A Complete Methodology for Kuzushiji Historical Character Recognition using Multiple Features Approach and Deep Learning Model

Aravinda C.V[1]
Department of C.S.E
N.M.A.M Institute of Technology
Nitte, Karnataka
INDIA-574110

Lin Meng[2] and ATSUMI Masahiko[3]
Department of E.C.E
Ritsumeikan University,
Kusatsu
Japan

Udaya Kumar Reddy K.R[4]
and Amar Prabhu G[5]
Department of C.S.E
N.M.A.M Institute of Technology
Nitte, Karnataka
INDIA-574110

***Abstract*—As per the studies during many decades, substantial research efforts have been devoting towards character recognition. This task is not so easy as it it appears; in fact humans' have error rate about more than 6%, while reading the handwritten characters and recognizing. To solve this problem an effort has been made by applying the multiple features for recognizing kuzushiji character, without any knowledge of the font family presented. At the outset a pre-processing step that includes image binarization, noise removal and enhancement was applied. Second step was segmenting the page-sample by applying contour technique along with convex hull method to detect individual character. Third step was feature extraction which included zonal features (ZF), structural features (SF) and invariant moments (IM). These feature vectors were passed for training and testing to the various machine learning and deep learning models to classify and recognize the given character image sample. The accuracy achieved was about 85-90% on the data-set which consisted of huge data samples round about 3929 classes followed by 392990 samples.**

Keywords*—*Kuzushiji character; zonal features; structural features; invariant moments

I. Introduction

The huge quantity of data, either modern or historical, we have in our occupancy nowadays, because of expansions of digital libraries for reliable and accurate systems for processing. These documents are important because they are more significant part of our cultural heritage. Countless commercial products have been present and getting released to convert digitized documents into text files, either in the Unicode or ASCII format [1]. Unfortunately these products process only machine printed documents successfully, but if these machines fails when it comes to handwritten documents especially historical documents which results in poor performance [2] [3]. To solve this problems, recognition of historical documents is one of the most challenging tasks in recent days. The Japanese writing pattern is an arousing curiosity study of innovation and tradition. It combines a set of Chinese logo grams and two Chinese-derived syllabaries into a complex logo syllabic system. Scripting evolved to Japan from China during the 5th century. The first Japanese characters were written in Chinese characters (kanji), a system called kanbun [14]. The Japanese language, has incurred verbs and post positions, requiring concatenation of suffixes and particles to words and clauses in a sentence [9]. In order to overcome the grammatical units, the Japanese historical used certain Chinese characters for their sound values. This lead to the system which was ambiguous, and hard to tell whether a character [5].

In term of ancient Japanese documents, people used kuzushiji character for writing documents and publishing books. These ancient documents and books are found one by one currently, and waiting to be understood, which store a larger number of potential knowledge. However, few people know the kuzushiji character currently [7] [13]. And the kuzushiji characters have many variation, sometimes characters are connected one by one. Furthermore, aging causes the documents are uncleanness and worm-eaten happened. These problems increase the difficult of understanding these ancient documents. Especially, the traditional OCR(Optical Character Recognition) can not achieve better performance for the kuzushiji characters recognition [8] [12].

The problems in automatic recognition for totally unconstrained handwritten characters are greater than that of printed characters. Mistakes in reading the handwritten characters are more rates than of printed ones [4] [6].

A common OCR system will have several components, which illustrate the organization of usage. The input documents are scanned by external devices to produce gray-level image or binary bit-mapped image [10] [11]. This computational technique is knows as threshold.The flow of work is as shown in Fig. 1

A. Problem Statement

There are several problem related to handwritten historical kuzushiji character. The most important problems are notified and mentioned below

1) **Shape Discrimination:** A single character has variety of font style and a lot of free flow styles.
2) **Distorting of the character:** This is mainly because of the following reasons mentioned below
   - **Noise:** The reason is disconnected line segments, breaks in lines, isolated dots.
   - **Translation:** This is for the movement of full character or its elements.

- **Rotation:** This changes in orientation.

3) **Size variations:** The area of the character may be of 10, 15, 20 which specifies that are 10,15,20 characters per inch. 10 pitch are usually bigger than width and height than in 15. In continuation of these problems characters which are proportional spacing and variable line spacing.

II. Segmentation Process

At first the document was binarized a top-down segment approach. Next lines of the documents were detected followed by segmentation of individual characters is as shown in Fig. 3.

