**ORIGINAL PAPER**

# An unsupervised automatic organization method for Professor Shirakawa's hand-notated documents of oracle bone inscriptions

Xuebin Yue[1,2] · Ziming Wang[3] · Ryuto Ishibashi[3] · Hayata Kaneko[3] · Lin Meng[3]

**Abstract**

As one of the most influential Chinese cultural researchers in the second half of the twentieth-century, Professor Shirakawa is active in the research field of ancient Chinese characters. He has left behind many valuable research documents, especially his hand-notated oracle bone inscriptions (OBIs) documents. OBIs are one of the world's oldest characters and were used in the Shang Dynasty about 3600 years ago for divination and recording events. The organization of OBIs is not only helpful in better understanding Prof. Shirakawa's research and further study of OBIs in general and their importance in ancient Chinese history. This paper proposes an unsupervised automatic organization method to organize Prof. Shirakawa's OBIs and construct a handwritten OBIs data set for neural network learning. First, a suite of noise reduction is proposed to remove strangely shaped noise to reduce the data loss of OBIs. Secondly, a novel segmentation method based on the supervised classification of OBIs regions is proposed to reduce adverse effects between characters for more accurate OBIs segmentation. Thirdly, a unique unsupervised clustering method is proposed to classify the segmented characters. Finally, all the same characters in the hand-notated OBIs documents are organized together. The evaluation results show that noise reduction has been proposed to remove noises with an accuracy of 97.85%, which contains number information and closed-loop-like edges in the dataset. In addition, the accuracy of supervised classification of OBIs regions based on our model achieves 85.50%, which is higher than eight state-of-the-art deep learning models, and a particular preprocessing method we proposed improves the classification accuracy by nearly 11.50%. The accuracy of OBIs clustering based on supervised classification achieves 74.91%. These results demonstrate the effectiveness of our proposed unsupervised automatic organization of Prof. Shirakawa's hand-notated OBIs documents. The code and datasets are available at http://www.ihpc.se.ritsumei.ac.jp/obidataset.html.

**Keywords** Oracle bone inscriptions · Noise reduction · Image processing · Morphological processing · Unsupervised classification

## 1 Introduction

Oracle Bone Inscriptions (OBIs) were written from the late Shang Dynasty to the Western Zhou Dynasty in ancient China. They primarily record divination and sacrifice information on tortoise shells and animal bones. OBIs are one of the oldest surviving characters in China and one of the earliest characters in the world [1–3]. As the originator of the Chinese character system, the organization of OBIs helps archaeologists analyze Chinese characters and study the history of ancient China for the better. However, the oracle bone pieces (OBPs) that recorded the OBIs have been buried in the Yin Ruins because of warfare until their excavation in 1899 [4–7].

✉ Xuebin Yue
  yuexuebin@zut.edu.cn

  Ziming Wang
  gr0518ev@ed.ritsumei.ac.jp

  Ryuto Ishibashi
  ri0097fx@ed.ritsumei.ac.jp

  Hayata Kaneko
  ri0099xx@ed.ritsumei.ac.jp

  Lin Meng
  menglin@fc.ritsumei.ac.jp

1   School of Automation and Electrical Engineering, Zhongyuan University of Technology, Zhengzhou 450007, Henan, China

2   Research Organization of Science and Technology, Ritsumeikan University, Noji-higashi, Kusatsu, Shiga 525-8577, Japan

3   College of Science and Engineering, Ritsumeikan University, Noji-higashi, Kusatsu, Shiga 525-8577, Japan

Due to being buried in the ground for long periods and the decomposition of saprophytic organisms, OBPs have been worn and eroded seriously. Therefore, relatively little OBIs information is kept intact, and only a few experts with a high degree of expertise evaluate and understand them correctly.

As an outstanding researcher of Chinese culture in the second half of the 20th century, Prof. Shirakawa systematically studies the original meaning of Chinese characters by studying ancient characters and has been highly appraised by academic circles at home and abroad. He combined his research experience and insights into ancient Chinese characters to leave precious hand-notated OBIs documents. Those hand-notated OBIs documents record many OBPs, as shown in Fig. 1. Although these hand-notated documents reproduce OBIs, these manuscripts have been in the showcase and have never been organized systematically. This paper aims to design an unsupervised automatic organization method to organize Prof. Shirakawa's OBIs, construct a handwritten OBIs data set for neural network learning, and contribute to cultural heritage inheritance.

In the early stage of OBIs research, the identification and classification of OBIs are primarily realized through the collaborative research of experts and scholars with relevant knowledge, which consumes a lot of manpower and material resources. With the development and progress of computer technology, researchers have realized the identification and organization of a part of OBIs with regular shapes by using some simple image processing techniques to identify line features [4], stroke-segments-vector and stroke elements [8], etc. However, image processing recognition methods are limited in practical application due to the high variability of OBIs hindering feature definition and the weakening of feature extraction. Recently, artificial intelligence (AI) technologies have developed rapidly and exerted excellent performance in agriculture [9], medicine [10], catering service industry [11, 12], ecological protection [13], and other fields. Researchers have also gradually applied AI to OBIs recognition and have achieved excellent achievements. For example, Meng et al. [4, 14] recognize OBIs based on the deep learning model, and Liu et al. [15] propose a method based on deep convolutional neural network for the recognition of incomplete OBIs. Yue et al. [16] provide a method for OBIs recognition methods based on deep learning to suppress the occurrence of overfitting effectively. However, researchers have yet to organize handwritten OBIs, and there is no handwritten dataset for the recognition of OBIs.

To solve the abovementioned problems, we propose an unsupervised automatic organization method to organize Prof. Shirakawa's OBIs and create a handwritten OBIs dataset to provide foundation support for AI-based OBIs research. In Prof. Shirakawa's hand-notated OBIs documents, much non-OBIs information has been retained clearly except for OBIs, such as some number information and
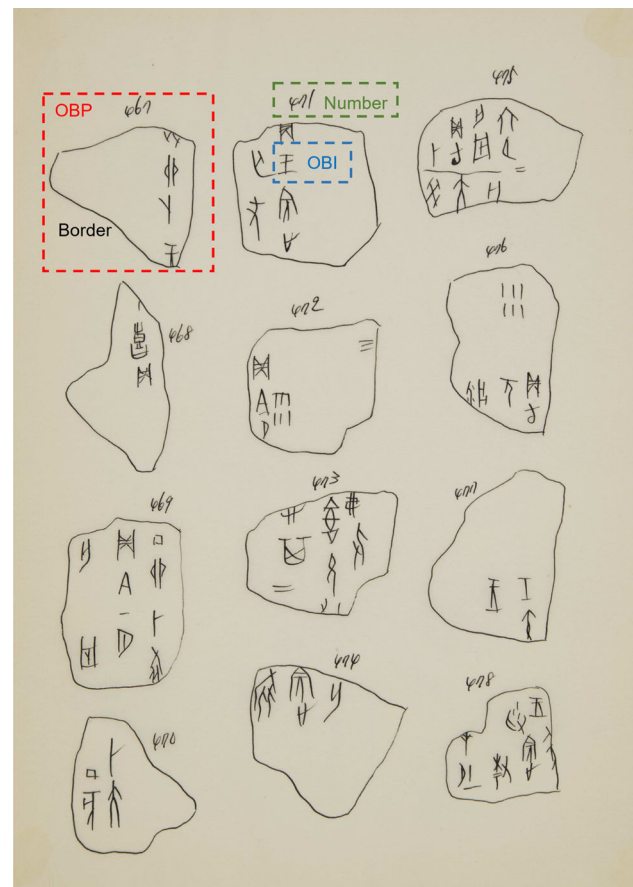


**Fig. 1** An illustration of Prof. Shirakawa's hand-notated OBIs documents

the edge of OBPs. In addition, the varying size, scattered distribution, and high degree of variation of OBIs make it challenging to organize these documents. Therefore, we propose a size-specific segmentation method based on OBPs and OBIs to segment OBPs from OBIs manuscripts. Second, we design a series of noise reduction methods to achieve the elimination of the numbers around OBPs and the removal of borders by eliminating complex interference noise.

Meanwhile, we propose a novel segmentation method combining supervised classification and feature detection on OBIs regions to achieve OBIs segmentation. Finally, we extract segmented OBIs characters based on the histogram of oriented gradients (HOG). We use dynamic $K$-means to realize unsupervised clustering of unlabeled OBIs, constructing a dataset suitable for neural network learning. The four significant contributions of this paper are as follows:

- We realize the automatic organization of Prof. Shirakawa's hand-notated OBIs documents and make his research manuscripts on OBIs publicly available for the first time.

- We propose a novel noise reduction method to remove extremely complex noises while extracting POR (Purer Oracle Regions).
- A novel segmentation method and the original supervised classification model are proposed to realize OBIs segmentation by losing less OBIs data.
- A unique method of unsupervised classification based on dynamic $K$-means is designed to dynamically cluster the segmented OBIs characters.

Section 2 of this paper reviews related work on OBIs. Section 3 introduces Prof. Shirakawa's hand-notated OBIs documents. Section 4 shows the flow of organization, including noise reduction, OBIs segmentation based on supervised classification, and unsupervised classification of OBIs. The evaluation results are reported in Sect. 5. A discussion and future works are provided in Sect. 6. In Sect. 7, we conclude with a summary.

## 2 Related work

### 2.1 Research on OBIs

Since the ancient Chinese characters OBIs first appeared in front of the world in 1899, scholars have been actively studying them to understand the ancient history of China and clearly understand the degree of cultural development at that time. So far, a large amount of OBIs literature has been published, including dictionaries [17, 18], rubbing images [19–21], and OBIs interpretations [22]. Among them, a precious document named Tieyun Canggui [23] is the first descriptive book about OBIs. Through these precious literature materials, many annotated datasets of character-level original images are made public in digital form, which is of great significance to our research on OBIs. At present, there are a few annotation datasets of character-level original images made known to the public for artificial intelligence analysis (examples include Oracle-20K [1], Oracle-50K [2], OBC306 [24], and HWOBC [25]).

In this paper, we sorted out Prof. Shirakawa's research documents and digitized his hand-notated OBIs documents. These hand-notated OBIs documents are the manuscripts copied by Prof. Shirakawa from the oracle bone rubbings based on understanding ancient Chinese characters, which have high historical research value. In order to better study and protect these precious OBIs manuscripts, we made them public and digitized them for the first time. Furthermore, an unsupervised automatic organization method is proposed to organize these OBIs documents and construct a data set suitable for neural network learning.

## 2.2 Noise reduction processing

In order to obtain a purer and more complete OBIs dataset, researchers perform many novel noise reduction methods. Meng et al. [26] extract the noise-reduced image by affine transformation and thinning. Meng et al. [27] apply Gaussian filtering and labeling to reduce noise. Yue et al. [5] realize the noise reduction of number information through a classifier that recognizes noise. Lyu et al. [28] propose a unique method to eliminate the noise problem in the image by detecting the image frame to save computing resources and time in the character recognition stage.

