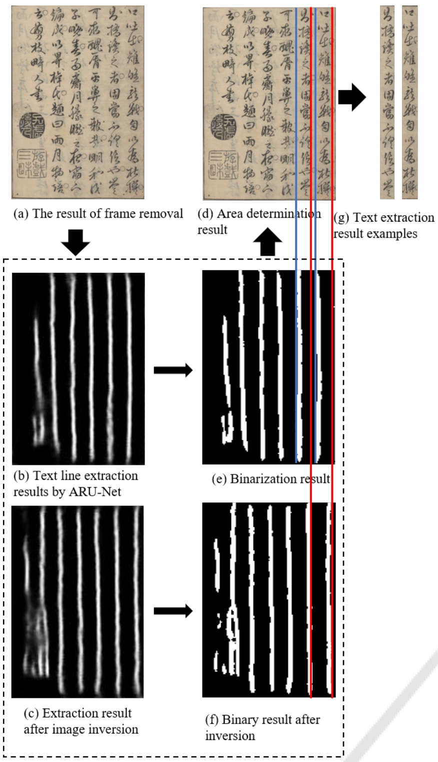


Figure 5: Detailed text segmentation flow.

respectively. Finally, the sentence line between them is cut out using the leftmost coordinate (blue line) for each white line in Fig. 5 (e) and the rightmost coordinate (red line) for each white line in Fig. 5 (f). As shown in Fig. 5 (g), the sentence line is obtained correctly.

In terms of character recognition, AlexNet is applied and Fig. 6 shows an example of character recognition by LeNet using the segmented text lines. Since the character is assumed to be a square, the size of the vertical and horizontal axes of the character is the distance between the red line and blue line of the character area shown in Fig. 5. The squares in Fig. 6 indicate the segmented characters. Then, from the beginning of the segmented character string (yellow square), 1/5 of the character string width is segmented as the stride length. Next, LeNet recognizes the segmented characters. Here, the sentence line is sliced, and the upper two letters become three-letter candidates. The confidence threshold is defined as 90%. Characters with the reliability of 90% or higher are recognized. In Fig. 6, the first candidates recognized by LeNet are 90.2%, 50.4%, 46.2%, 90.5%, and 100%, respectively, so high confidence characters of 90.2%, 90.5%, and 100% Recognition result.

However, by analyzing the experimental results, we found the proposed method can not segment the text line correctly in some cases. As known, the accuracy of character recognition depends on the correct segmented text line, the AlexNet cannot predict the characters in the mis-segmented text line.

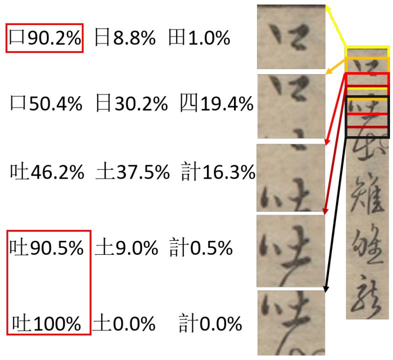


Figure 6: Character recognition example.

4 FRAME DELETION

Peak calculation method has been proposed for proposed and achieves a good performance in the case of that documents only have one frame (Bing Lyu and Meng, 2019). However, some pages have several frames or the frame numbers are difficult to be judged. The example is shown in Fig. 8 (a) which has two frames and can not achieve better performance by method (Bing Lyu and Meng, 2019).

4.1 Frame Deletion by ARU-Net

Here, we propose several methods for overcoming the problem of peak calculation method and detecting frames. ARU-Net is a two-step deep learning model for detecting sentence lines of old books. As shown in Fig. 7, the result processed by A-Net and softmax is merged with the result processed by RU-Net. Similarly, the resized two original images were also processed in the same way, but each processing was deconvolved, and finally, the three fused results were used for classification.

Detecting frame by image processing is a useful method for some images (Bing Lyu and Meng, 2019), however, some failure cases still exist. As shown in Fig. 8 (b) which has two frames in the scanned literature. The result of frame deletion using image processing is shown in Fig. 8 (a). Only the blank part outside the first frame can be deleted and a large part of the frame still exists which causing that the text line can not be segmented correctly.

