

# Character Segmentation in Collector’s Seal Images: An Attempt on Retrieval Based on Ancient Character Typeface

Kangying Li<sup>1</sup>, Biligsaikhan Batjargal<sup>2</sup> and Akira Maeda<sup>3</sup>

<sup>1</sup>Graduate School of Information Science and Engineering, Ritsumeikan University, Japan

<sup>2</sup>Kinugasa Research Organization, Ritsumeikan University, Japan

<sup>3</sup>College of Information Science and Engineering, Ritsumeikan University, Japan  
gr0319ss@ed.ritsumeik.ac.jp

## Abstract

The collector’s seal is a stamp for the ownership of the book. There is many important information in the collector’s seals, which is the essential elements of ancient materials. It also shows the possession, the relation to the book, the identity of the collectors, the expression of the dignity, etc. Majority of people from many Asian countries usually use artistic ancient characters to make their own seals instead of modern characters. A system that automatically recognizes these characters can help enthusiasts and professionals better understand the background information of these seals more efficiently. However, there is a lack of training data of labeled images, and many images are noisy and difficult to be recognized. We propose a retrieval-based recognition system that focuses on single character to assist seal retrieval and matching.

## 1 Background

The use of the collector’s seals in Asian ancient books is a worthy topic of detailed discussions. The collector’s seal has the functions of expressing the sense of belonging, demonstrating the inheritance, showing the identity, representing the dignity, displaying aspiration and interest, identifying the version, expressing speech, etc., in different forms. For individual collectors, in addition to their own names, there are also words, which can show their personal or their family situations including ancestral origin, residence, family background, family status, rank and so on. For example, Natsume Soseki, a famous Japanese writer, had many kinds of collector’s seals. The foreign books, which he had collected during his life, were recorded using various collector’s seals that he used in different periods. We can get wider information about his book-collecting hobbies by analyzing these collector’s seals. One of the most important functions of the collector’s seals is to record where a book has been kept and the footsteps of history of a book being handed over. For book collecting institutions, the migration process of the books and their purchase histories can be also reflected in the contents of the collector’s seals. The names of many book collecting institutions have been gradually changed with the historical changes. Through the collector’s seals, we can understand the books themselves, their collectors, or more background information of the collecting institutions. Through collector’s seals in libraries, we can find out the

collective experience and the source of inheritance of a book. In Asian countries where kanji characters are used, ancient characters are usually used to make collector's seals. Also, as time passes, the shape of kanji may be changed, and multiple variations of a character might be created. There are also characters derived from kanji characters, or the characters that just look like kanji characters. For example, Vietnam's "Chữ Nôm" is a character system that are originated from kanji's original shape. Therefore, it is hard for non-professionals to understand all the contents of a collector's seal or a single character in it. Meanwhile, as far as a single ancient character is recognized, utilizing the data from scholars studying kanji characters, we can also have a broader understanding of ancient character's culture. Especially for individual collectors, exploring the information in every single character of their names is an important method for us to get a comprehensive understanding of their family affairs.

It is expected to construct a retrieval-based ancient character recognition system that can match the user's query character even if there is only one labeled typeface image, and can update the character type at any time instead of retraining a model. Therefore, we propose a text extraction system for image data of collector's seals, and this system is also expected for the future research in automatic text feature generation to find the background information of Asian ancient books from external databases using the extracted text contents in the character recognition task. In this research, we perform character segmentation using Mean-shift clustering and retrieve a single ancient character from ancient character typeface images by using the extracted features.

## 2 Related Work

**Character segmentation for off-line recognition.** Nguyen et al. (2016)[1] proposed a text-line segmentation method. They proposed a morphological method, zone projection and character separation for each segmented text-line by vertical projection, Stroke Width Transform (SWT), bridge finding and Voronoi diagrams to get the results. This method handles the segmentation task of Japanese handwritten characters very well. However, seal characters usually have irregular positions or distributions and large differences in character size, which may cause bad influence for the segmentation result by using the proposed method. Zahan et al. (2018) [2]proposed a segmentation process for printed Bangla script, and this method has a good performance on segmenting the characters with topologically connected structure. However, the target image of this method is quite different from our target image. Hence, we propose a method to deal with the problem of irregular character distributions.

