Recognition of Oracle Bone Inscriptions by Extracting Line Features on Image Processing

Lin Meng
College of Science and Engineering, Ritsumeikan University, 1-1-1 Noji-higashi, Kusatsu, Shiga 525-8577, Japan
menglin@fc.ritsumei.ac.jp

Keywords: Recognition of Oracle Bone Inscriptions, Hough Transform, Clustering.

Abstract: Oracle bone inscriptions is a kind of characters, which are inscribed on cattle bone or turtle shells with sharp objects about 3000 years ago. Understanding these inscriptions can give us a lot of insight into world history, character evaluations, global weather shifts, etc. However, for some political reasons the inscriptions remained buried in ruins until their discovery about 120 years ago. The aging process has caused the inscriptions to become less legible. In this work, we design a system and proposal a recognition method for recognizing oracle bone inscriptions as a template image from an oracle bone inscription database, by using the line feature of the inscriptions. First we use Gaussian filtering and labeling to reduce noise and use affine transformation and thinning to extract the skeleton. Then we use Hough transform to extracting the line feature points by proposing a method of clustering. Finally, we calculate the minimum distance of the line feature points between the original image and the template images to perform the recognition. Experimental results shows that almost 80% of inscriptions are recognized as the most minimum distance and the second-most minimum-distance. And the proposal can recognized well, even if the noise and tilt happened in original images.

1 INTRODUCTION

Oracle bone inscriptions (OBIs), which first came into being about 3000 years ago in China, are some of the oldest characters in the world. OBIs were inscribed on cattle bone or turtle shells with sharp objects (Ochiai, 2008),(Pu and Xie, 2009) and are a kind of early literature used to record the history, weather, political activity, etc. taking place in China at that time. Understanding these OBIs can give us a lot of insight into world history, notable births, character evaluations, global weather shifts, etc. However, for political reasons OBIs remained buried in ruins until their discovery about 120 years ago. The aging process has caused these inscriptions to become less legible, and due to a lack of early research on the subject, it is now increasingly difficult to understanding what it is the OBIs have to say.

Because few people can read OBIs, the major OBIs recognition is that the experts of historian recognize OBIs by their experience. Currently, some researchers have suggested using image processing to recognize OBIs automatically. However, the recognition rate with this approach is not high enough and needs to be improved.

The most common OBI recognition method is rubbing, where the OBI surface is reproduced by placing a piece of paper over the subject and then rubbing the paper with rolled ink. Figure 1 shows an example of an oracle bone rubbing with the middle part showing an enlarged view. Several characters visible on the left side of the rubbing refer to a divination predicting that it will rain that day from 11 p.m. to 1 a.m.

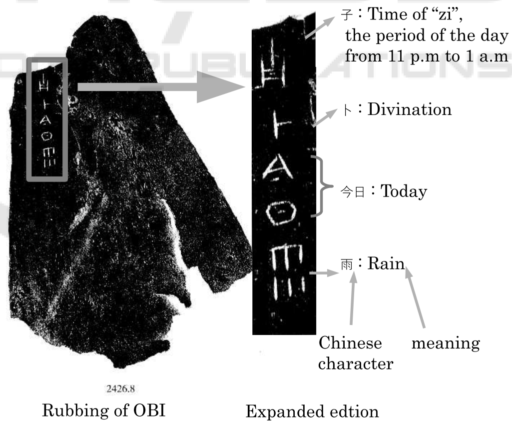


Figure 1: An example of OBIs.

A lot of these characters are made up of lines, which makes sense since they were inscribed by sharp objects. With this feature in mind, we have designed an OBI recognition system that uses Hough transform

Meng, L.
Recognition of Oracle Bone Inscriptions by Extracting Line Features on Image Processing.
DOI: 10.5220/0006225706060611
In *Proceedings of the 6th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2017)*, pages 606-611
ISBN: 978-989-758-222-6

image processing. The system recognizes inscriptions from an inscription database that contains images of normalized inscriptions similar to a dictionary. The normalized inscriptions are generated using character font software to make the characters smooth, clear, and straight, with uniformly thick strokes. These characters have been examined by historians, and the database is created by the researchers who belongs the letters college of Ritsumeikan University. More than 2000 normalized inscriptions are stored in the database(Ochiai, 2014).