In addition, we also refer to many noise reduction methods for various image datasets for reducing the unique large-scale noise in Prof. Shirakawa's hand-notated OBIs documents. For example, Maity et al. [29] discuss noise reduction on various images using a fuzzy filter, a genetic algorithm, particle swarm optimization, artificial bee colony optimization, and neural networks. Ke et al. [30] introduce a novel model to exclude the disturbance caused by noisy objects with large quantities and distribution in the microscope images based on deep learning and handcrafted features. Lu et al. [31] propose a deep learning approach to directly estimate the speckle noise in the logarithmic domain based on the convolutional neural network to enhance the sonar imaging performance. Kaur et al. [32] provide a new technique, minimum patch-based noise reduction orthogonal matching pursuit (OMP), which reduces the noise from an image and uses an edge preservation patch that preserves the edges of the image.

### 2.3 Character segmentation

In the document organization process, character segmentation is a crucial issue. Precise character segmentation not only reduces noise interference but also helps to extract character information. Lyu et al. [28] propose a method for bounding box detection inside the text part of the frame for early Japanese documents, which eliminates the interference of noise outside the frame and avoids the recognition of text outside the frame. Yue et al. [5] propose a morphology-based segmentation method to segment OBIs in documents and then use a lightweight neural network to remove noise from incorrectly segmented characters. Jain et al. [33] propose a segmentation method for document images based on a multi-channel filtering approach to texture segmentation. Yue et al. [10] propose a three-level extraction technology to improve the accuracy of medicine package character segmentation.

### 2.4 Characters recognition

Neural networks have achieved significant success in large-scale image recognition in recent years. Krizhevsky et al. [34] apply the basic principles of convolutional neural networks

(CNNs) to more profound and broader networks, making CNNs widely used in computer vision. Currently, various state-of-the-art models have been proposed, such as VGG [35], ResNet [36], Inception [37], Xception [38], DenseNet [39], InceptionResNetV2 [40], and the lightweight model MobileNet [41, 42].

Researchers are trying to identify OBIs through image processing and deep learning. For example, Meng et al. [4] enhance OBIs images by rotation, Gaussian noise addition, cutting, brightness change, and flipping, then use the AlexNet network for recognition. Meng et al. [43] equip and extend a single-shot multi-box detector for OBIs detection and analyze false detections to achieve better accuracy. Yue et al. [10] use various image processing techniques and light neural networks to realize the recognition and organization of Japanese kanji on medicine packages. Meng et al. [26] propose a recognition method to recognize OBIs as template images from the OBIs dataset using the line features of inscriptions. Wang et al. [44] propose the STSN framework, which realizes structure-texture separation, disentanglement, transformation, adaptation, and recognition. By generating model decomposition features, align handprinted and scanned data to avoid the negative impact of noise. At the same time, cross-domain exchange learning implements the transformation and trains the classifier to predict scanned character labels. Wang et al. [45] propose an unsupervised domain adaptation method for Oracle Character Recognition (OrCR), which improves the model's robustness to scanned samples through pseudo-labeling and enhances consistency while introducing unsupervised transfer loss to learn more discriminative features.

## 2.5 Image classification

The current image classification is mainly for the classification of labeled images. However, the segmented characters are not labeled automatically in organizing documents, so the clustering algorithm is mainly used to classify unlabeled images. Zhang et al. [46] use a convolutional neural network to map character images in an Euclidean space, where the distance between different samples can measure their similarity, which can be performed to classify by the nearest neighbor rule. Lin et al. [47] proposes a wood color classification method with the support of cloud and edge computing and divides the feature vector set into different clusters through the $K$-means algorithm to achieve the purpose of color classification. Yue et al. [5] extract a histogram of oriented gradient descriptors as features and then cluster the oracle characters by a dynamic $K$-means method.

# 3 Prof. Shirakawa's hand-notated OBIs documents

We organize Prof. Shirakawa's research notes containing handwritten OBIs documents. Unfortunately, these documents are still displayed in the showcase and are unknown worldwide. In order to protect and use these extremely precious OBIs files more effectively, we have digitally preserved them, which significantly facilitates the researcher's use of this dataset. In this article, we officially introduce this dataset to the world.

Prof. Shirakawa's hand-notated OBIs documents consist of 79 images. For convenience, we call each image in this data set a trace. These traces are formed by copying oracle bone rubbings. Figure 1 shows an example of a trace, which contains 12 OBPs and corresponding OBP number information. Each piece of OBP information consists of an irregularly shaped closed-loop edge and multiple OBIs within the edge. The number information of the OBP is composed of several numbers, which are distributed around the OBP according to the shape of the OBP's edge. Therefore, the trace data has the characteristics of a complete feature outline, no snowflake noise, and more coherent and clear stroke features. Because of the uniqueness and advantages of these traces, in-depth research on this data undoubtedly contributes significantly to the exploration and protection of ancient Chinese culture and history. Therefore, we intend to use computer technology to organize these trace data and construct an OBIs data set suitable for neural network learning. However, there are still several challenges to overcome when dealing with this data, such as:

- The number information in trace data is easily identified as OBIs because of its size and composition, which are similar to OBIs. Furthermore, OBIs are connected to the edges directly in traces that cannot be adequately cut out, resulting in the loss of a large amount of precious OBIs data.
- In the trace data, OBIs' varying size, and uneven distribution make performing OBIs segmentation by traditional character segmentation challenging. In addition, the strokes of some OBIs are scattered and connected with the surrounding OBIs, which leads to strokes being wrongly segmented into OBIs. Redundant strokes break the OBIs' structure.
- Due to the uniqueness of Prof. Shirakawa's hand-notated OBIs data, some organizational methods based on existing OBIs data sets are not suitable for such data. In addition, there are fewer samples in the hand-notated OBIs data, and the number of OBIs categories is still being determined, significantly increasing the organization's difficulty.
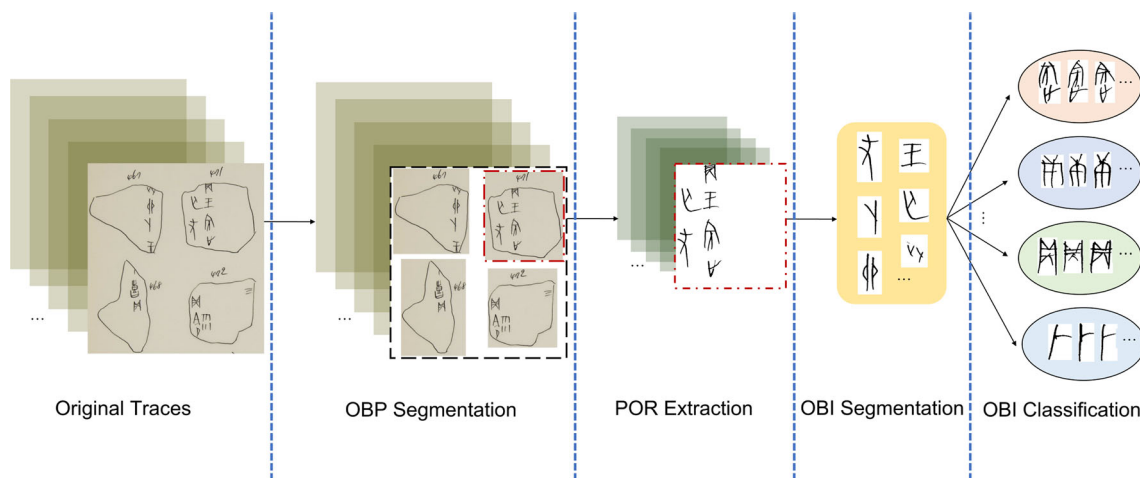
**Fig. 2** The organization flow of OBIs organization

This paper proposes various solutions to the above problems, aiming to remove interference noises, extract all the OBIs correctly, and classify them effectively.

## 4 Organization flow

Figure 2 shows the overview of the organization flows for Prof. Shirakawa's hand-notated OBIs documents, which consists of OBP segmentation, POR extraction, OBIs segmentation, and OBIs classification.

- *OBP segmentation:* A segmentation method based on size specificity between OBPs and OBIs is proposed to separate OBPs from the trace data.
- *POR extraction:* A suite of noise reduction is designed to realize POR extraction by eliminating complex interference noise, like number information and OBP edges.
- *OBIs segmentation:* A segmentation method is proposed to separate the PORs into smaller regions. In addition, a combination of supervised classification and feature detection on these OBIs regions is proposed to realize the OBIs segmentation.
- *OBIs classification:* A unique method of unsupervised classification based on dynamic $K$-means is designed to dynamically cluster the OBIs types in the trace data.

### 4.1 OBP segmentation by image processing

Due to detecting noise directly from trace data being challenging, we first separate OBP from trace to facilitate batch noise reduction on OBP to ensure the effectiveness of OBIs extraction. Figure 3 shows intermediate results of OBP segmentation processing, which consists of preprocessing, morphological processing, and feature detection.

### 4.1.1 Preprocessing

Since the trace comprises black OBIs and a lighter background, converting the trace image into grayscale [48] and implementing binarization [49] facilitate highlighting the stroke information that contains OBIs and achieves a significant reduction in computation. The grayscale and binarization results are shown in Fig. 3b, c.

- *Grayscale:* Converting the original trace image with RGB channels to a single-channel grayscale image reduces calculations and highlights features.
- *Binarization:* The OTSU [49] method is applied to obtain the threshold for binarization in this research. Equation 1 shows the OTSU threshold decision, where the proportion of foreground pixels in the whole image is $\omega_0$, the average grayscale is $\mu_0$, the proportion of background pixels in the whole image is $\omega_1$, and the average grayscale is $\mu_1$. The threshold $T$, which maximizes the variance $\mathcal{G}$ between classes, is obtained by traversal.

$$\mathcal{G} = \omega_0 \times \omega_1 \times (\mu_0 - \mu_1)^2 \qquad (1)$$

### 4.1.2 Morphological processing

As shown in Fig. 3c, the OBP forms a closed-loop area, but it is challenging to extract the OBP directly. We use the opening and erosion operations in morphological processing to expand the OBP closed-loop area and extract the closed-loop area [50].