**Seals image retrieval.** Fujitsu R&D Center (2016) [3]announced a seal retrieval technique for Chinese ancient document images. By applying color separation technique to split the seal from the background image, they proposed a two-level hierarchical matching method based on global feature matching. Unfortunately, we have not been able to find any relevant literature that describes the details of this technique. This system is aimed at the whole content of seals, and retrieval scope depends on the existing seal data in the database. Because most of the seals are personal assets and usually consist of meaningful, separate and independent characters. Therefore, we propose a method to search and match the seal images by splitting it to character units.

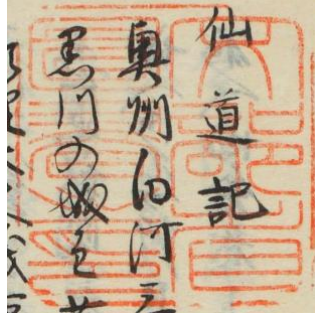
## 3 Methodology

We describe our approach in the following sections. Since we only extract features from typeface images and store them in a database, we perform some pre-processing for user-provided seals images. We describe our data pre-processing in Section 3.1. In the beginning of Section 3.2, we describe the

process of extracting essential features from images. These features include deep features and geometric features, therefore in Section 3.2, we also describe the deep features that we use. In Section 3.3, we describe the extraction of geometric features. Introduction of the feature matching and ranking calculation will be shown in Sections 3.4 and 3.5.

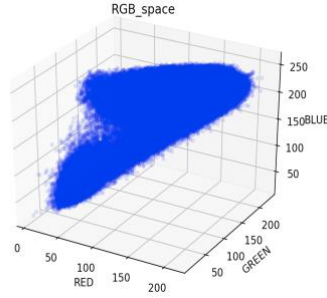
### 3.1 Data pre-processing and character segmentation

As shown in Figure 1, in classical materials, seals often overlap with handwritten words.



**Figure 1:** An example of a seal overlaps with handwritten words.

Therefore, splitting the seal pattern from the image is an important task. We use k-means clustering (Hartigan, et al.,1979) [4] to cluster image color information. As shown in Figure 2, We can project RGB three channels information of an image into the three-dimensional space.

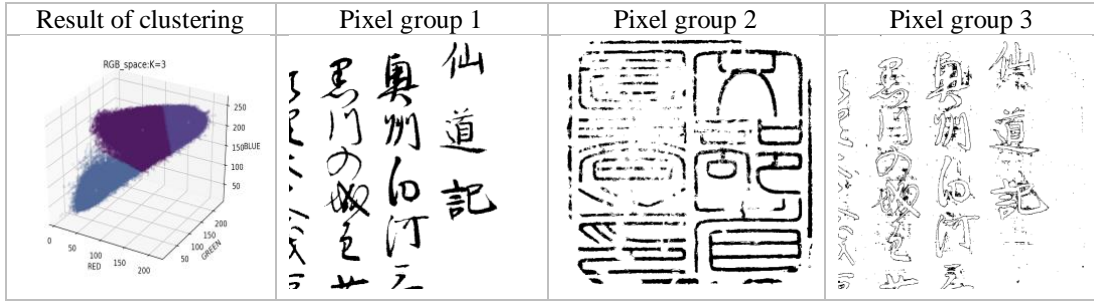


**Figure 2:** Three-dimensional representation of color information

Hence, the Euclidean distance is used to represent the color relationship between two pixels which is defined by (1).

