KUZUSHIJI RECOGNITION

AKSHAR VIBHAV[1], ASHWINI R[2], BHARGAVI R KAMAT[3], BINDU R N[4]
Department of Computer Science, The National Institute of Engineering
Manandavadi Road, Mysore, India

ABSTRACT Recognition of historical documents is a challenging problem due to the noised, damaged characters, and background. Kuzushiji is a cursive script used in Japan for a long time. Therefore, most of the historical documents are written in Kuzushiji, which only a very few can read. In Kuzushiji documents, Hiragana and Katakana are used. Since Kuzushiji is a cursive script, character segmentation-based methods do not work well. This leads to the idea of creating a new method to recognize the Kuzushiji scripts. In this paper, we propose a machine learning model which uses the concepts of CenterNet and Heatmaps to detect the characters and CNN model to classify the letters and identify their corresponding Japanese character. This system is successful in recognizing multiple lines, connected and cursive characters without performing character or line segmentation. It is capable of handling long range context, large vocabularies, and non-standardized character layouts. We tested our model on the test cases given by the Kaggle website. The result of the experiments provides an accuracy of 15% due to the hardware limitations we had while training the data.

1. INTRODUCTION

Through the development of human civilizations, writing systems have been changed over time in every language. Humanity's rich history has left behind a large number of historical documents, containing stories and other experiences which are essential to our cultural heritage. These can be understood by only few trained experts which is very slow and time consuming. Archaeologists have unearthed a lot of clay tablets, yet only a few trained experts can translate them. As a result, most of them have never been read. This is a global problem. One such example is the case of Japan. From 800 until 1900CE, Kuzushiji, a cursive script was in use. It was removed from the academic curriculum in 1900. As a result, the majority of the Japanese speakers cannot read these documents which are 150 years old. The volume of these texts, comprising over millions of books with millions of pages in each book is readable by only few scholars.

This motivated the use of machine learning to understand these texts. But, this is a difficult task. Kuzushiji is written in a script which is different from the modern Japanese, which makes even the basic recognition. There are several other reasons why Kuzushiji recognition is challenging. It is important to capture both the local and global context. It is seen that some characters are written in a contextually dependent way. This makes it important to consider multiple characters while classifying instead of taking each character into consideration. The total number of characters is very large. Many characters appear few times. Kuzushiji is a mixture of Hentaigana and Hirangana. Many characters can be written in multiple ways, based on Hentaigana, which makes it even more difficult to understand. Kuzushiji texts are often written together with backgrounds and pictures which makes it difficult to separate from text.

FIGURE 1. Example of Kuzushiji texts written on an artistic background

Many approaches and models have been developed to recognize Kuzushiji scripts.

The traditional systems follow two steps: text line detection and text line recognition. In text line detection, documents are segmented into text lines and then, each line is recognized using a text line recognizer. The traditional system works well on the printed scripts and handwritten documents. However, their performance is insufficient for historical documents like Kuzushiji documents. They find it difficult to deal with the cursive and connected characters. These models follow a procedure where text lines and characters are segmented before they are recognized. But segmentation reduces the performance of the model.

To overcome the above problems, we propose a model which first detects the characters. To detect the characters, we use the concept of CenterNet, heatmaps and bounding boxes. Once the characters are detected, we then use a CNN model to classify and recognize the characters. This model makes sure that all characters in document or an image is detected and recognized.

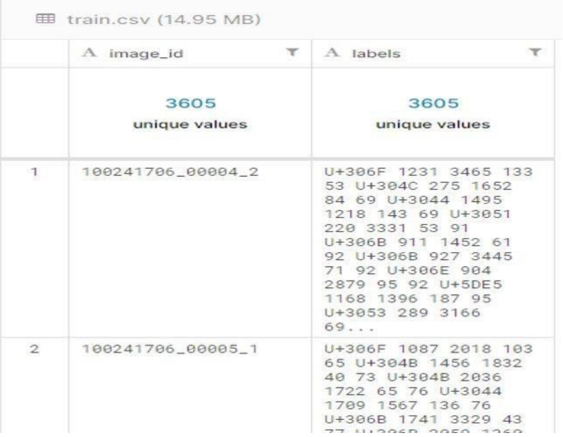
2. RELATED WORKS

[1] Anh Duc Le, Tarin Clanuwat, Asanobu Kitamoto proposed a human inspired recognition system to recognize the Kuzushiji documents. It was based on the human reading behavior which determined the first character of a text line. Then scan the next character and continue till it reaches the end of a text line and then moved to the next text line. [2] Chulapong Panichkriangkrai proposed an Internet based interactive transcription system. This model Does segmentation, modify the character segmented and the make transcription. [3] Yuta Hashimoto, Yoichi Iikura developed a mobile learning application for reading as an aid to learn Kuzushiji characters and documents. The application has three modules. They are character reading, text reading, and community modules. The character module is employed to find out character shapes of 102 hentaigana and 176 kanji characters. The reading module helps users to find out the way to read Kuzushiji characters from real classical texts. The classical texts are accompanied by transcriptions. The community module lets users communicate with each other to exchange learning materials and learning experiences. [4] K.Ikeda, R. Hayashi, K. Nagasaki, A. Morishima proposed a human assisted OCR for transcribing books and other old library projects. [5]Tadashi Horiuchi used modular neural networks for recognizing Kuzushiji characters.It consists of a rough-classifier and a set of fine-classifiers to recognize Kuzushiji script.