Therefore, we use ARU-Net to carry out text detection in Fig. 8 (b), and the result was shown in Fig. 8 (c). Then the detected results were binarized, as shown in Fig. 8 (d). The processing results are calcu-

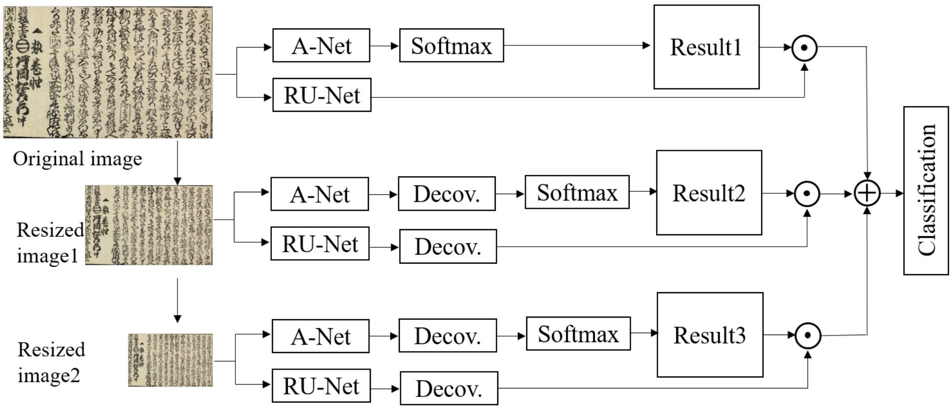


Figure 7: Overview of ARU-Net.

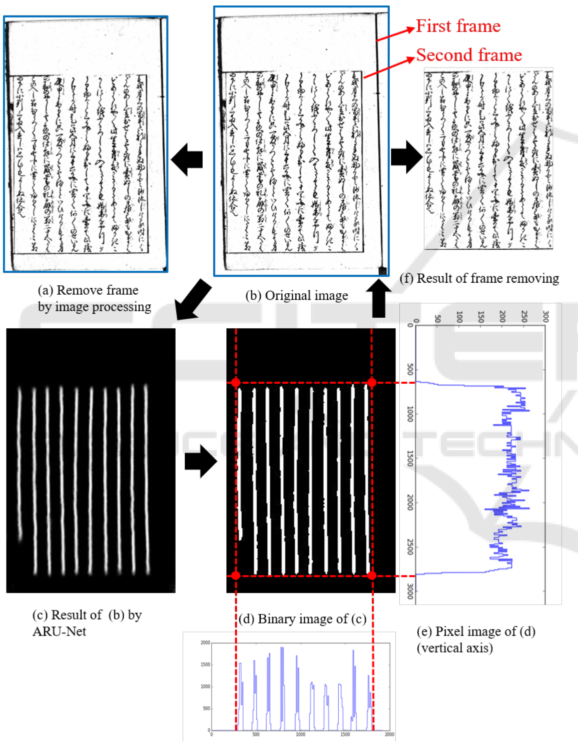


Figure 8: Remove the Frame using ARU-Net.

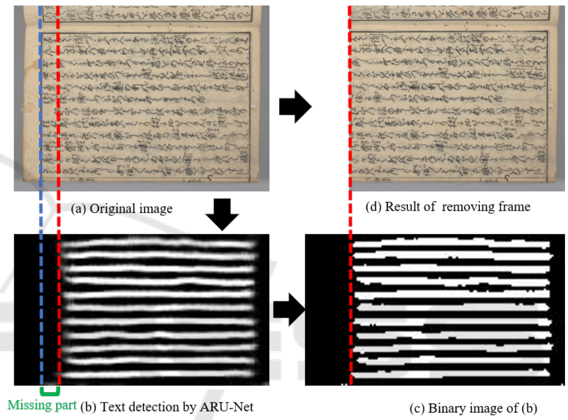


Figure 9: Problems with using ARU-Net to remove frame.

lated in pixels on the horizontal and vertical axes, as shown in Fig. 8 (e). Find the four coordinates of the place where the pixel starts to rise in the pixel graph. The four red dots in Fig. 8 (d) are the four coordinates of the image frame. Finally, according to the four coordinates, the part beyond the frame of the image was cut off, and the result as shown in Fig. 8 (f).