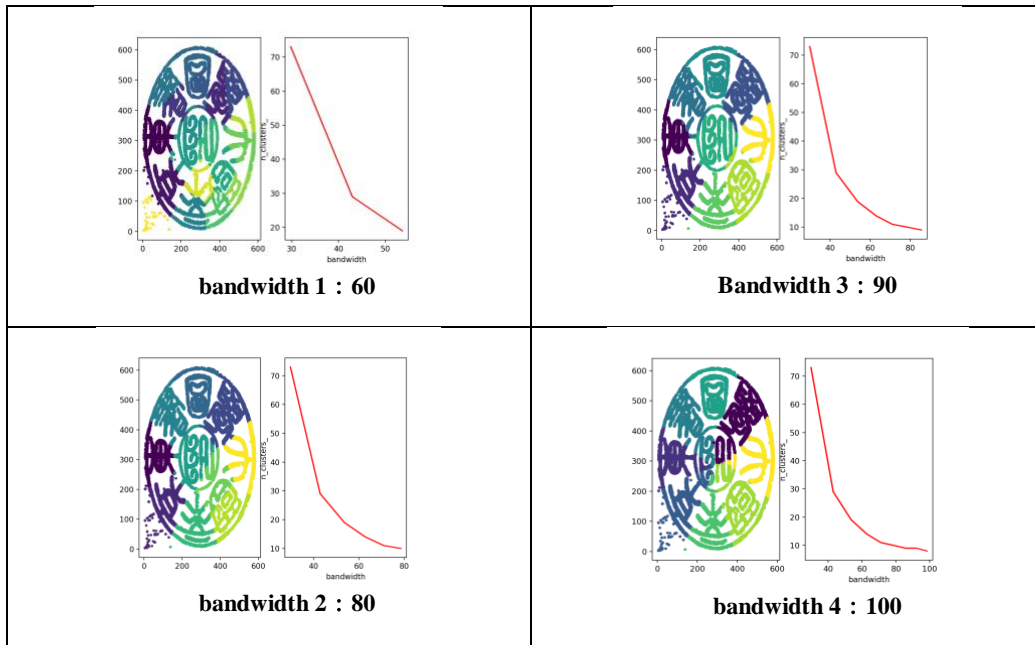
$$D_{rgb} = \sqrt{(R_1 - R_2)^2 + (G_1 - G_2)^2 + (B_1 - B_2)^2} \quad (1)$$

Where  $R_1, R_2, G_1, G_2, B_1, B_2$  represent RGB values for pixels 1 and 2 respectively, and we use  $D_{rgb}$  to cluster pixels with similar colors. According to the principle of k-means algorithm, we regard this task as extracting K groups of pixels with similar colors from images. Our system automatically extracts areas with more red components. As shown in Figure 3, the Pixel group 2 is extracted as the analysis target of our system.



**Figure 3:** Results of k-means clustering.

For the determination of a single character, we use the Mean-shift clustering to segment each character. Because kanji characters are independent and balanced in structure, we regard every character as a module, and each module has its centroid. By using these centroids, we can use a clustering to extract character fields. Since we already know the coordinates of each pixel of the seal areas, the density information of each pixel can be obtained by kernel density estimation. We use Mean-shift clustering (Comaniciu, et al.,1999) [5] to cluster the pixels of an image. The clustering results can be optimized by adjusting the variable bandwidth, under the influence of different variables bandwidth. Figure 4 shows the results of clustering under different bandwidth settings.



**Figure 4:** Visualization of clustering results under different bandwidth.

In each unit, the graph on the left side is the clustering result of each pixel of the seal, and each color represents a cluster, and as for the graph on the right side, the X-axis is the value of bandwidth, and the Y-axis is the number of clusters. The results that we need can be selected in the process when change rate of the total number of clusters becomes steady, for example, when the bandwidth equals to 90, we suppose each cluster in the result is treated as a candidate result of character segmentation. Hence, we

calculate a bandwidth interval, and the bandwidth value is obtained equidistantly in the interval. By using these bandwidth values, segmentation candidates are obtained. The algorithm flow is shown in Algorithm 1.