A. Pre-Processing

First the image was converted from 3d color image to gray-scale image is as shown in Fig. 4. Next the same image was binarized for black and white using Otsu threshold technique is as shown in Fig. 5.The binarized image was dialated using 20*20 kernel of 1s. The intention of this step for kuzushiji characters was applied mainly because of disconnected characters in document. For the better enhancement of the segmentation process the disconnected regions of the single character needs to be combined to make it as one character for recognition. The produced image was blurred by applying Gaussian Filter technique for noise removalis as shown in Fig. 2.

B. Canny Edge Detection

The Canny edge detector is an edge detection operator that uses a multi-stage algorithm to detect a edges in images. The filtered image was skeletonized by applying canny edge detection algorithm. The advantage of applying this algorithm for segmentation resulted to Noise reduction, Gradient calculation, Non-maximum suppression, Double threshold, Edge Tracking by Hysteresis for kuzushiji character is as shown in Fig. 6.

C. Contour Technique

The segmentation is one of the most challenging task in the tedious process for the separation or segregation of required information in the character. The contour is one of the best active models in segmentation process, for separation of region of interest.This active contour segmentation was used for the separation of pixels of interest for character segmentation is as shown in Fig. 7.

D. Convex Hull Technique

After applying the contour as mentioned in the Section II-C, next step was processed by applying the convex hull algorithm for better enhancement. The advantage of this algorithm was tend to be useful for the proposed problem. By using this convex hull algorithm a set of points was identified as the smallest convex polygon which enclosed all the points, since the character was curve in shape. The polygon was sketched around each individual character which enclosed all the points calculated in the previous Section II-C belonging to the individual character as shown in the Fig. 7 and 8. The draw back of convex hull found to be from the result was observed that convex hull gives an approximate bounding polygon which enclosed the entire character. As per the problem statement this was not feasible,since the area of the character should get segmented in the form of bounding box. To achieve this a bounding box was constructed form the points obtained after applying the convex hull technique as mentioned. A bounding box was drawn using 4 quadrant properties which are X and Y coordinates of top left corner, width and height of the bounding

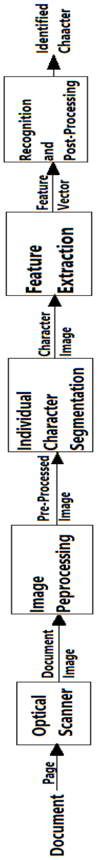

Fig. 1. Flow of Block Diagram for OCR

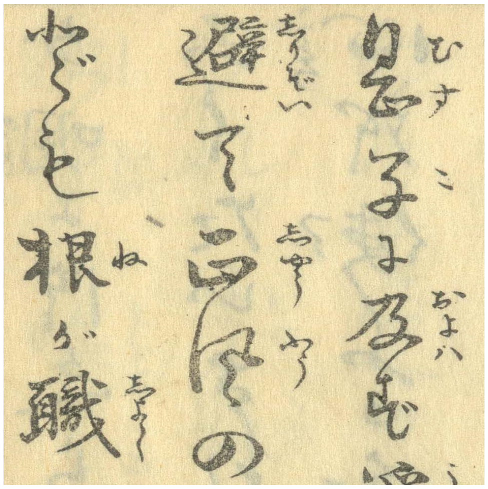

Fig. 2. Input of the Image

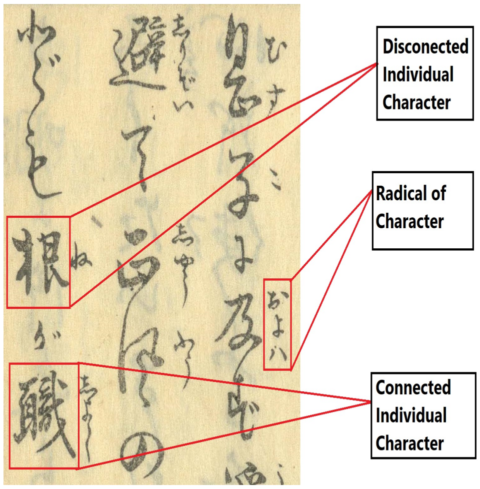
Disconected Individual Character
Radical of Character
Connected Individual Character