However, due to the low accuracy of ARU-Net's judgment on the beginning and end of sentences, the situation in Fig. 9 also occurred. As shown in Fig. 9 (b), part of the text is missing due to the inaccuracy of ARU-Net detection text.

4.2 Two Stages Frame Deletion

By analyzing the experimental results, we found that the peak calculation method and the ARU-Net method still has some disadvantages. Here, we propose a method that combines the two methods and uses two stages for overcoming the problems. Specifically, in the first image processing, a part of the image frame is deleted,and then we use ARU-Net to delete the frame again.

The effect is shown in Fig. 10. The original image is first removed by image processing, and the result is shown in Fig. 9 (b). However, there is still some problem like an extra part. Therefore, ARU-Net is used in text detection in Fig. 9 (b) to obtain four coordinate. Finally, these four coordinates were used to remove the frame again in Fig. 9 (c), and the final result is shown in Fig. 9 (d). As can be seen, the effect of removing the frame is very good, which proved the effectiveness of our method.

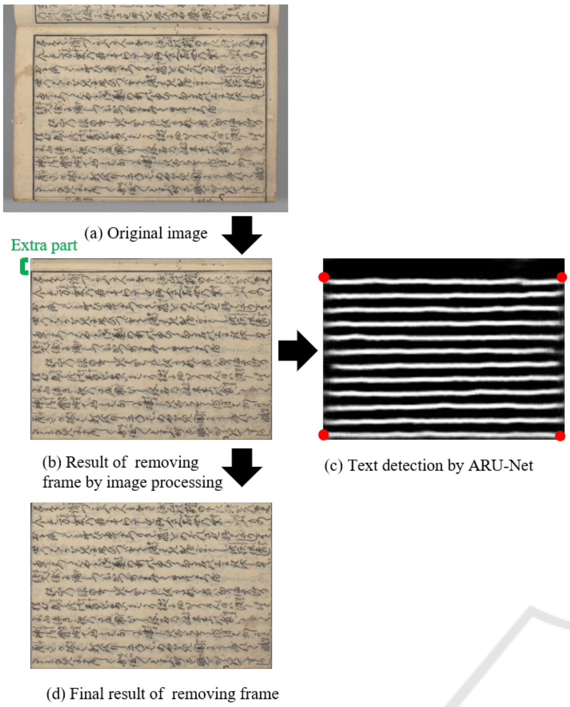


Figure 10: Remove frame by two stage.

5 EXPERIMENTATION

In order to verify the validity of our method for different early Japanese books, we conducted experiments.

5.1 Experimental Conditions

We select 10 pages from early Japanese books collected in Center for Open Data in the Humanities (CODH), for measuring the proportion of the cut part in the whole image.

The OS is ubuntu16.04 LTS and the programming language is Python.

5.2 Experimental Result of Frame Deletion

The experimental results are shown in Fig. 11. We compare different methods to remove the blank part outside the frame. The yellow line is the ratio of the area removed only through image processing to the area of the original image, the red line is the ratio of the area removed only through ARU-Net to the area of the original image, and the blue line is the ratio of the area removed only through our proposed method to the area of the original image.

It is obvious from Fig. 11 that the area removed

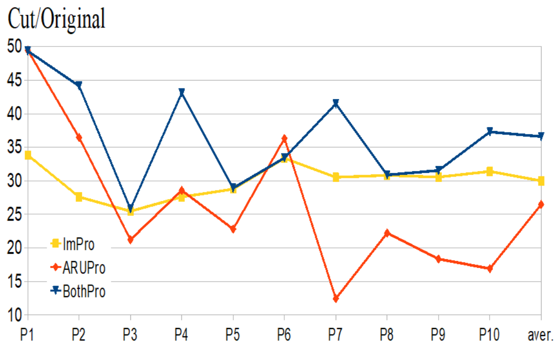


Figure 11: Experimental Result.

by our proposed method is larger than that removed by image processing or ARU-Net alone. On average, 36.6% of the blank parts of the original image were removed by our proposed method, while 30.0% and 26.5% were removed by image processing and ARU-Net, respectively. The amount of blank parts removed increased by 6.6% and 10.1%.

5.3 Experimental Result of Characters Recognition

After text line segmentation, we cut the character for character recognition. The LeNet is applied for character recognition. Currently, we only use a slight training dataset and testing dataset for measuring the performance of character recognition.