3. METHOD

3.1 DATASET

The training dataset is taken from the Kaggle website. It has a train.csv file which has image_id and labels. Image_id is the code for the image. Each image has a string of all labels. The string is represented as a space separated series of values like Unicode character, X, Y,Width and Height.

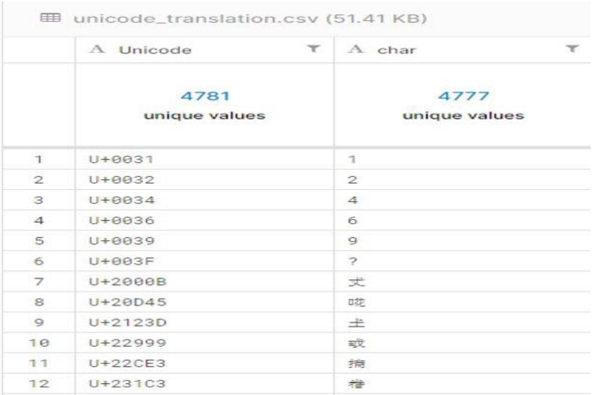


train.csv (14.95 MB)

	image_id	labels
	3605 unique values	3605 unique values
1	100241706_00004_2	U+306F 1231 3465 133 53 U+304C 275 1652 84 69 U+3044 1495 1218 143 69 U+3051 220 3331 53 91 U+306B 911 1452 61 92 U+306B 927 3445 71 92 U+306E 904 2879 95 92 U+5DE5 1168 1396 187 95 U+3053 289 3166 69...
2	100241706_00005_1	U+306F 1087 2018 103 65 U+304B 1456 1832 40 73 U+304B 2036 1722 65 76 U+3044 1709 1567 136 76 U+306B 1741 3329 43

FIGURE 2. train.csv

The dataset has a Unicode_translation.csv file which is a supplemental file mapping between unicode IDs and Japanese characters.



unicode_translation.csv (51.41 KB)

	Unicode	char
	4781 unique values	4777 unique values
1	U+0031	1
2	U+0032	2
3	U+0034	4
4	U+0036	6
5	U+0039	9
6	U+003F	?
7	U+2000B	𠀋
8	U+20D45	𠵅
9	U+2123D	𡈽
10	U+22999	𢦙
11	U+22CE3	𢳣
12	U+231C3	𣇃

FIGURE 3. Unicode_translation.csv

The dataset has a [train/test]_images.zip file which has images consisting of many characters and non-characters. Each character in an image can be of any size and can be repeated any number of times in an image.

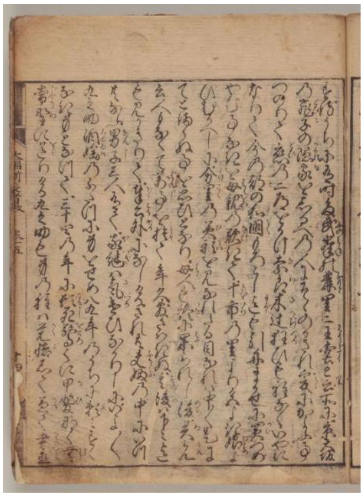


FIGURE 4. Train/test image

The proposed model is of two stages: detection and classification.

3.2 DETECTION

For detecting the characters in a picture,one of the well-known keypoint-based detectors is CornerNet is used. Later, CenterNet (Object as Points), which is one of the keypoint based detectors derived from CornerNet is applied to overcome few of the restrictions caused due to cornernet. CenterNet is used to predict the center pixel of the object with heatmap.

3.3 CLASSIFICATION

For classification, CNN model is used. Random cropping of image, LeakyReLu, average pooling, batchsize normalization are applied to extract the features from the trained set and is used to recognize and classify the characters.