- *Opening operation:* The opening operation utilizes a filter based on geometric operations. Some small objects, like isolated points, are removed by sequential erosion
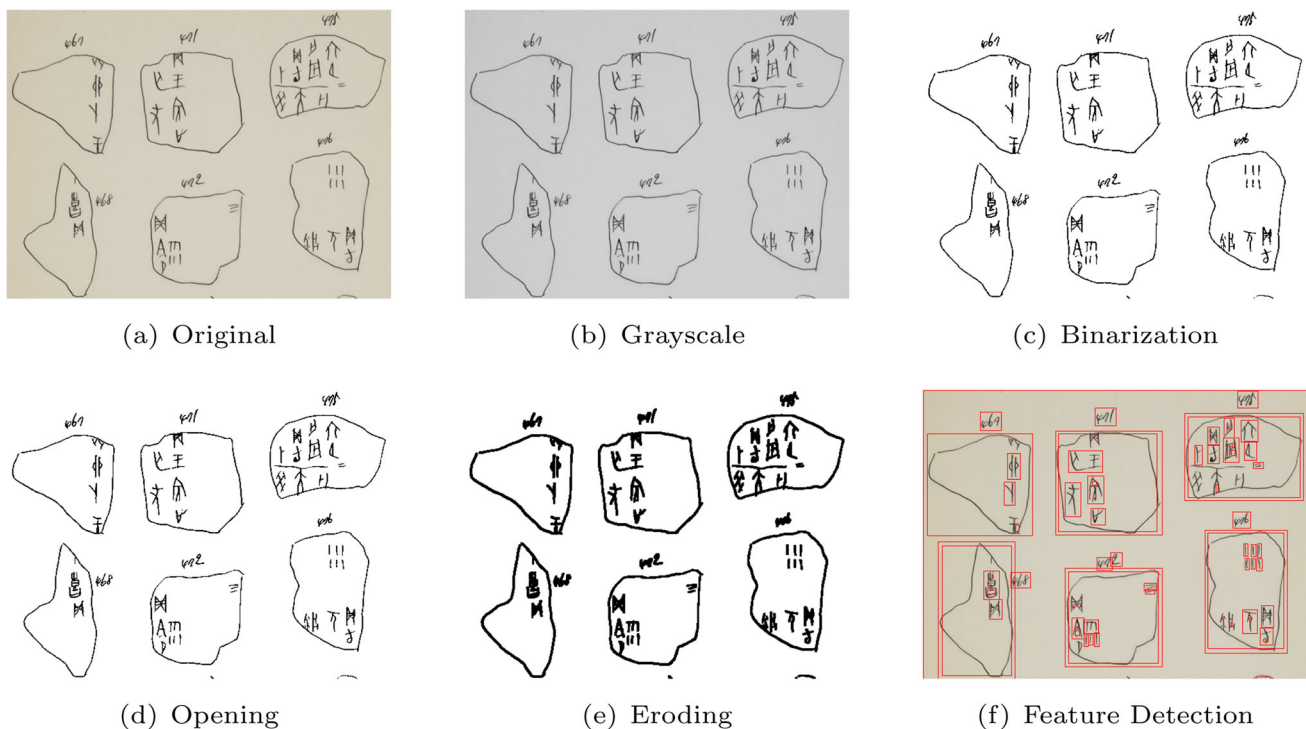
(a) Original      (b) Grayscale      (c) Binarization

(d) Opening      (e) Eroding      (f) Feature Detection

**Fig. 3** The results of the intermediate process of OBP segmentation

and expansion [51]. The result of the opening operation is shown in Fig. 3d.

- *Eroding operation:* The eroding operation erodes the object's edge. Some broken features of OBIs and edges are connected by eroding operation, and the result is shown in Fig. 3e.

### 4.1.3 Feature detection

In feature detection, the smallest and largest $x$ and $y$ axes of each stroke and edge are obtained [5]. Then each object's minor $x$-axis and $y$-axis are countered as the start point. The difference between the smallest and largest $x$-axis is calculated as the **width** of the objects, and the difference between the smallest and largest $y$-axis is calculated as the **height** of the objects. This way, the bounding box in which the object is located could be obtained. Figure 3f shows the detected objects bounded by bounding boxes.

### 4.1.4 OBP detection

The significant difference in size between OBPs and OBIs is used to realize OBP detection. After feature detection on the sample trace image, we calculate the bounding boxes' size and sort them in order, as the yellow bar is shown in Fig. 4. Furthermore, the red bar shows the difference between the size of adjacent features. As we can see, the location of the highest red bar is where the feature size changes most

drastically, and we regard the location as the demarcation point between OBIs and OBPs. The more significant features are OBPs, while the smaller ones are OBIs or some slight noise [28].

### 4.2 POR extraction by noise reduction

We propose a unique algorithm to remove noises and achieve POR extraction. Figure 5 shows intermediate noise reduction results, consisting of number information elimination and OBP edge decomposition.

#### 4.2.1 Number information elimination

As shown in Fig. 5, we preprocess the original trace data. Specifically, grayscale, smoothing, binarization, and morphological processing have been carried out on OBIs, closed loops, and integral OBP edges. In order to remove number information clearly, this paper attempts to connect the stroke features of number information into one object through morphological processing and eliminate them. However, the distance between the number information and the OBP edge is relatively close, and excessive morphological processing is likely to cause the two to be linked together so that the digital information cannot be successfully eliminated. In addition, since the positional relationship between the OBP edge and the number information is different in the trace dataset, it is impossible to connect numbered strokes in all traces by set-
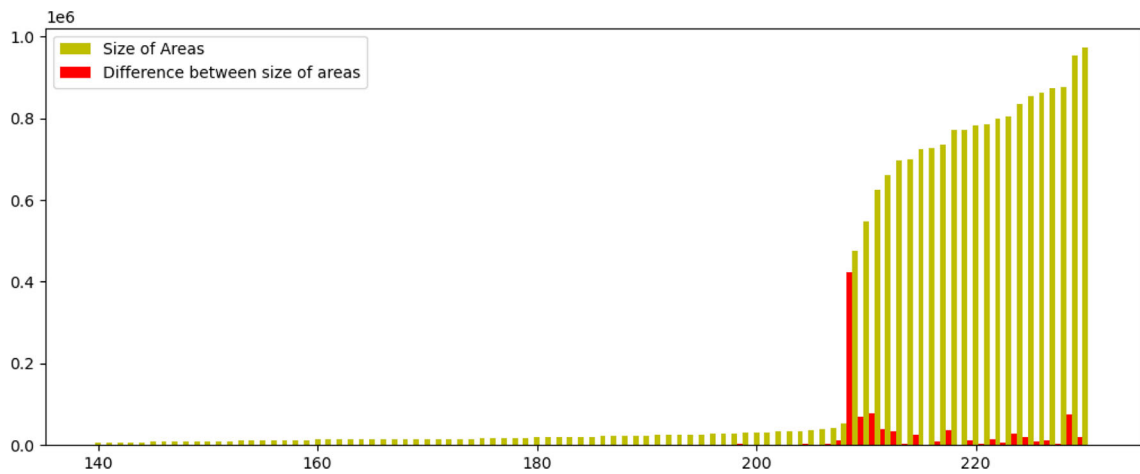
**Fig. 4** Size arrangement and variation of each feature in trace data
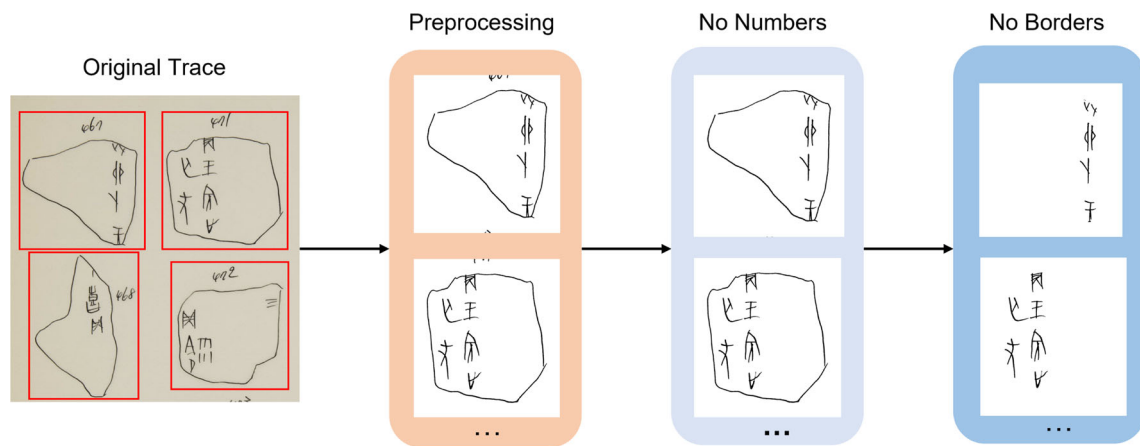


**Fig. 5** The results of the intermediate process of POR extraction

ting just one suitable morphological coefficient. Therefore, we propose a new method for image processing by three different morphological coefficients to locate and remove digital information far away from the edge and close to the edge, respectively, to remove digital information.

- *Eliminating number by three times*

  OBP performs morphological processing with morphological coefficients of 30, 20, and 10 in turn and locates and deletes the number information three times. For the first time, morphological processing is applied for an OBP image by a more significant morphological coefficient 30. The purpose is to combine the outer number features and remove the outermost feature if it is considered number information. Similarly, we aim to concatenate the number of features near OBP edges the second time and remove the outermost feature if judged as the number information. Considering features close to the edge easily adhesion to the edge through morphological processing, a smaller value of 20 is used as the morphological coefficient for the second process. In the third time, to ensure that the number information is completely cleared, a small morphological coefficient of 10 is used for morphological processing to detect the number of feature fragments that have not been removed in the first and second times, then delete them by removing the outermost feature of number information.

- *Algorithm of eliminating number information*

  The noise reduction process for eliminating number information is shown in Algorithm 1. Initially, we scan each line of the image from two horizontal directions until the first black pixel is scanned, and record the scanning distance $\mathcal{L}\_hi$ and $\mathcal{R}\_hi$ by $\mathcal{L}eft$ and $\mathcal{R}ight$, respectively. The $\mathcal{L}\_hi$ represents the number of pixels in the $i$-th row of the OBP image from the left until the first black pixel is scanned. Similarly, $\mathcal{R}\_hi$ means the number of pixels in the $i$-th row of the OBP image from the right until the first black pixel is scanned. Furthermore, since the pixel distribution of the edge and the number information is discontinuous, we find drastic changes in

$\mathcal{L}eft$ and $\mathcal{R}ight$ to identify possible positions of the number and record them in $\mathcal{P}oint$. As shown in Fig. 6, the $A$, $B$, $C$ three points shown in the image are judged to be at a huge distance from the surrounding features, and there may be numbered positions. Then, in addition to the number information, the position information recorded in the $\mathcal{P}oint$ also includes the inflection information of the edge, as shown at point $B$ in Fig. 6. Because of the size difference between number information and OBP edges, we select a parameter that is smaller than the edge and more significant than the number information as the $S\_thresh$ to filter the position of number information accurately, like points $A$ and $C$. Finally, the detected number information is turned white to complete the noise reduction operation once. We remove all numbering information by performing operations like the above three times.
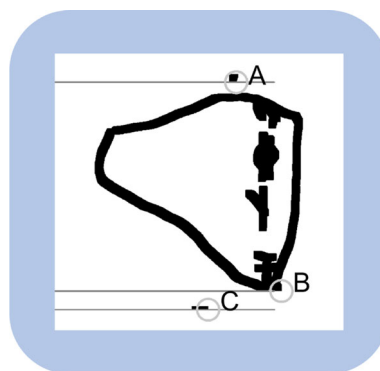
---

**Algorithm 1:** Elimination of OBP Numbers

**input** : OBPs images with number noise: $\mathcal{O}BP\_Ori$.
**Threshold of Areas between OBI and OBPs:** $S\_thresh$.
**output**: OBPs Dataset without number noise $\mathcal{O}BP\_f$.