Table 1 shows the detail of the dataset and the character recognition accuracy. The results show that the recognition accuracy achieved at about 90%, and proved the effectiveness of our proposal.

5.4 Discussion

Although it can be seen from the experiment that our method produces good results for most images, there are some problems. As shown in Fig. 11, it is obviously better for P4 to process only using ARU-Net. For P3 and P8, the results obtained by our proposed method and image processing are roughly the same. Another problem is that just increasing the removal area is easy to cut out some of the text in the frame when we cut out the blank part, as shown in Fig. 12. And should not only calculate the removal area, but also verify the correctness of the calculation of the removal area.

6 CONCLUSIONS

In this paper, we introduce several methods to delete the frame of scanned literature of early Japanese

Table 1: Results of character recognition accuracy.

class	training image numbers	testing image numbers	epoch	time	accuracy
10	19323	1000	200	0:10:35	0.930
20	28412	2000	300	0:32:28	0.884
30	35165	3000	400	1:06:44	0.899
40	41425	4000	500	1:49:01	0.890

Figure 12: Problems of processed image.

books to help automatic understanding. the proposal includes ARU-Net based method and the two-step method. In the experiment, the frame deletion of our proposed method is greatly improved, which verifies the correctness of our method. But the correctness of the cut out blank part has not been verified, which will be our future work. Moreover, our purpose is to realize the automatic recognition of early Japanese books, so the automatic extraction of text lines and automatic character recognition are also our future work.

ACKNOWLEDGEMENTS

This research is supported by the Art Research Center of Ritsumeikan University. In addition, We would like to thank Prof. Akama Ryo Prof.Takaaki Kaneko for his advice.

REFERENCES

A. Krizhevsky, I. Sutskever, G. H. (2012). Imagenetclassification with deep convolutional neural networks. Advances in Neural Information Processing System Systems 25.

Bing Lyu, Ryo Akama, H. T. and Meng, L. (2019). The early japanese books text line segmentation base on image processing and deep learning. In *The 2019 International Conference on Advanced Mechatronic Systems (ICAMechS 2019)*.

C. Szegedy, W. Liu, Y. J. P. S. S. R. D. A. D. E. V. V. A. R. (2015). Goingdeeper with convolutions. 2015 IEEE Conference on Computer Vision and Pattern Recognition.

C.C. Tappert, C.Y. Suen, T. W. (1990). U-net: Convolutional networks for biomedical image segmentation. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

K. He, X. Zhang, S. R. J. S. (2016). Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition.

L. Meng, C.V. Aravinda, K. R. U. K. R. e. a. (2018). Ancient asian character recognition for literature preservation and understanding. In *Euromed 2018 International Conference on Digital Heritage*. Springer Nature.

L. Meng, T. I. and Oyanagi, S. (2016). Recognition of oracular bone inscriptions by clustering and matching on the hough space. *J. of the Institute of Image Electronics Engineers of Japan*, 44(4):627–636.

L. Meng, Y. F. e. a. (2016). Recognition of oracular bone inscriptions using template matching. *Int. J. of Computers Theory and Engineering*, 8(1):53–57.

L. Yann, L. Bottou, Y. B. and Haffner, P. (1998). Gradient-based learning applied to document recognition. In *Proceeding of the IEEE*.

Meng, L. (2017). Recognition of oracle bone inscriptions by extracting line features on image processing. In Pro. of the 6th Int. Conf. on Pattern Recognition Applications and Methods (ICPRAM2017).

Redmon and Farhadi, A. (2018). Yolov3: An incremental improvement. In *Computer Vision and Pattern Recognition (ECCV 2018)*.

Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *Advances in Neural Information Processing Systems 28 (NIPS 2015)*.

T. Grning, G. Leifert, T. S. and Labahn, R. (2018). A two-stage method for text line detection in historical documents. In *Computer Vision and Pattern Recognition*.

W. Liu, D. Anguelov, D. C. S. S. R. C. F. and Berg, A. C. (2016). Ssd: Single shot multibox detector. In *Computer Vision and Pattern Recognition (ECCV 2016)*.

Zhou, X., Wang, D., and Krähenbühl, P. (2019). Objects as points. In *arXiv preprint arXiv:1904.07850*.