Fig. 3. Sub Section of Page-Labelled

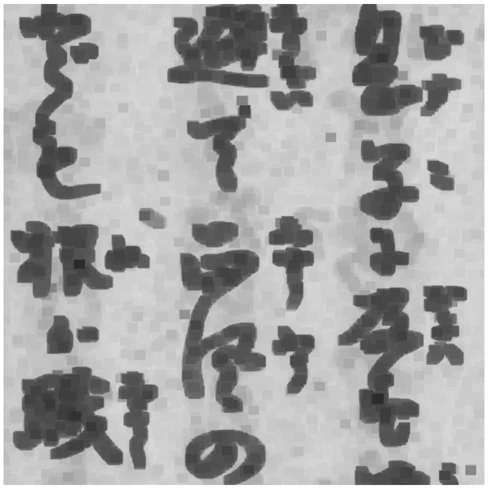

Fig. 4. Gray Scale

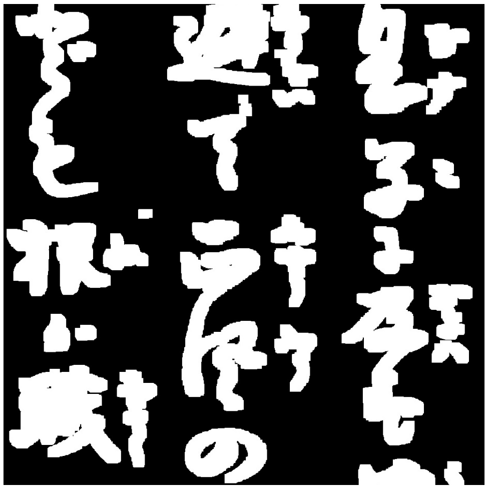

Fig. 5. Binary Image

box form the point of X, Y is as shown in Fig. 9.

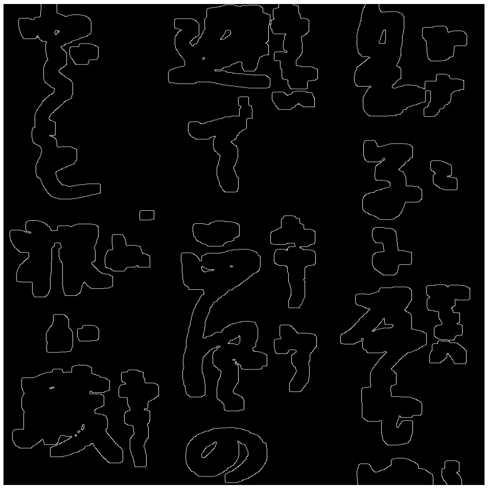

Fig. 6. Canny Edge Detection

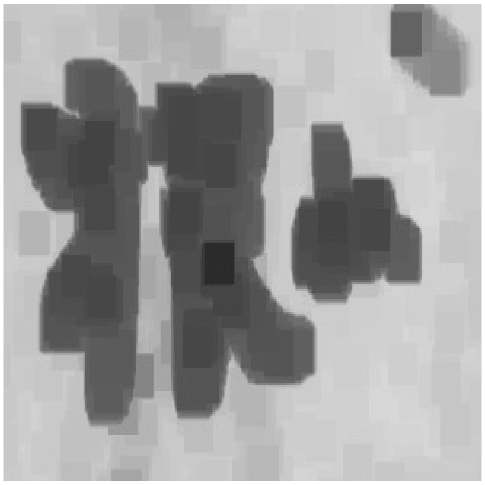

Fig. 7. Individual Character for Contour Technique

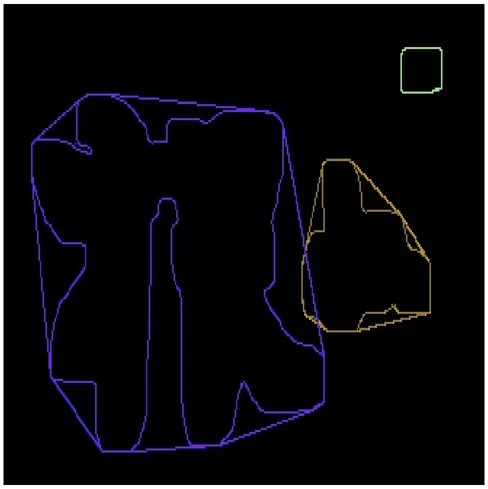

Fig. 8. Contour and Convex Hull of Character

E. Threshold

As mentioned in the previous Section II-D by applying the bounding box to all the obtained convex hull, which crops