1 **for** $\mathcal{I}ter \leftarrow 30$ **to** $10$ *step:* $10$ **do**
2     $\mathcal{O}BP\_morph$ = Morphological processing on $\mathcal{O}BP\_Ori$ by iterations $\mathcal{I}ter$;
3     $\mathcal{H}eigth$ = Length of $\mathcal{O}BP\_Ori$;
4     **for** $\langle i \leftarrow 1$ **to** $\mathcal{H}eight$ **do**
5        $\mathcal{L}\_hi \leftarrow Distance$ between the feature and the left border in row hi;
6        $\mathcal{R}\_hi \leftarrow Distance$ between the feature and the right border in row hi;
7        $\mathcal{L}eft$ add $\mathcal{L}\_hi$;
8        $\mathcal{R}ight$ add $\mathcal{R}\_hi$;
9     **end**
10     $\mathcal{P}oints \leftarrow \mathcal{P}ositon$ of the adjacent elements in $R\_hi$ and $L\_hi$ that changes drastically;
11     **for** $\mathcal{P}oint \leftarrow \mathcal{P}oints[0]$ **to** $\mathcal{P}oints[-1]$ **do**
12        $\mathcal{A}rea \leftarrow The$ area of contour where $\mathcal{P}oint$ exist;
13        **if** $Area < S\_thresh$ **then**
14           Return the contour to white;
15        **else**
16           The contour is identified as the edge of the OBP;
17        **end**
18     **end**
19 **end**

---

### 4.2.2 OBP edge decomposition

In this part, edges are eliminated in four directions by decomposing edges into tiny fragments from outside to inside. Specifically, we scan the OBIs trace images at a specific small interval $d$ from two horizontal and two vertical directions, respectively, and convert the next $h$ pixels into white
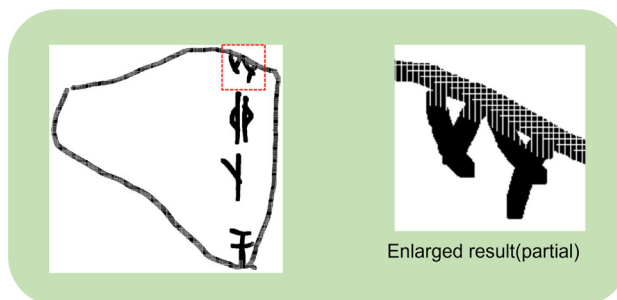


**Fig. 6** Locate number information



**Fig. 7** The edge is broken down into tiny fragments

when the first black pixel is scanned. The variable $d$ is set to a small number, ensuring that edge features are decomposed into more petite pieces than OBIs. The variable $h$ controls the extent of decomposition so as not to destroy the OBIs. This way, these OBP edges are decomposed into countless small strips or small squares with white lines, as Fig. 7 shows. These strips and squares are smaller than $d \times h$, much smaller than the OBIs. These more petite strips and squares can be filtered by setting the threshold to achieve edge decomposition.

## 4.3 OBIs segmentation based on supervised classification

This section provides a segmentation approach based on the supervised classification to achieve OBIs segmentation without destroying the OBIs structure. The process is divided into POR segmentation by histogram thresholding, supervised classification of OBIs regions, and OBIs segmentation by feature detection.

### 4.3.1 POR segmentation by histogram thresholding

Due to the ancients' vertical writing habits, the OBIs distribution's column information is easier to identify than the row information, as shown in Fig. 8a. Therefore, we propose a histogram-based POR segmentation method and first segment the POR column by column information. Specifically,

**Fig. 8** The preliminary extraction of OBIs



(a)          (b)          (c)
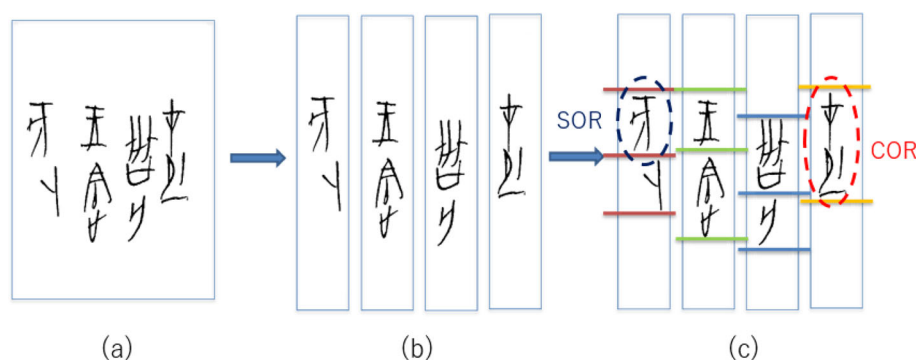
histogram thresholding is proposed to identify the columns in the image where objects do not exist and to determine whether there is a dividing line between columns or not according to the sparsity of the surrounding object distribution. The result of column segmentation is shown in Fig. 8b. After completing the vertical segmentation, the same method is used to perform horizontal segmentation on each separated image, and the result is shown in Fig. 8c. The results show that some OBIs with obvious gaps with surrounding characters are directly extracted by this method. In this study, we call those images in which only one OBIs exists in single OBIs regions (SORs). In addition, some OBIs cannot be segmented due to adhesion to surrounding characters. We call such regions where multiple OBIs co-exist complex OBIs regions (CORs). SOR is regarded as OBIs segmented successfully, and COR is regarded as a fraction of the POR. Finally, the OBIs segmentation is completed by using feature detection on CORs. A classifier capable of recognizing them is essential to realize the COR response processing on SORs and CORs. In the next step, we aim to implement this automatic classification between CORs and SORs.

### 4.3.2 Binary classification of OBIs regions

In this part, based on the character features and classification benchmarks in SORs and CORs, a unique image preprocessing technique is proposed and combined with a neural network to achieve more accurate classification.

- *Unique preprocessing method Blocking for classification*
  The number of OBIs is regarded as a classification benchmark in this classification. The image with only one OBIs is considered SOR, and the image with multiple OBIs is considered COR. Based on this classification benchmark, the different glyph structures in the images that are to be significantly classified interfere with the classification effect. This study paints the bounding boxes where all the stroke features are located in black to achieve the effect of weakening the stroke features to lessen the interference

of these redundant stroke features and increase the number of OBIs features. As shown on the left side of Fig. 9, the left side shows four original images of SOR and COR, respectively, and the right side shows the result after preprocessing. The unique image preprocessing technique is called *Blocking* in this paper. It is found that the preprocessed image has no stroke features at all, and only the approximate distribution of objects is known from it. Through this method, the stroke characteristics of different OBIs are successfully weakened, and the layout characteristics of OBIs are strengthened, which makes it easier to recognize the number of OBIs.

- *Binary classification based on LightNet*
  Fig. 9 shows the preprocessed SORs and CORs; the processed structure is simpler. Therefore, for the binary classification of such simple structures, this paper proposes a light neural network model called **LightNet** for classifying SORs and CORs, and the structure is shown in Table 1. The input image size is set to $128 \times 128 \times 3$. The first layer (input layer) has four nodes, the second layer has eight, and the third has 16. The fourth and fifth layers are fully connected, with the nodes set to 64 and 32, respectively. In the convolutional layers, we incorporate L1 regularization. Following this, to collectively prevent overfitting, we have implemented Dropout operations after the first two fully connected layers. The "SELU" is the activation function. Finally, the last layer (output layer) has two outputs and uses the "Softmax" activation function. This paper manually creates a training dataset containing 260 OBIs region images that have been decomposed from PORs [52]. There are 160 SORs and 100 CORs. 70% of the dataset's images are used for training and 30% for validation. LightNet effectively captures and learns the critical features of preprocessed SORs and CORs due to its highly simplified structure. Concurrently, LightNet emphasizes critical discriminative patterns, overcoming overfitting and attaining enhanced generalization performance within a reduced parameter space. In this way, we finally get a better classification result with an accuracy of 84.98%.
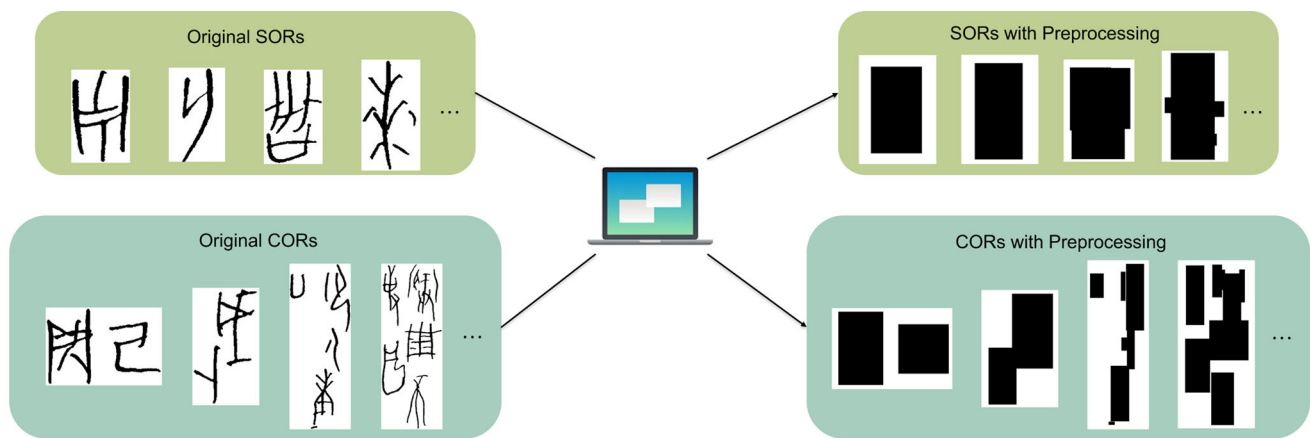
**Fig. 9** SORs and CORs before and after preprocessing

**Table 1** Overall architecture of LightNet

| Operation | Channel | Filter size | Activation |
| --- | --- | --- | --- |
| Conv2D | 4 | $5 \times 5$ | SELU |
| Max pooling | | $2 \times 2$ | |
| Conv2D | 8 | $5 \times 5$ | SELU |
| Max pooling | | $2 \times 2$ | |
| Conv2D | 16 | $5 \times 5$ | SELU |
| Max pooling | | $2 \times 2$ | |
| Flatten | | | |
| Dense | 64 | | SELU |
| Dropout, 0.5 | | | |
| Dense | 32 | | SELU |
| Dropout, 0.5 | | | |
| Dense | 2 | | Softmax |

### 4.3.3 OBIs segmentation by feature detection

This part is devoted to extracting multiple OBIs from CORs through image preprocessing and feature detection. Specifically, after preprocessing, such as grayscale transformation and binarization, morphological processing is performed to connect the strokes of the same character as much as possible. Then, feature detection is applied to get the coordinates of the bounding box where the character is located. Extracting these bounding boxes helps to realize OBIs segmentation on CORs. It is worth noting that batch morphological processing cannot achieve accurate OBIs segmentation for every COR, and the mutual adhesion of some characters causes OBIs to be destroyed and cannot be extracted correctly. CORs, on the other hand, have a smaller area and fewer characters than complete PORs, which considerably decreases the potential of mutual interference between characters and suppresses data loss as much as possible.