---

**Algorithm 1:** Image segmentation

---

**Input:** coordinate set  $\{ (x_1, y_1), (x_2, y_2) \dots (x_n, y_n) \}$  from non-background area obtained by K-means clustering

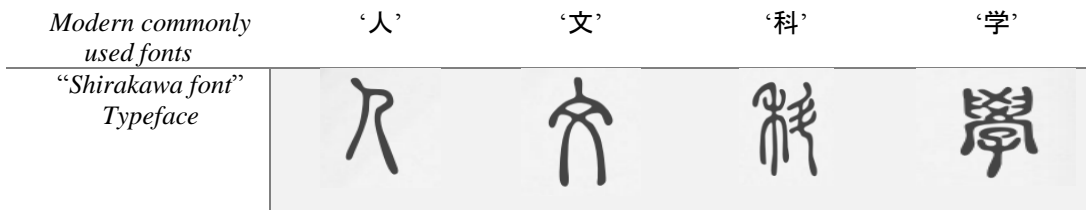
**Output:** Set of object location hypotheses  $U$

- 1: Initialize bandwidth interval :  $[Bandwidth_{min}, Bandwidth_{max}] = \{ Bandwidth \in \mathbb{R} : \{ Bandwidth_{min} < Bandwidth_{max} \}, \text{ take the value at a moderate distance in the interval to get bandwidth set } \{ b_1, b_2, b_3 \dots b_n \in [Bandwidth_{min}, Bandwidth_{max}] \}$
  - 2: Get the result of number of clusters  $\{ Nclusters\_b1, Nclusters\_b2 \dots Nclusters\_bn : Nclusters\_bn \in f_{\text{meanshift\_clustering}}(b_n) \}$  for each bandwidth using Mean-shift clustering
  - 3: Find the value of  $b_n$  when  $Nclusters\_bn$  become to the minimum, take  $b_n$  as the *Endbandwidth*.
  - 4: Fit a polynomial  $Q(b)$  with  $\{ (b_1, Nclusters\_b1), (b_2, Nclusters\_b2) \dots (b_n, Nclusters\_bn) \}$  using least squares polynomial fit.
  - 5: Get the second derivative  $Q''(b) = (\frac{d^2 Nclusters\_bn}{db_n^2})$  of  $Q(b_n)$
  - 6: **foreach**  $b_i \in \{ b_1, b_2, b_3 \dots b_n \in [Bandwidth_{min}, Bandwidth_{max}] \}$  **do**:  
     **if**  $Q''(b_i) = \text{Min}\{ (Q''(b_{i+1}) - Q''(b_i)) / Q''(b_i) \}$  **then**  
         Initial *Initbandwidth*  $\leftarrow b_i$   
     **else** continue
  - 7: Obtain regions  $U = \{ u_1, u_2, u_3, u_4 \dots \}$  using Mean-shift clustering with bandwidth during  $[Initbandwidth, Endbandwidth]$
- 

The algorithm implementation is available on GitHub (Li, K et al., 2019)[6] for your reference.

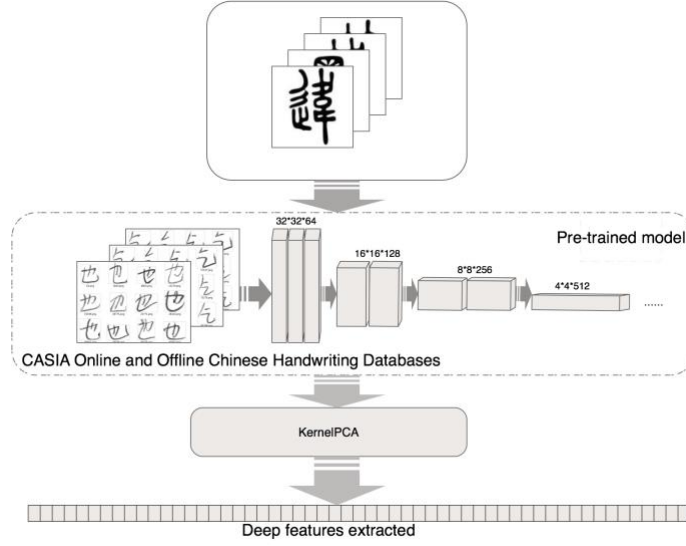
### 3.2 Extracting CNN features from images

The typeface images converted from the font file (Shirakawa Font project, 2016) [7] are shown in Figure 5.