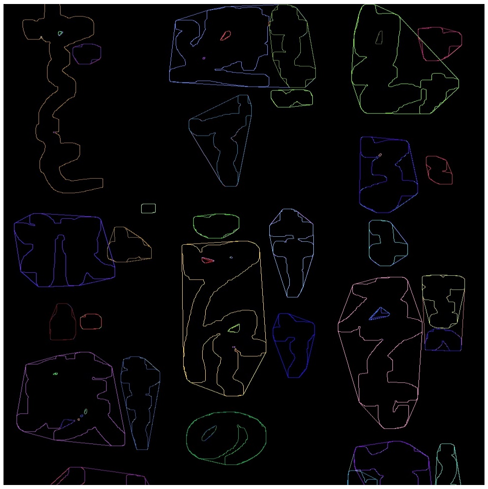

Fig. 9. Contour and Convex Hull for Sub-Section of Page

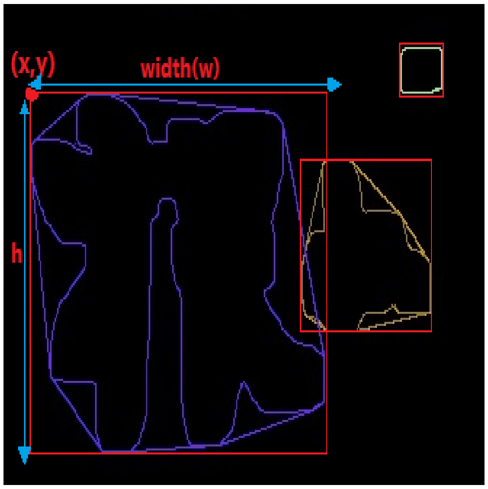

Fig. 10. Bounding Box

the area of interest of individual character along with minor distortion in the given page data sample and also the radicals' of the character is as shown in Fig. 11. However, the minor distortion and radicals' need to be ignored for recognition purpose. This critic point made to apply threshold for the said problem. In continuation of analysing the page samples it was found that mainly three types of regions from page were cropped. The representation of the experiment visual is as shown in the Fig. 10.

- The area of interest (Individual Character).
- The small distortion and noise along with radicals' whose width and height were smaller than the area of interest (Individual Character).
- The combination of characters written as 1 character whose width and height were much bigger than the area of interest.

F. K-means Clustering

To solve the threshold problem as mentioned in the previous Section II-E the width and height of the individual character was considered as as separate array and k-means clustering were applied on the array separately with 2 clusters under the assumption that all the characters, small radicals' and noises will be grouped as 1 cluster. Later all the combined characters and some of the individual character will be grouped in 2nd cluster. The minimum of the cluster center were taken as the height threshold and width threshold respectively.The complete flow of segmentation technique is as shown in the Fig. 11

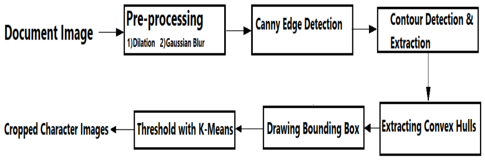

Fig. 11. Flow of Segmentation Technique

```
$T_{\mathrm{h}} = min\,(cluster\_center\,(height))$
$T_{\mathrm{w}} = min\,(cluster\_center\,(width))$
for every bounding box do:
if height $>$ $T_{\mathrm{h}}$ and width $>$ $T_{\mathrm{w}}$
crop character
else
ignore.
```

Finally, all the characters whose height was shown greater than $1/4^{\mathrm{th}}$ * page height were ignored. Hence the threshold condition can be modified as,

height of bounding box $>$ T_{h}
width of bounding box $>$ T_{w}
height of bounding box $<$ $1/4^{\mathrm{th}}$ * page height.

Experimentation found that this method extracted at-least 90% of the area of interest (Individual character)

III. Recognition Process Flow of Work Carried Out

A. Data Set Description

The huge data-sets were considered which consists of about 3929 classes kuzushiji characters for the model selection experiment.The total number of training images was more than 600,000 images. The kuzushiji data-set are obtained from Center for Open Data in the Humanities (CODH). Out of this there were three types of data-sets: 1)Katakana, 2) Kanji, 3) Hirangana.

The datasets was initially 64*64 image sample size which consisted of white text written on black background. The data-set was inverted with respect to color to get black text on white background which was resized to 100 * 100 for experimental purposes. Morphological opening and closing were carried out to remove salt and pepper noise on the image.