## 4.4 OBIs clustering based on unsupervised classification

In the above study, thousands of OBIs are successfully cropped from Prof. Shirakawa's hand-notated OBIs documents. In the following sections, we could focus on designing a method of unsupervised classification called dynamic $K$-means to realize the OBIs classification. Specifically, a novel method is proposed to find the optimal clustering value $K$ dynamically [5]. In this study, the HOG features [53] of OBI images are used to evaluate the $K$ dynamically, and then find the best $K$ and perform $K$-means [54] clustering to achieve the unsupervised classification of OBIs.

### 4.4.1 HOG features extraction

The Histogram of oriented gradient (HOG) is a feature descriptor for object detection in computer vision and image processing. It composes features by calculating the gradient direction histogram of the local area. In an image, the appearance and shape of the local target are accurately described by the directional density distribution of the gradient or edge. Hence, the HOG contains the appearance and shape information of the local target. This paper uses Eqs. 2–5 to extract the HOG features of OBIs. $G_x(x, y)$ and $G_y(x, y)$ represent the horizontal and vertical gradients. $H(x + 1, y)$ is the pixel value of the adjacent pixel to the right of the pixel $(x, y)$. Similarly, $H(x - 1, y)$, $H(x, y + 1)$, $H(x, y - 1)$ are the pixel values of the adjacent pixels on the left, lower and upper sides of the pixel $(x, y)$, respectively. Next, the two directional gradient values are used to calculate the total gradient strength value $G(x, y)$ and the gradient direction $\alpha(x, y)$. Then, we fuse these features into a two-dimensional array and dynamically find the value of OBIs category $K$ through the feature array [5].

$$G_x(x, y) = H(x + 1, y) - H(x - 1, y) \tag{2}$$

$$G_y(x, y) = H(x, y + 1) - H(x, y - 1) \tag{3}$$

$$G(x, y) = \sqrt{G_x(x, y) + G_y(x, y)} \tag{4}$$

$$\alpha(x, y) = tan^{-1}\left(\frac{G_y(x, y)}{G_x(x, y)}\right) \tag{5}$$

#### 4.4.2 OBIs categories calculation by dynamic *K*-means

This study adopts an unsupervised learning clustering method based on $K$-means to realize OBIs classification [55]. The goal of clustering is to minimize the sum of squares of various total distances. Euclidean distance is regarded as the criterion for similarity and distance judgment, and the sum of squares of errors (SSE) [56] as the objective function for clustering is shown in Eq. 6. The HOG features of OBIs are used to achieve clustering. $K$ is the number of clusters, $C_i$ is the center point of the $i$ cluster, and $dist(x, C_i)$ is the distance from $x$ to $C_i$. The variable $K$ needs to be set manually, but the number of OBIs categories $K$ is unknown, and it is challenging to find the $K$ even if it takes much manpower [5].

$$\text{SSE} = \sum_{i=1}^{k} \sum_{x \in C_i} dist(x, C_i)^2 \tag{6}$$

$$S(i) = \begin{cases} 1 - \frac{a(i)}{b(i)} & (a(i) < b(i)) \\ 0 & (a(i) = b(i)) \\ \frac{b(i)}{a(i)} - 1 & (a(i) > b(i)) \end{cases} \tag{7}$$

This paper presents an unsupervised approach termed dynamic $K$-means to accomplish OBIs classification to overcome the calculation of unknown $K$. Specifically, the Silhouette Coefficient method [57] is proposed to evaluate the results of clustering dynamically, which uses similarity measures between objects in a dataset to evaluate the quality of clustering. The idea is to compare the similarity of an object in a cluster with other objects in the same cluster and other clusters. The number of clusters with the highest Silhouette Coefficient represents the best choice for the number of clusters. Equation 7 depicts the Silhouette Coefficient, where $S(i)$ is the Silhouette Coefficient, $a(i)$ is the average distance from sample $i$ to other samples in the same cluster, and $b(i)$ is the average distance from sample $i$ to all samples in other clusters. The lower $a(i)$, the more the sample $i$ belongs to this cluster, and the greater $b(i)$, the less the sample $i$ belongs to other clusters. When $S(i)$ approaches 1, it indicates that the clustering of a sample $i$ is reasonable. When $S(i)$ approaches -1, it indicates that sample $i$ should be classified into other clusters, and when $S(i)$ equals 0, it indicates that sample $i$ is on the boundary of two clusters. All $K$ instances from 2 to the maximum number are calculated, and the best one (closest to 1) is chosen as the final result.
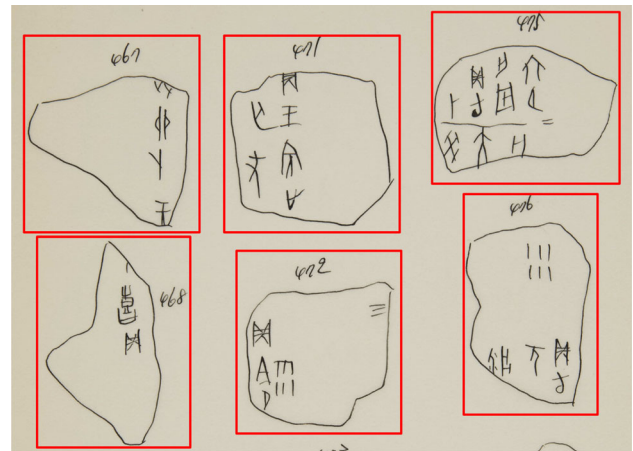


**Fig. 10** The result of OBPs segmentation

## 5 Evaluation

Noise reduction on OBIs traces, OBIs segmentation, and OBIs classification are the three critical components of this study. Uniquely, the stepwise segmentation of OBIs has been realized by the classification of SOR and COR in this paper. This section confirms the effectiveness of a series of methods proposed in this paper by evaluating the effect of noise reduction on various types of noise, the classification accuracy of SOR and COR, and the accuracy of OBIs clustering.

### 5.1 Noise reduction on OBI traces

- *OBP segmentation* For Prof. Shirakawa's hand-notated OBIs documents, 364 OBPs are successfully separated from 79 original traces by the size specificity of OBPs. The partial separation result of one of the trace images is shown in Fig. 10.
- *Noise reduction on number information* A series of processes are performed on the OBP to filter the number of information successfully. Out of 364 OBPs, number information is completely removed in 356 OBPs by the discreteness between number information and edges. The accuracy of noise reduction on number information reaches 97.85%. The results are shown in Fig. 11, where the first column is the original OBP image, the second column is the image after a series of processing such as grayscale and binarization, and the third column is the result of noise reduction on number information.
- *Noise reduction on edges* A unique algorithm is proposed to decompose OBP edges into extremely small fragment features filtered clearly. In this paper, the removal of OBP edges and the extraction of PORs are implemented for 356 images that have achieved the removal of number information. The noise reduction result is shown in the fourth column of Fig. 11. Through these operations,

features other than OBIs have been removed with high accuracy, and the extraction of POR is realized. The number removal rate of up to 97.85% and the edge removal rate of 100% confirm the feasibility of the noise reduction method on number information and edges.

## 5.2 Classification of SORs and CORs

The PORs have been successfully segmented into the smaller OBIs regions in the vertical and horizontal directions by Histogram Thresholding in this paper to facilitate the OBIs segmentation. The intermediate results are shown in Fig. 12. The first row shows four POR images, the second row shows the result of the first round of vertical segmentation, and the third row represents the result of the second round of horizontal segmentation. It is clear that after segmentations in two directions, many OBIs (SOR) have been successfully extracted, and some images still contain multiple OBIs (COR). Further operations are required to complete the OBIs extraction. Therefore, this experiment is devoted to classifying the SOR and COR to achieve OBIs segmentation.

### 5.2.1 Comparison the classification accuracy of state-of-the-art deep learning models before and after preprocessing Blocking

To prove that the unique image preprocessing method called **Blocking** helps to improve the classification accuracy of SOR and COR, in addition to the original model LightNet proposed in this study, we also evaluate the performance of eight state-of-the-art models in deep learning on the dataset before and after **Blocking** preprocessing, which contains VGG16 [35], DenseNet121 [39], InceptionV3 [37], MobileNet [42], Xception [38], ResNet50 [36], InceptionResNetV2 [40], MobileNetV2 [41]. Equations 8-10 are used to calculate measurement indices, including *Accuracy*, *Precision*, and *Recall*. Specifically, *TP* means COR is correctly identified, *TN* means SOR is correctly identified, *FN* means COR is incorrectly predicted as SOR, and *FP* means SOR is incorrectly identified as COR. Two datasets have been created, respectively, before and after **Blocking**, and 200 images of SOR and COR in each dataset are used as the test dataset for these comparative experiments. In order to maintain the consistency of the results, the epoch of all models is set to 100, the optimizer is Adam, and the learning rate is 0.001.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{8}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{9}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{10}$$

In the middle column of Table 2, we use the original dataset **D_ori**, which is not preprocessed by **Blocking**, to show the classification results on nine models, and the right column represents the classification accuracy on preprocessed dataset **D_block** by nine models. As we can see, VGG16 and MobileNetV2 identify all the images in the test data as SOR in **D_ori** and **D_block**, and the classification accuracy of the other models show varying degrees of improvement in **D_block**, respectively. Specifically, the classification accuracy of DenseNet121 increases by 2.00%, InceptionV3 increases by 7.00%, MobileNet increases by 16.50%, ResNet50 increases by 0.50%, Xception increases by 3.50%, InceptionResNetV2 increases by 3.50%, and LightNet increases by 11.50%. These comparisons fully confirm the effectiveness of **Blocking** proposed in this paper.

### 5.2.2 Compare the classification results of the top-six model on dataset D_block

As shown in Table 2, DenseNet121, InceptionV3, ResNet50, Xception, InceptionResNetV2, and LightNet show relatively correct classification rate on the dataset **D_block**, reaching 84.50%, 85.00%, 81.50%, 85.50%, 85.00%, and 85.50%, respectively. Considering the classification accuracy of the six models above is relatively close, a single experiment cannot obtain rigorous comparison results. This subsection further studies the above six models, using a larger test data (232 COR images, 259 SOR images) in **D_block** and performing six repeated experiments on these six relatively excellent models. The results are shown in Table 3. For the experiment's rigor, each model's classification accuracy is averaged six times as the classification result of the model. The LightNet proposed in this article has the best classification performance, with an average classification accuracy of 84.98%. In addition, this subsection also ranks the total parameters of the nine models that appear in this paper. The number of total parameters in nine models from small to large is 0.15 *M* for LightNet, 2.26 *M* for MobileNetV2, 3.23 *M* for MobileNet, 7.04 *M* for DenseNet121, 20.87 *M* for Xception, 21.81 *M* for InceptionV3, 23.59 *M* for ResNet50, 54.34 *M* for InceptionResNetV2, 65.06 *M* for VGG16, as shown in Table 4. We compare six models with high classification accuracy on both average and total parameters, finding that the LighteNet model proposed in this article has better classification accuracy and fewer parameters than other network models simultaneously. The above results fully verify the effectiveness of the LightNet.