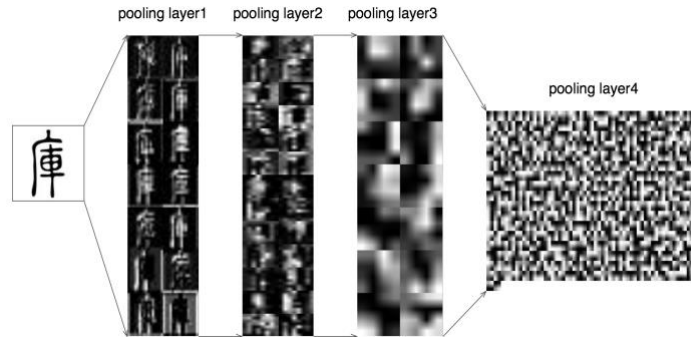
**Figure 5:** Images converted from a font file.

First, we normalize the typeface image. Then we crop the typeface image according to maximum and minimum coordinates of the black pixels, and then standardize them to the size of  $64 \times 64$ . As there are many variations of ancient characters, even a slight change of the structure will affect the extraction of geometric features. We use the pre-trained model to extract the deep features (convolutional neural networks (CNN) features) of fonts, trying to subtract some minor changes in character's structure, to assist the characters of same category become closer to each other in the feature space.



**Figure 6:** Extractor of CNN features.

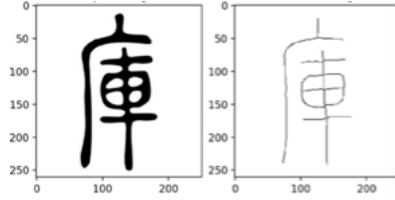
As shown in Figure 6, due to the lack of ancient character dataset, we selected CASIA Online and Offline Chinese Handwriting Databases (Liu et al., 2011)[8], in which the characters have some similar shape feature with the ancient characters as the training data to train the model using VGG16 (Simonyan et al., 2014)[9]. The visual expression of feature map in the max pooling layer of the pre-trained model is shown in Figure 7. We use kernelPCA (Mika et al., 1999)[10] to reduce the dimension of the output from the middle layer.



**Figure 7:** The visual expression of feature map in the max pooling layer.

### 3.3 Extracting geometric features from images

To capture some details of the character's structure, we attempted to extract the geometric features of these characters.

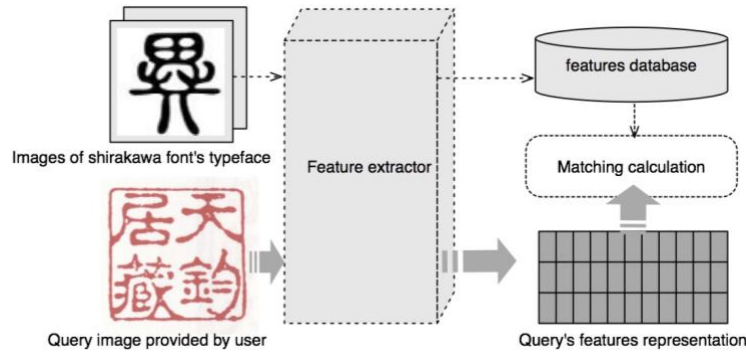


**Figure 8:** The skeleton map of a character.

As shown in Figure 8, using the method proposed by Zhang's research (Zhang et al.,1984)[11], a character's skeleton map is obtained. Different from general thinning method, Zhang's method ignores the existence of the stroke width, and can obtain a skeleton map without noise, so each stroke can be represented by a unique corresponding continuous single pixel. Then we use Harris corner (Harris et al.,1988)[12] to obtain the coordinates of the intersection of each stroke. The coordinate points of skeleton map and the coordinate points of stroke intersections are stored in the database as representations of geometric features.