B. Feature Extraction

To recognize the particular character four types of features were extracted from the training set image:

1) Zonal feature: The given character image was divided into zones of equal width and height of the image. This image was divided into 25 zones where each

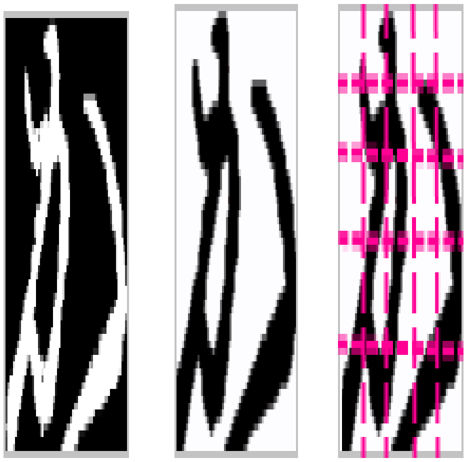

Fig. 12. Zonal Feature Projection Profile

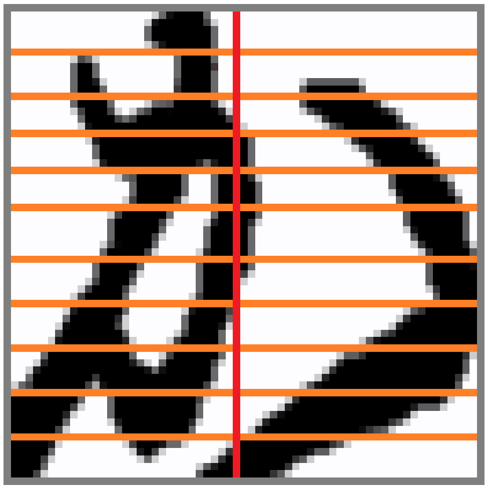

Fig. 13. Distance Feature Vertical Projection Profile

zone had width of 20 pixel and height of 20 pixel which makes 400 pixel in each zones.
The density of the black pixels in each zone were calculated in order to construct a feature vector histogram. From the zonal features a histogram of 25 bins were obtained which described the density in 25 zones of the image is as shown in Fig. 12.

2) Distance Feature: The distance feature was then extracted both from horizontal and vertical direction of the image is as shown in Fig. 13. **Vertical Profile:** A centroid was calculated in the vertical direction which was image width/2 for the mentioned problem statement 50 units was calculated is as shown in Fig. 14. The image was divided into two sections based on this centroid namely left section and right section. Each section was then subdivided into 10 subsections. In each of the subsection, Euclidean distance between the furthest pixel in the sub-section $(i.e, pixel - closest - to - the - outer - boundary)$ and the center point of the centroid for that section was found. $distance = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$ This created a 20 bin histogram which was appended to the feature vector.
3) Invariant Moments: Similar to Vertical profile, a centroid was found in the horizontal direction with $c_{\mathrm{h}} = imageheight/2$, which resulted in 50 units. The image was divided into 2 sections upper X and lower section Y, these sections were divided into 10 subsections to find the distance between the furthest black pixel and cnetroid of subsection. This created a 20 bin histogram which was appended to the feature vector array.
4) Hu Moments: By using this feature moments 7 bin histogram was created. The total size of the feature vector resulted in $25 + 20 + 20 + 7 = 72$bins.

IV. Classification

A. Support Vector Machine

SVM is generally useful for statistical learning and determining the point location of decision boundaries which results the optimal separation of classes. The SVC classifier was used in implementing the “one-against-one” approach for multi- class classification problem where the label’s were drawn from finite set of several elements.The samples of kuzushiji characters, round about 3923 class was taken as the number of classes, then this $(3923 * 3923 - 1)/2$ classifiers are constructed and each one samples was trained data from two classes. The decision function shape option allows to transform the results of the “one-against-one” classifiers to a decision function of shape $(297300 samples, 3923 classes)$. Applying each classifier to the test data vectors gives one vote to the exact class. The results of a recent analysis of multi-class strategies are provided in Fig. 15.