### 5.2.3 Experimental results of overfitting solution

The **D_block** dataset has the characteristics of a small amount of data, fuzzy semantics, and unclear classification features, which makes the model overfit when training the dataset.

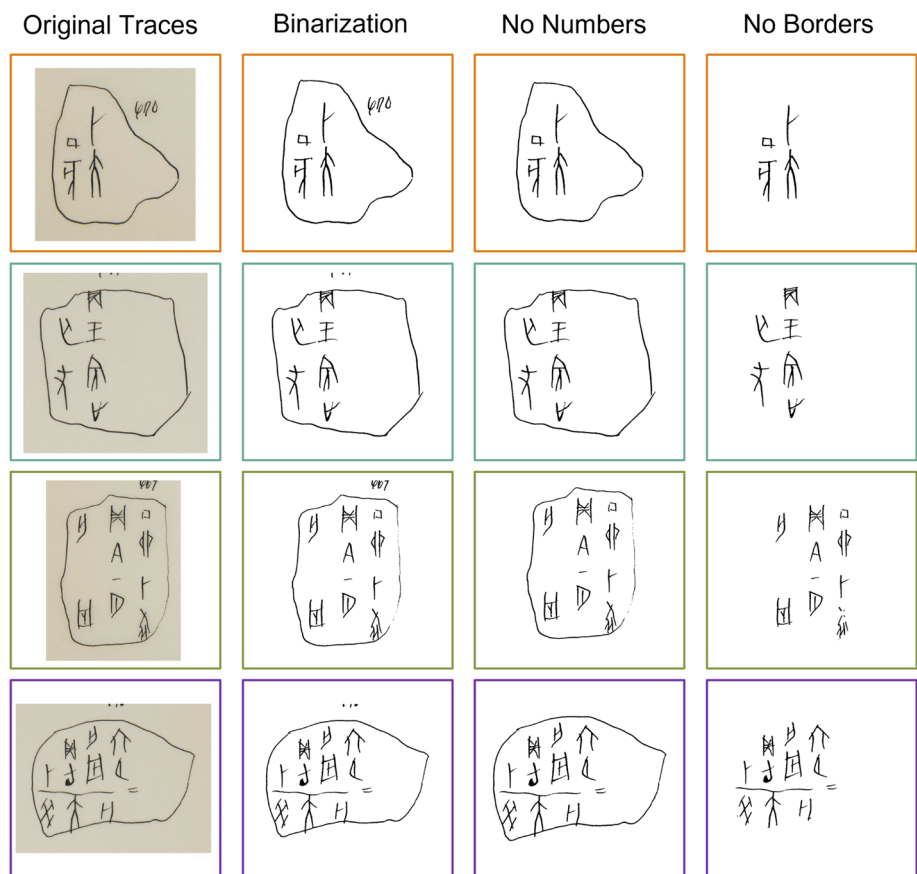**Fig. 11** The partial results of the preliminary segmentation on PORs
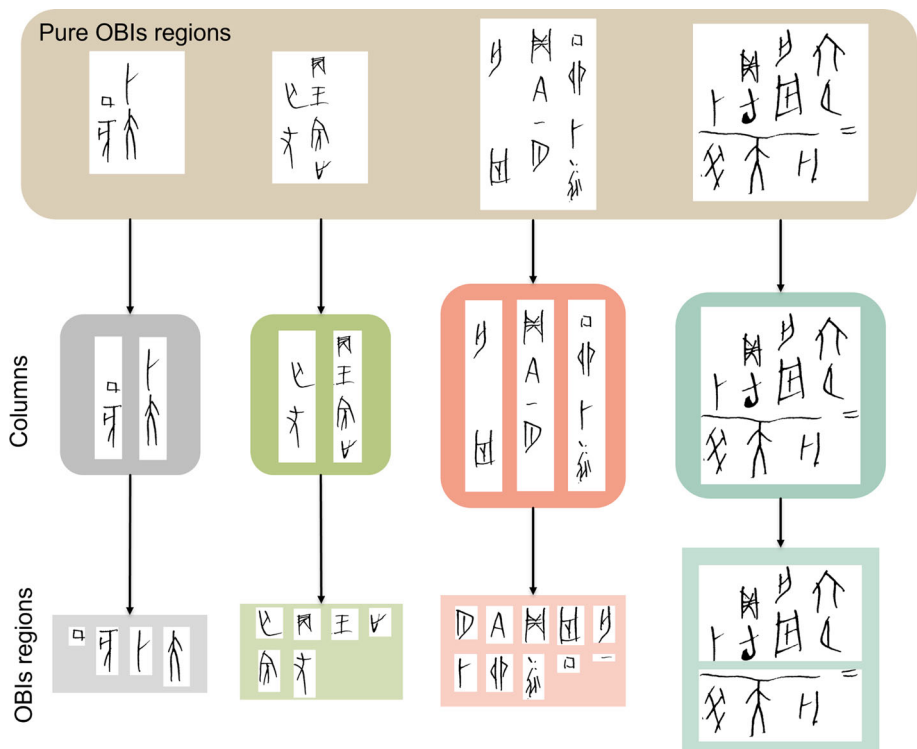


**Fig. 12** The effect of preliminary extraction on PORs

**Table 2** Experimental results of different models on **D_ori** and **D_block**

| Model | D_ori | | | | | D_block | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | TP | TN | FN | FP | Accuracy (%) | TP | TN | FN | FP | Accuracy (%) |
| VGG16 | 0 | 100 | 100 | 0 | 50.00 | 0 | 100 | 100 | 0 | 50.00 |
| DenseNet121 | 88 | 77 | 12 | 23 | **82.50** | 88 | 81 | 12 | 19 | 84.50 |
| InceptionV3 | 69 | 87 | 31 | 13 | 78.00 | 82 | 88 | 18 | 12 | 85.00 |
| MobileNet | 22 | 99 | 78 | 1 | 60.50 | 63 | 91 | 37 | 9 | 77.00 |
| ResNet50 | 68 | 94 | 32 | 6 | 81.00 | 72 | 91 | 28 | 9 | 81.50 |
| Xception | 67 | 97 | 33 | 3 | 82.00 | 84 | 87 | 16 | 13 | **85.50** |
| InceptionResNetV2 | 86 | 77 | 14 | 23 | 81.50 | 82 | 88 | 18 | 12 | 85.00 |
| MobileNetV2 | 0 | 100 | 100 | 0 | 50.00 | 0 | 100 | 100 | 0 | 50.00 |
| LightNet | 69 | 79 | 31 | 21 | 74.00 | 88 | 83 | 12 | 17 | **85.50** |

**Table 3** Re-comparison of the top-six models

| Model | Acc_1 (%) | Acc_2 (%) | Acc_3 (%) | Acc_4 (%) | Acc_5 (%) | Acc_6 (%) | Acc_ave (%) |
|---|---|---|---|---|---|---|---|
| DenseNet121 | 83.28 | 82.94 | 84.98 | 84.64 | 78.16 | 84.30 | 83.05 |
| InceptionV3 | 84.46 | 85.32 | 83.96 | 82.93 | 84.64 | 83.28 | 84.09 |
| ResNet50 | 82.25 | 82.94 | 83.68 | 80.55 | 84.30 | 79.86 | 82.26 |
| Xception | 80.20 | 82.23 | 83.96 | 83.28 | 82.25 | 83.28 | 82.53 |
| InceptionResNetV2 | 84.30 | 83.28 | 80.89 | 85.32 | 84.98 | 84.64 | 83.90 |
| LightNet | 84.98 | 84.64 | 83.28 | 86.35 | 85.67 | 84.98 | **84.98** |

**Table 4** Total parameters comparison of nine models

| Model | Light Net | Mobile NetV2 | Mobile Net | Dense Net121 | Xception Xception | Inception V3 | Res Net50 | Inception ResNetV2 | VGG16 |
|---|---|---|---|---|---|---|---|---|---|
| Total parameters | 0.15M | 2.26M | 3.23M | 7.04M | 20.87M | 21.81M | 23.59M | 54.34M | 65.06M |

Many techniques, such as dropout layers and various regularizations, have been developed to prevent overfitting. These measures have been applied to improve the model's classification performance to prevent widespread overfitting. We conduct experiments to evaluate the effectiveness of the proposed LightNet model on **D_block** from dropout layers and regularization, respectively. Table 5 summarizes the effect of adding dropout layers (different coefficients) in the model on classification accuracy by repeating the experiment six times and averaging the accuracy. We can see that better accuracy is obtained when the model has no dropout layer, which reaches 84.76%. When the dropout layer coefficients are set to 0.1, 0.2, 0.3, 0.4, 0.5, and 0.6, the average accuracy of classification is 83.50%, 83.28%, 82.48%, 83.56%, 82.76%, and 82.94%, respectively. The repeated experiment shows that the network structure of LightNet proposed in this paper is very simple. Adding a dropout layer would fail to achieve a better result and even reduce model classification accuracy. Table 6 summarizes the classification results of the proposed LightNet model when adding different combinations of these

L1 and L2 regularizations, where 1 and 0 indicate that the strategy is added or not added to the model, respectively. Analyzing these six repeated comparative experiments. It is not difficult to find that the most excellent results have been achieved by adding L2 regularization, reaching an average classification accuracy of 84.81%.