### 3.4 Image matching using multiple features

We use the following method for calculating similarity to match the typeface image and the query image. The matching process is shown in Figure 9. The segmented user query image and the typeface image use the same feature extraction method to extract the corresponding features including CNN features and geometric features.



**Figure 9:** Matching process.

The cosine similarity is used to compare the similarity of CNN features, and Hausdorff distance (Huttenlocher et al.,1992) [13] is used to compare the similarity of geometric features between two images.

### 3.5 Ranking calculation

Since the features are different, and value ranges are totally different, similarity scores of different features cannot be directly compared. Hence, we convert the results of each similarity calculation to scores. Then each feature is given different weights. The results of Hausdorff distance are normalized to values in the range [0, 1] by using Min-Max normalization, and the function is defined by (2).

$$S_{i\_hausdorff} = \frac{\frac{1}{D_{hausdorff,i}} - \min\{\frac{1}{D_{hausdorff,i}}\}}{\max\{\frac{1}{D_{hausdorff,i}}\} - \min\{\frac{1}{D_{hausdorff,i}}\}} \quad (2)$$

Where  $i=0, 1, \dots, N$ ,  $N$  is the number of images in the database,  $S_{i\_hausdorff}$  is the similarity score between image  $i$  and the query image, and  $D_{hausdorff,i}$  is the result of Hausdorff distance between image  $i$  and query image calculated by using Hausdorff distance. The result of multi-feature similarity score is calculated by (3).

$$S_{total} = \frac{W_{cf}S_{cnnFeature} + W_{gf}S_{geometricFeatures}}{W_{cf} + W_{gf}} \quad (3)$$

Where  $S_{total}$  is the total score of similarity,  $S_{geometricFeatures}$  is the sum of similarities of geometric features which calculated use  $S_{i\_hausdorff}$ .  $W_{cf}$  and  $W_{gf}$  are the weights of similarity score of CNN feature  $S_{cnnFeature}$  and geometric feature  $S_{geometricFeatures}$ . Finally, we use the total score of similarity to predict the input image's category.

## 4 Experiments and Results

In this section, we will describe our experiments in three parts. In section 4.1, we will introduce the database used in the experiments. The results of image segmentation and image retrieval will be explained in Sections 4.2 and 4.3.

### 4.1 Datasets




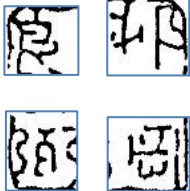



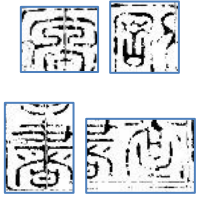
We select the test data from Collectors' Seal Database (National Institute of Japanese Literature, available from 2011)[14]. This database contains 39,686 images of collector's seals, including only pictorial seals and seals carved in various types of calligraphy. The main color of the seals is red. Because the original data does not mark the coordinate information of a single character, in this experiment we counted the characters with high frequency in this database, and marked their position information as a small-scale experimental object.

### 4.2 Result of segmentation

Different types of seals are used to test our proposed segmentation algorithm, and the results are shown in Table 2. From the experiment results, it can be seen that our proposed method achieves good results in processing images with irregular characters distribution. Nevertheless, from the Result 3, larger characters are segmented into other character candidate area, thus more efforts are still needed to



conduct further research on dealing with images with different character sizes and close character spacing.

1: Regular seal	Result 1	2: Characters with irregular glyph	Result 2
			
3: Irregular character distribution	Result 3	4: With noisy background and overlaps with handwritten words	Result 4
			

**Table 1:** Segmentation results.











### 4.3 Retrieval results

Here we show the Mean Reciprocal Rank (4) results of ten characters with the highest frequency in the printed text. For each character, we tested with twenty images.