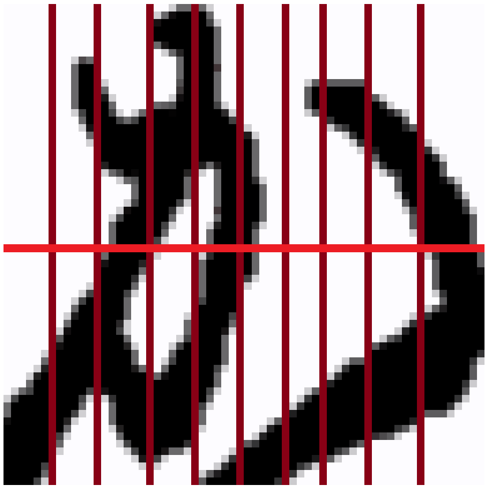

Fig. 14. Distance Feature Horizontal Projection Profile

NOTE Visualization of support vector classifier. The graph is plotted for 25 sample points which enclose only 2 features since the data samples is huge for reference is as shown in Fig. 15.

B. Neural Network Classifier

The usage of Neural nets was taken for the classification and recognition, since it consists of artificial network of

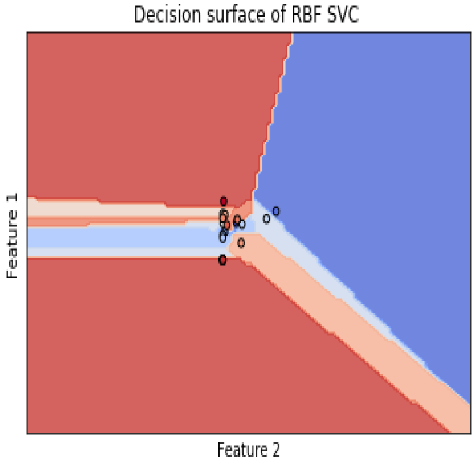

Fig. 15. Distance Surface of Radial Bias Function

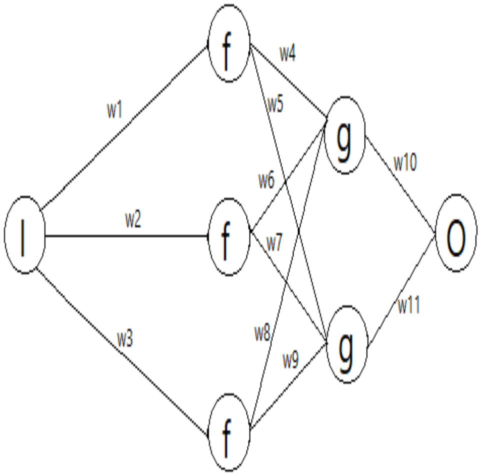

Fig. 16. Simple Neural Network Architecture

TABLE I. Architecture of Simple Neural Network

Layer (type)	Output Shape	Param
dense1 (Dense)	(None, 72)	5256
dense2 (Dense)	(None, 820)	59860
dropout1 (Dropout	(None, 820)	0
dense3 (Dense)	(None, 1640)	1346440
dropout2 (Dropout)	(None, 1640)	0
dense4 (Dense)	(None, 3923)	6437643

functions called parameters which was able to learn all the feature of the images for analyzing the new data after receiving one or multiple inputs as shown in the Fig. 16 followed by the validation results is as shown in the Fig. 17 and the layers is as mentioned in the Table I.

C. Accuracy Comparison Table

The results as shown in the Table II

TABLE II. Results Comparison of Classifiers

Layer (type)	Output Shape	Param
Sl.No	Model	Accuracy
1	Support Vector Machine	87.4%
2	Neural Network Classifier	90%