4. IMPLEMENTATION

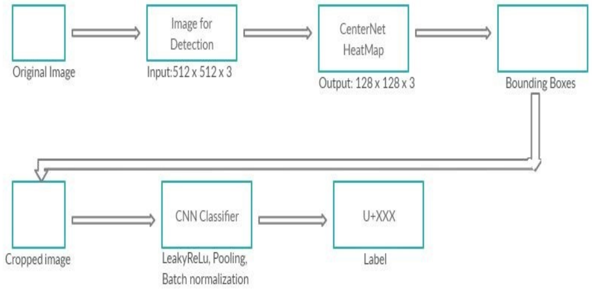


FIGURE 5. System architecture

First, the model for feature extraction is built, that is the layers of CNN model is built. Feature extraction is done by random cropping of the images, using activation functions like LeakyReLu, by average pooling and batch normalization. Second model is built to detect the characters in the image. It is done by using the concept of CenterNet, which is used to predict the center pixel of the character using heatmaps. Then, NMS (Non-maximum suppression) algorithm is used to obtain the best bounding boxes for each character. Third model is built to classify and recognize the characters. CNN is used for classification. Later all the weights obtained from these three models are pipelined. Figure 5 shows the system architecture. The input to the model for detection is of the size 512 x 512 x 3. Characters are detected using CenterNet and heatmaps. Then the bounding boxes are drawn using NMS algorithm. Then these images are cropped and sent to the CNN classifier where the initial convolution layer for feature extraction is built first. The CNN classifier recognizes the characters in every image of the test dataset and the corresponding Unicode of the characters with other labels is stored in a csv file.

4.1 RESULT AND ACCURACY ON THE TEST DATASET

Due to the hardware limitations, the accuracy of the model is 15.45%. With a lower learning rate, more number of iterations through the trained data set, with a higher value of epochs, the accuracy could be much better and high.

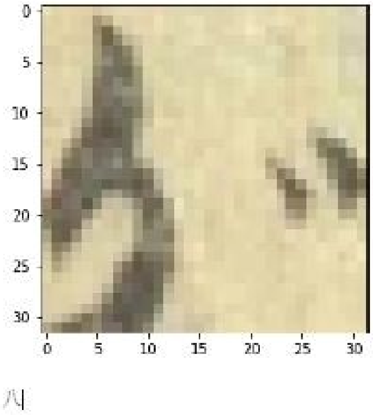
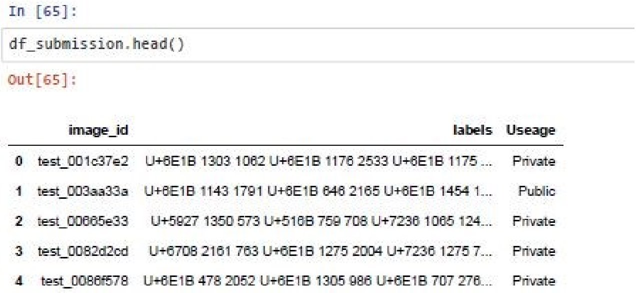


FIGURE 6. A kuzushiji character and its corresponding Japanese character



```
In [65]:
df_submission.head()
Out[65]:
```

```
     image_id                                                      labels  Useage
0  test_001c37e2  U+6E1B 1303 1062 U+6E1B 1176 2533 U+6E1B 1175 ...  Private
1  test_003aa33a  U+6E1B 1143 1791 U+6E1B 646 2165 U+6E1B 1454 1...   Public
2  test_00665e33   U+5927 1350 573 U+516B 759 708 U+7236 1065 124...  Private
3  test_0082d2cd   U+6708 2161 763 U+6E1B 1275 2004 U+7236 1275 7...  Private
4  test_0086f578  U+6E1B 478 2052 U+6E1B 1305 986 U+6E1B 707 276...  Private
```

FIGURE 7. The csv file in which the output is stored.
(Test image id with all the Unicode of the characters recognized)

5. CONCLUSION

This paper proposes a deep learning method to recognize the cursive, connected kuzushiji characters in an image. Our future task is to work more on the accuracy and also to translate kuzushiji characters to English alphabets.

6. REFERENCES

[1] Anh Duc Le, Tarin Clanuwat, Asanobu Kitamoto;" **A human-Inspired Recognition System for Pre-Modern Japanese Historical Documents"**, IEEE 2019

[2] Chulapong Panichkriangkrai; "**Internet-Based Interactive Transcription Support System for Woodblock-Printed Japanese Historical Book Images**"; IIAI-AAI 2018

[3] Yuta Hashimoto, Yoichi Iikura; "**The Kuzushiji Project: Developing a Mobile Learning Application for Reading Early Modern Japanese Texts**"; DHQ 2017

[4] K. Ikeda, R. Hayashi, K. Nagasaki, A. Morishima; "**Human-assisted OCR of Japanese books with different kinds of microtasks**"; iSchools, iConference 2017

[5] Tadashi Horiuchi; **"A Study on Japanese Historical Character Recognition Using Modular Neural Networks**"; ICICIC 2009

[6] https://www.kaggle.com/c/kuzushiji-recognition/overview