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i} \quad (4)$$

where  $Q$  is the total number of images retrieved,  $i$  is the image number, and  $rank$  is the ranking order.

When all the weights are set as initial value 1, the results are shown in Table 2. We found that characters with simple shapes showed better results.

Characters	MRR score	Characters	MRR score
	0.008		0.258
	0.016		0.488
	0.115		0.256
	0.142		0.121
	0.028		0.482

**Table 2:** Retrieval results.

## 5 Conclusion

In this study, we used two clustering algorithms to preprocess seal images and have obtained the good results. Not alike to train a neural network, clustering algorithm is a method that can extract part of the required information from an image without consuming a lot of computational resources. Then we use the combination of deep features and geometric features to retrieve ancient kanji characters through the calculation of similarities. It reduces the time of re-training a model when adding new category of characters and also enables flexible use of intermediate output of neural networks. We can know which seal belongs to which famous person using the recognition results, then we can learn more about this person's hobbies from his or her collection's information, and this will be the next target of our research. However, the image retrieval performances need to be further improved. How to make better use of only one typeface image is needed to be focused in our future research.

## References

- [1] Nguyen, K. C., Nakagawa, M., (2016) Text-line and character segmentation for offline recognition of handwritten Japanese text. IEICE technical report. (pp.53-58)
- [2] Zahan, T., Iqbal, M. Z., Selim, M. R., Rahman, M. S., (2018). Connected Component Analysis Based Two Zone Approach for Bangla Character Segmentation. In 2018 International Conference on Bangla Speech and Language Processing (ICBSLP) (pp.1-4), IEEE.
- [3] Fujitsu Research & Development Center Co. Ltd. Seal Retrieval Technique for Chinese Ancient Document Images. Retrieved from: <https://www.fujitsu.com/cn/en/about/resources/news/press-releases/2016/frdc-0330.html>
- [4] Hartigan, J. A., Wong, M. A., (1979). Algorithm AS 136: A k-means clustering algorithm. Journal of the Royal Statistical Society. Series C (pp.100-108).
- [5] Comaniciu, D., Meer, P. (1999). Mean shift analysis and applications. In Proceedings of the Seventh IEEE International Conference on Computer Vision (pp. 1197-1203), IEEE.
- [6] Li, K., Batjargal, B., Maeda, A., (2019) Seal Character segmentation. Retrieved from [https://github.com/timcanby/collector-s\\_seal-ImageProcessing](https://github.com/timcanby/collector-s_seal-ImageProcessing)
- [7] The Shirakawa Shizuka Institute of East Asian Characters and Culture, Shirakawa Font project. Retrieved from: <http://www.ritsumei.ac.jp/acd/re/k-rsc/sio/shirakawa/index.html>
- [8] Liu., Cheng. L., CASIA online and offline Chinese handwriting databases. Document Analysis and Recognition (ICDAR2011), IEEE.
- [9] Simonyan, K., Zisserman, A., (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
- [10] Mika, S., Schölkopf, B., Smola, A. J., Müller, K. R., Scholz, M., Rätsch, G. (1999). Kernel PCA and de-noising in feature spaces. In Advances in neural information processing systems (pp. 536-542).
- [11] Zhang, T. Y., Suen, C. Y. (1984). A fast parallel algorithm for thinning digital patterns. (27(3), pp.236-239), ACM.
- [12] Harris, C. G., Stephens, M. (1988). A combined corner and edge detector. In Alvey vision conference (Vol. 15, No. 50, pp. 10-5244).
- [13] Huttenlocher, D. P., Rucklidge, W. J., Klanderman, G. A. (1992). Comparing images using the Hausdorff distance under translation. In Proceedings 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (pp. 654-656), IEEE.
- [14] National Institute of Japanese Literature, Collectors' Seal Database, Retrieved from [http://base1.nijl.ac.jp/~collectors\\_seal/](http://base1.nijl.ac.jp/~collectors_seal/)