## 5.3 OBIs classification

This article proposes a method called dynamic $K$-means, which dynamically finds the optimal clustering value $K$ in an interval through the Silhouette Coefficient and completes the OBIs classification by clustering. As shown in Eqs. 11 and 12, $a_i$ represents the average distance $C_{in}$ to other points in the cluster, the point to which $x_i$ belongs. $b_i$ shows that the average distance $C_{near}$ to points belonging to another cluster is closest to $x_i$. After that, the Silhouette Coefficient $S_i$ is calculated by dividing $b_i - a_i$ by the larger $a_i, b_i$. The calculation process is shown as Eq. 13. The Silhouette Coefficient would be in the interval $[-1, 1]$. When the coefficient is closer to 1,

**Table 5** Comparison of LightNet performance by adding different dropout layer coefficients

| Coefficients | Acc_1 (%) | Acc_2 (%) | Acc_3 (%) | Acc_4 (%) | Acc_5 (%) | Acc_6 (%) | Acc_ave |
|---|---|---|---|---|---|---|---|
| **0** | 84.64 | 84.30 | 84.98 | 84.64 | 84.30 | 85.67 | **84.76** |
| 0.1 | 81.91 | 84.30 | 84.30 | 82.59 | 83.62 | 84.30 | 83.50 |
| 0.2 | 81.91 | 84.98 | 83.27 | 85.32 | 82.59 | 81.57 | 83.28 |
| 0.3 | 81.23 | 79.86 | 80.89 | 83.62 | 84.98 | 84.30 | 82.48 |
| 0.4 | 81.91 | 85.67 | 82.94 | 82.59 | 83.62 | 84.64 | 83.56 |
| 0.5 | 84.92 | 83.96 | 81.23 | 83.61 | 82.59 | 80.20 | 82.76 |
| 0.6 | 83.62 | 79.86 | 82.25 | 83.28 | 85.32 | 83.28 | 82.94 |

**Table 6** Comparison of LightNet performance by adding different regularization strategies

| NO. | L1 | L2 | Acc_1 | Acc_2 | Acc_3 | Acc_4 | Acc_5 | Acc_6 | Acc_ave |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 85.32 | 84.30 | 84.30 | 81.91 | 86.35 | 84.98 | 84.53 |
| **2** | **0** | **1** | 85.67 | 84.64 | 85.67 | 82.94 | 84.98 | 84.98 | **84.81** |
| 3 | 1 | 0 | 84.64 | 84.30 | 84.98 | 84.64 | 84.30 | 85.67 | 84.76 |
| 4 | 1 | 1 | 83.62 | 84.98 | 81.57 | 85.32 | 87.03 | 85.32 | 84.64 |

the evaluation performance is higher. If the value is negative, the cluster to which it belongs may be misclassified.

$$a_i = \frac{1}{|C_{\text{in}} - 1|} \times \sum_{x_j \in C_{\text{in}}} \|x_i - x_j\| \qquad (11)$$

$$b_i = \frac{1}{|C_{\text{near}}|} \times \sum_{x_j \in C_{\text{near}}} \|x_i - x_j\| \qquad (12)$$

$$S_i = \frac{b_i - a_i}{\max(a_i, b_i)} \qquad (13)$$

This experiment dynamically analyzes the Silhouette Coefficient between 50–100, 100–150, 150–200, and 200–250 to discover the optimal number of clusters $K$. The results are shown in Fig. 13. The optimal clustering values $K$ obtained before limiting the search range to 50–100, 100–150, 150–200, and 200–250 are $C\_90$, $C\_148$, $C\_190$ and $C\_244$ respectively.

In this paper, OBIs in trace datasets are uniformly classified, and the number of OBIs categories is large, which makes it difficult to record and evaluate the classification results of each category. Therefore, this study's evaluation metric for classification is measured by randomly selecting the categories. When $K$ is $C\_90$, $C\_148$, $C\_190$ and $C\_244$, the classification accuracy is shown in Table 7 respectively. It can be seen that when the number of classifications is $C\_244$, the classification has achieved satisfactory results, and the classification success rate reaches 74.91%. Figure 14 shows the final clustering results. It can be seen that the same OBIs are clustered together, and finally, a hand-notated OBIs data set suitable for neural network learning is constructed.

## 6 Discussion and future work

### 6.1 POR extraction by noise reduction

In the number information elimination stage, we adopt a three-level extraction method based on morphological processing. Although this method effectively removes digital information, the execution process still requires much time due to the complexity of the extraction. To improve efficiency, we plan to combine numbers and remove them using word recognition methods in future work. Similarly, in OBP edge decomposition, the proposed method needs to traverse the first black pixel of each row, which results in a long processing time. In future work, we plan to treat each edge as a closed loop and directly remove it based on the width of the edge to improve efficiency. In view of the above problems, in future work, we plan to extract and recognize the OBI characters in Shirakawa's manuscripts directly through object detection and other methods.

### 6.2 OBIs segmentation with classification of COR and SOR

During the POR segmentation process, different forms of OBIs appeared, including a single character, a single separated character, and multiple connected characters. Therefore, we must devise different methods to deal with various forms. For characters that are difficult to segment, we divide them into SORs and CORs and segment OBI characters through vertical and horizontal segmentation. The Light-Net we built achieves the classification of SOR and COR with an accuracy of 85.50%. In these OBIs regions, the SOR misidentified as COR is subjected to morphological
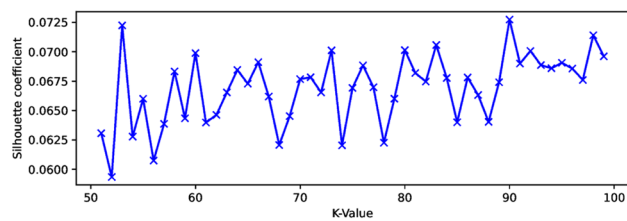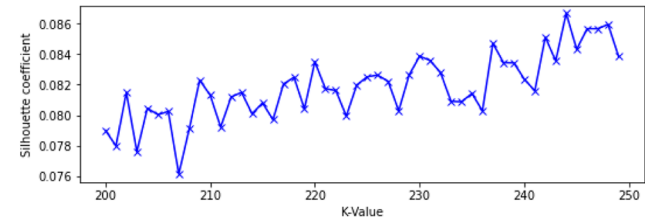
(a) K value ranges from 50 to 100.
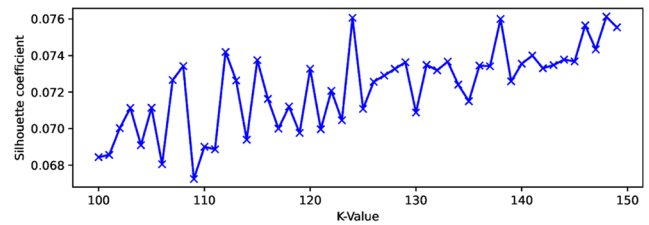


(b) K value ranges from 100 to 150.
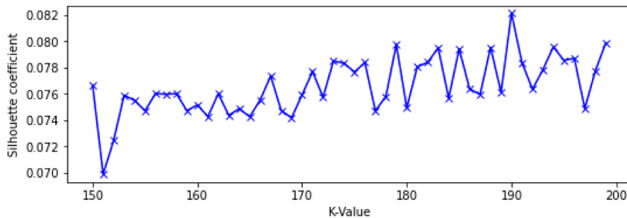


(c) K value ranges from 150 to 200.



(d) K value ranges from 200 to 250.

**Fig. 13** The result of Silhouette coefficient between $K$ value ranges from 50 to 250

**Table 7** The accuracy of classification based on Silhouette Coefficient

| Class | $\mathcal{C}\_90$ (%) | $\mathcal{C}\_148$ (%) | $\mathcal{C}\_190$ (%) | $\mathcal{C}\_244$ (%) |
|---|---|---|---|---|
| Accuracy | 62.18 | 65.07 | 55.30 | **74.91** |

processing, which destroys the complete OBIs. The COR mistakenly identified as SOR is directly saved as a complete OBIs, resulting in the loss of multiple OBIs data. In addition, the above two cases generate invalid characters, which greatly impacts classification accuracy. Although this paper proposes preprocessing **Blocking** to improve the classification accuracy of LightNet significantly, it still fails to classify all OBIs regions correctly. By observing Table 2, both VGG16 and MobileNetV2 exhibit clear signs of overfitting, and the results in the detection outcomes of these two models heavily lean toward a specific category. This suggests that VGG16 and MobileNetV2, being overly complex, struggle to generalize well to the small sample dataset we have proposed. In Table 6, we alleviate the overfitting issue of LightNet through various regularization methods. Moreover, it once again underscores the superiority of our proposed LightNet on small datasets. In future work, we are committed to proposing more methods to optimize the classification of SOR and COR for higher classification accuracy.

## 6.3 Classification of OBIs

In OBIs classification, a dynamic $K$-means is proposed to find the number of different OBIs categories in the unlabeled
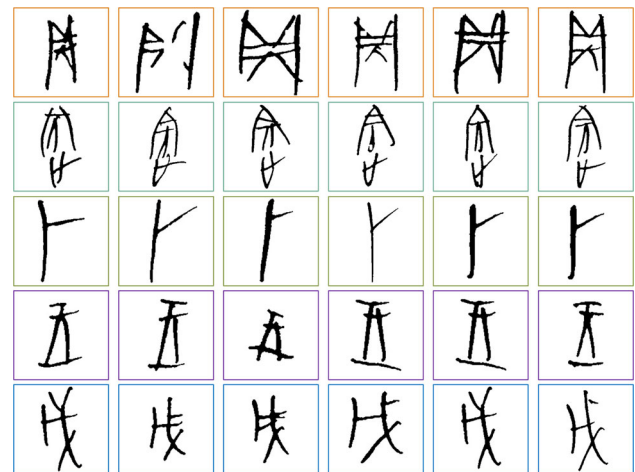


**Fig. 14** The partial results of OBIs classification

dataset and implement automatic clustering for the unsupervised classification of OBIs. The dynamic $K$-means cannot achieve high-accuracy classification because of the variety of OBIs in the trace dataset and disparities in carving among the same OBIs. We dynamically evaluated the four intervals between 50 and 250. We found the best $K$ value is 244, but it can be found that the Silhouette Coefficient continues to rise in this interval, and there is no sign of an overall decline. It shows that the $K$ value obtained now may be the local optimal $K$ rather than the global optimal $K$. Therefore, we intend to expand the evaluation interval of the dynamic $K$-means in the following experiments and explore the global optimal $K$ location to obtain higher clustering accuracy. At the same time, we plan to perform unsupervised clustering of OBIs using hierarchical clustering (a bottom-up or top-down

method that forms a hierarchical structure by gradually merging or splitting clusters).

# 7 Conclusion

In this paper, we first propose a suite of noise reduction to achieve POR extraction with an accuracy of 97.85%. Next, POR segmentation by Histogram Thresholding and supervised classification of SORs and CORs have been proposed to realize OBIs segmentation. In particular, a unique preprocessing **Blocking** has been proposed to improve the accuracy of classification with 11.50% in LightNet proposed in this paper. As a result, LightNet has achieved an accuracy of 85.50%, which performs better than eight other state-of-the-art deep learning models on **D_block**. Finally, dynamic $K$-means is proposed to dynamically detect the optimal $K$ that represents the number of OBIs categories in the trace dataset and is combined with clustering to complete the supervised classification of OBIs. As a result, the best $K$ is predicted to be 244, and the classification accuracy reaches 74.91%. The results show that the proposed method effectively organizes Professor Shirakawa's handwritten OBI documents and constructs a data set for neural network learning. This study not only helps to understand Prof. Shirakawa's research documents but also helps to unravel the ancient history of China through the understanding of OBIs, which is of great significance for accelerating the research in the field of OBIs and promoting their digitization and intelligence. In future work, we plan to focus on improving the accuracy of character segmentation and classification in Prof. Shirakawa's hand-notated OBIs documents. On the other hand, we devote ourselves to researching the small sample data of OBIs and contribute to the research related to the classification and recognition of OBIs.