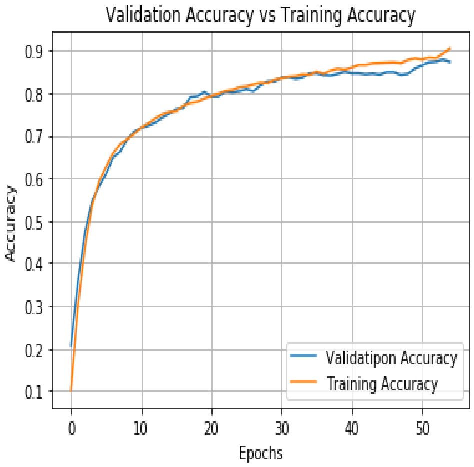

Fig. 17. Validation and Training Accuracy

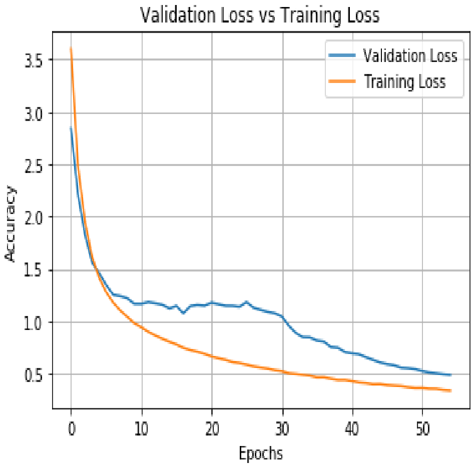

Fig. 18. Validation Loss and Training Loss

V. Conclusion

In this research work a complete methodology and multiple feature extraction technique was applied for historical documents for recognition purposes [15]. This methodology can be applied to either machine printed or handwritten documents. It is not necessary nor any prior knowledge of the fonts nor the existence of standard database because it can adjust depending on the type of documents that needed to process [16].

VI. Future Work

The next work is to focus on optimizing the recognition rate and finding out the new algorithm for feature extraction for better enhancement.

VII. Acknowledgment

This work was supported by a Grant-in-Aid for Scientists (18K18337) from the Japan Society for the Promotion of Science (JSPS), and the Ritsumeikan University Art Research Center.

References

[1] T.M.Rath and R. Manmatha, "Word spotting for historical documents", International Journal on Document Analysis and Recognition (IJDAR), Vol.9, No 2 – 4, pp. 139 – 152 , 2006

[2] V. Lavrenko, T. M. Rath, R. Manmatha: "Holistic Word Recognition for Handwritten Historical Documents", Proceedings of the First International Workshop on Document Image Analysis for Libraries (DIAL'04),pp 278287, 2004.

[3] T. Adamek, N. E. O'Connor, A. F. Smeaton, "Word Matching Using Single-Closed Contours for Indexing Handwritten Historical Documents", International Journal on Document Analysis and Recognition (IJDAR), special Issue on Analysis of Historical Documents, 2006.

[4] T. Konidaris, B. Gatos, K. Ntzios, I. Pratikakis, S. Theodoridis and S. J. Perantonis, " Keyword - Guided Word Spotting in Historical Printed Documents Using Synthetic Data and User feedback ", International Journal on Document Analysis and Recognition (IJDAR), special issue on historical documents, Vol. 9, No. 2-4, pp. 167-177, 2007.

[5] V.G.Gezerlis and S.Theodoridis, "Optical Character Recognition for the Orthodox Hellenic Byzantine music notation", Pattern Recognition, Vol.35, pp. 895 – 914, 2002.

[6] L. Laskov, "Classification and Recognition of Neume Note Notation in Historical Documents", International Conference of Computer Systems and Technologies (CompSysTech), 2006.

[7]] K. Ntzios, B. Gatos, I. Pratikakis, T. Konidaris and S.J. Perantonis, "An Old Greek Handwritten OCR System based on an Efficient Segmentation-free Approach", International Journal on Document Analysis and Recognition (IJDAR), special issue on historical documents, Vol. 9, No. 2-4, pp. 179-192, 2007.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. 2015 "Deep Residual Learning for Image Recognition." arXiv:1512.03385

[9] G.Huang, Z.Liu, L.van der Maaten, K.Q.Weinberger. 2016 "Densely Connected Convolutional Networks." 2016. IEEE Conference on Pattern Recognition and Computer Vision (PRCV2016)

[10] Szegedy, C., Liu, W., Jia, Y.Q., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. 2016 "Rethinking the Inception Architecture for Computer Vision." IEEE Conference on Pattern Recognition and Computer Vision(PRCV 2016)

[11] Francois Cholle. 2017 "Xception: Deep Learning with Depthwise Separable Convolution." IEEE Conference on Pattern Recognition and Computer Vision(PRCV 2017)

[12] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Wey, Marco Andreetto and Hartwig Adam. 2017. "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications." arXiv:1704.04861.

[13] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, Liang-Chieh Chen. 2018. "Mobilenetv2: Inverted residuals and linear bottleneck." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR 2018).

[14] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu and Alexander C. Berg, 2018. "SSD: Single Shot MultiBox Detector." arxiv:1512.02325

[15] http://codh.rois.ac.jp/char-shape/book/ (2020.2.26 accessed)

[16] Aravinda C.V, Meng Lin, and Amar Prabhu G. "Kuzashi recognition API" http://www.atait.se.ritsumei.ac.jp/KuzushijiMser/ (2020.2.26 accessed)