## Declarations

**Conflict of interest** The authors declare no competing interests.

## References

1. Guo, J., Wang, C., Roman-Rangel, E., Chao, H., Rui, Y.: Building hierarchical representations for oracle character and sketch recognition. IEEE Trans. Image Process. **25**(1), 104–118 (2016). https://doi.org/10.1109/TIP.2015.2500019

2. Han, W., Ren, X., Lin, H., Fu, Y., Xue, X.: Self-supervised learning of Orc-Bert augmentator for recognizing few-shot oracle characters. In: Proceedings of the Asian Conference on Computer Vision (ACCV) (2020)

3. Fujikawa, Y., Li, H., Yue, X., Aravinda, C.V., Prabhu, G.A., Meng, L.: Recognition of oracle bone inscriptions by using two deep learning models. In: CoRR (2021). arXiv:2105.00777

4. Meng, L., Kamitoku, N., Yamazaki, K.: Recognition of oracle bone inscriptions using deep learning based on data augmentation. In: 2018 Metrology for Archaeology and Cultural Heritage (MetroArchaeo), pp. 33–38 (2018). https://doi.org/10.1109/MetroArchaeo43810.2018.9089769

5. Yue, X., Lyu, B., Li, H., Fujikawa, Y., Meng, L.: Deep learning and image processing combined organization of Shirakawa's hand-notated documents on OBI research. In: 2021 IEEE International Conference on Networking, Sensing and Control (ICNSC), vol. 1, pp. 1–6 (2021). https://doi.org/10.1109/ICNSC52481.2021.9702164

6. Tian, Y., Gao, W., Liu, X., Chen, S., Mo, B.: The research on rejoining of the oracle bone rubbings based on curve matching. ACM Trans. Asian Low Resour. Lang. Inf. Process. (2021). https://doi.org/10.1145/3460393

7. Lin, X., Chen, S., Zhao, F., Qiu, X.: Radical-based extract and recognition networks for Oracle character recognition. Int. J. Doc. Anal. Recognit. **25**, 219–235 (2022)

8. Li, Q., Yang, Y.: A human–computer interactive dynamic description method for Jiaguwen characters. Proc. Eng. (2012). https://doi.org/10.1016/j.proeng.2012.01.081

9. Kamilaris, A., Prenafeta-Boldú, F.X.: Deep learning in agriculture: a survey. Comput. Electron. Agric. **147**, 70–90 (2018)

10. Yue, X., Lyu, B., Li, H., et al.: Real-time medicine packet recognition system in dispensing medicines for the elderly. Meas. Sens. **18**, 100072 (2021). https://doi.org/10.1016/j.measen.2021.100072

11. Yue, X., Li, H., Shimizu, M., et al.: YOLO-GD: a deep learning-based object detection algorithm for empty-dish recycling robots. Machines **10**(5), 294 (2022)

12. Yue, X., Li, H., Meng, L.: An ultralightweight object detection network for empty-dish recycling robots. IEEE Trans. Instrum. Meas. **72**, 1–12 (2023). https://doi.org/10.1109/TIM.2023.3241078

13. Meng, L., Hirayama, T., Oyanagi, S.: Underwater-drone with panoramic camera for automatic fish recognition based on deep learning. IEEE Access **6**, 17880–17886 (2018). https://doi.org/10.1109/ACCESS.2018.2820326

14. Liu, G., Xing, J., Xiong, J.: Spatial pyramid block for oracle bone inscription detection. In: Proceedings of the 2020 9th international conference on software and computer applications. ICSCA 2020, pp. 133–140. Association for Computing Machinery, New York (2020). https://doi.org/10.1145/3384544.3384561

15. Liu, M., Liu, G., Liu, Y., Jiao, Q.: Oracle bone inscriptions recognition based on deep convolutional neural network. J. Image Graph. (2020). https://doi.org/10.18178/joig.8.4.114-119

16. Yue, X., Li, H., Fujikawa, Y., Meng, L.: Dynamic dataset augmentation for deep learning-based oracle bone inscriptions recognition. J. Comput. Cult. Herit. (2022). https://doi.org/10.1145/3532868

17. Xu, Z.: Jiaguwen Zidian (Dictionary of Oracle Bone Inscriptions). Sichuan Lexicographical Publishing House, Chengdu (1989)

18. Zhang, J., Guo, X.: Yinxu Jiaguxue Dacidian (Dictionary of Oracle Bone Studies in Yin Ruins). China Social Sciences Press, Beijing (2021)

19. Pu, M.: Oracle Bone Inscriptions in the Collection of Shanghai Museum, vol. I, 1st edn. Shanghai Lexicographical Publishing House, Shanghai (2009)

20. Guo, M.: Jiaguwen Heji (The Great Collection of the Oracle Bone Inscriptions). Zhonghua Book Company, Beijing (1978)

21. Social Science, I.: Xiaotun Nandi (Oracle Bone Inscriptions from Nandi, Xiaotun). Zhonghua Book Company, Beijing (1983)

22. Yu, S.: Jiaguwenzi Gulin (Collected Commentaries on the Oracle-Bone Characters). Zhonghua Book Company, Beijing (1996)

23. Liu, E.: Tieyun Canggui. Baoshoucanque Zhai Lithographic Publishing, Beijing (1903)

24. Huang, S., Wang, H., Liu, Y., Shi, X., Jin, L.: OBC306: a large-scale oracle bone character recognition dataset. In: 2019 International Conference on Document Analysis and Recognition (ICDAR), pp. 681–688 (2019). https://doi.org/10.1109/ICDAR.2019.00114

25. Li, B., Dai, Q., Gao, F., Zhu, W., Li, Q., Liu, Y.: HWOBC-A handwriting oracle bone character recognition database. J. Phys. Conf. Ser. **1651**, 012050 (2020). https://doi.org/10.1088/1742-6596/1651/1/012050

26. Meng, L.: Two-stage recognition for oracle bone inscriptions. In: Image Analysis and Processing-ICIAP 2017, pp. 672–682. Springer (2017)

27. Meng, L.: Recognition of oracle bone inscriptions by extracting line features on image processing. In: ICPRAM, pp. 606–611 (2017)

28. Lyu, B., Li, H., Tanaka, A., Meng, L.: The early Japanese books reorganization by combining image processing and deep learning. CAAI Trans. Intell. Technol. **7**, 627–643 (2022)

29. Maity, A., Pattanaik, A., Sagnika, S., Pani, S.: A comparative study on approaches to speckle noise reduction in images. In: 2015 International Conference on Computational Intelligence and Networks, pp. 148–155 (2015). https://doi.org/10.1109/CINE.2015.36

30. Ke, J., Deng, J., Lu, Y.: Noise reduction with image inpainting: an application in clinical data diagnosis. In: ACM SIGGRAPH 2019 Posters. SIGGRAPH '19. Association for Computing Machinery, New York (2019). https://doi.org/10.1145/3306214.3338593

31. Lu, Y., Liu, R.W., Chen, F., Xie, L.: Learning a deep convolutional network for speckle noise reduction in underwater sonar images. ICMLC '19, pp. 445–450. Association for Computing Machinery, New York (2019). https://doi.org/10.1145/3318299.3318358

32. Kaur, R., Kaur, K.: Remote sensing noise reduction using minimum patch based on OMP. In: Proceedings of the 8th International Conference on Signal Processing Systems. ICSPS 2016, pp. 83–88. Association for Computing Machinery, New York (2016). https://doi.org/10.1145/3015166.3015168

33. Jain, A.K., Sushil, B.: Text segmentation using Gabor filters for automatic document processing. Mach. Vis. Appl. **5**(3), 169–184 (1992)

34. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Commun. ACM **60**(6), 84–90 (2017)

35. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition (2014). arXiv:1409.1556

36. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)

37. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)

38. Chollet, F.: Xception: deep learning with depthwise separable convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)

39. Huang, G., Liu, Z., Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)

40. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.: Inception-v4, inception-ResNet and the impact of residual connections on learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31(1) (2017). https://doi.org/10.1609/aaai.v31i1.11231

41. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.-C.: MobileNetV2: inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)

42. Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H.: MobileNets: efficient convolutional neural networks for mobile vision applications. CoRR (2017) arxiv:1704.04861

43. Meng, L., Lyu, B., Zhang, Z., C V, A., Kamitoku, N., Yamazaki, K.: Oracle bone inscription detector based on SSD, pp. 126–136 (2019). https://doi.org/10.1007/978-3-030-30754-7_13

44. Wang, M., Deng, W., Liu, C.-L.: Unsupervised structure–texture separation network for oracle character recognition. IEEE Trans. Image Process. **31**, 3137–3150 (2022). https://doi.org/10.1109/TIP.2022.3165989

45. Wang, M., Deng, W., Su, S.: Oracle character recognition using unsupervised discriminative consistency network. Pattern Recognit. **148**, 110180 (2024). https://doi.org/10.1016/j.patcog.2023.110180

46. Zhang, Y., Zhang, H., Liu, Y., Yang, Q., Liu, C.: Oracle character recognition by nearest neighbor classification with deep metric learning. In: 2019 International Conference on Document Analysis and Recognition (ICDAR), pp. 309–314 (2019). https://doi.org/10.1109/ICDAR.2019.00057

47. Lin, Y., Chen, D., Liang, S., Qiu, Y., Xu, Z., Zhang, J., Liu, X.: Wood color classification based on color spatial features and K-means algorithm. In: IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society, pp. 3847–3851 (2020). https://doi.org/10.1109/IECON43393.2020.9255104

48. Liu, C., Yin, F., Wang, D., Wang, Q.: Online and offline handwritten Chinese character recognition: benchmarking on new databases. Pattern Recogn. **46**(1), 155–162 (2013)

49. Otsu, N.: A threshold selection method from gray-level histograms. IEEE Trans. Syst. Man Cybern. **9**(1), 62–66 (1979). https://doi.org/10.1109/TSMC.1979.4310076

50. Nursuriati, J., Tengku, S., Zainab, B.: Noise removal and enhancement of binary images using morphological operations. In: 2008 International Symposium on Information Technology, vol. 3, pp. 1–6 (2008)

51. Fahmy, M.F., Thabet, M.: A fingerprint segmentation technique based on morphological processing. In: IEEE International Symposium on Signal Processing and Information Technology, pp. 000215–000220. IEEE (2013)

52. Hu, S., Wang, Q., Huang, K., Wen, M., Coenen, F.: Retrieval-based language model adaptation for handwritten Chinese text recognition. Int. J. Doc. Anal. Recognit. **26**, 109–119 (2023)

53. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1, pp. 886–893 (2005)

54. Steinley, D., Brusco, M.J.: Initializing K-means batch clustering: a critical evaluation of several techniques. J. Classif. **24**(1), 99–121 (2007)

55. Abuaiadah, D.: Using bisect K-means clustering technique in the analysis of Arabic documents. ACM Trans. Asian Low Resour. Lang. Inf. Process. **8**, 8 (2016). https://doi.org/10.1145/2812809

56. Tippaya, T., Nuntawut, K., Pongsakorn, D., Kittisak, K., Nittaya, K.: The clustering validity with silhouette and sum of squared errors, pp. 44–51 (2015)

57. Aranganayagi, S., Thangavel, K.: Clustering categorical data using silhouette coefficient as a relocating measure. In: International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2007), vol. 2, pp. 13–17 (2007)