The recognition system is comprised of four steps for recognition. The first step is noise reduction processing, where Gaussian filtering and labeling are applied to reduce noise. The second step is feature extraction pre-processing, which includes affine transformation(Schneider and Eberly, 2003) and thinning(L. Lam and Suen, 1992) for extracting the skeleton of OBIs. The third is line feature processing, which extracts the line feature points by Hough transform (Ballard, 1981). The fourth is recognition by calculating the minimum distance between the extracted line feature points of original and template OBI images.

The contributions of this paper are as follows:

1. Design of an OBI recognition system from noise reduction to recognition.

2. Proposal of a method for OBI recognition by Hough transform and clustering.

Section 2 of this paper discusses related work and section 3 describes the recognition method. Experiments and results are reported in section 4. We conclude in section 5 with a brief summary.

2 RELATED WORK

As technologies evolve, various researchers have attempted to recognize OBIs by image processing. However, few English papers have reported on OBI. We do know that the recognition rate needs to be improved.

(Li and Woo, 2008) and (Q. Li, 2011) presented a recognition method that treats OBIs as a non-directed graph for recording the features of end-points, three-cross-points, five-cross-points, blocks, net-holes, etc. However, due to the age of OBIs, some of the holes and cross-points that occur are not actually a part of the OBIs themselves, which increases the difficulty of the recognition. (Li and Woo, 2000) proposed a DNA method for recognizing OBIs. However, neither (Li and Woo, 2000) nor (Q. Li, 2011) provided details on any experiments.We have previously proposed several methods for recognizing OBIs

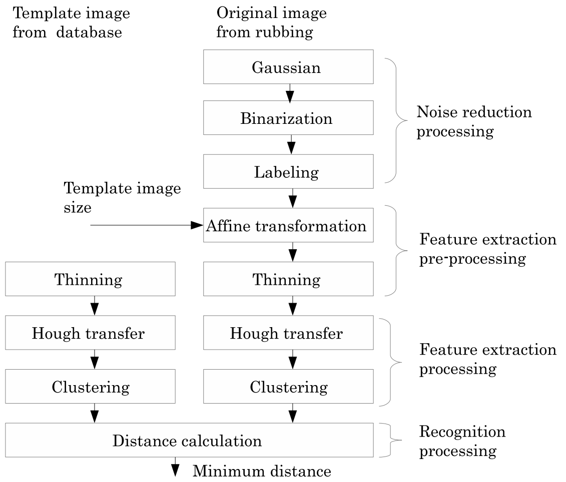


Figure 2: Flow of OBI recognition.

by template matching and by using Hough transform (L. Meng, 2016),(L. Meng and Oyanagi, 2015). However, the template matching was weak when the original character tilt, and (L. Meng and Oyanagi, 2015) did not properly process the tilt, either.

In the present work, we propose a complete recognition system from noise reduction to recognition, and consider the tilt.

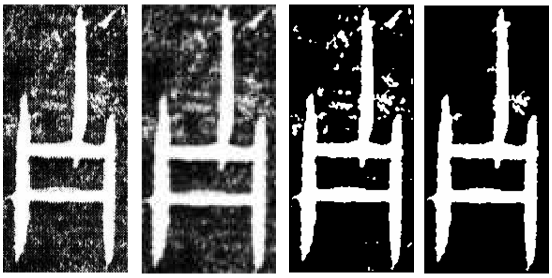
3 RECOGNITION PROCESSING

Figure 2 shows the OBIs recognition flow. The main processing includes noise reduction processing, feature extraction pre-processing, line feature extraction processing and recognition processing.

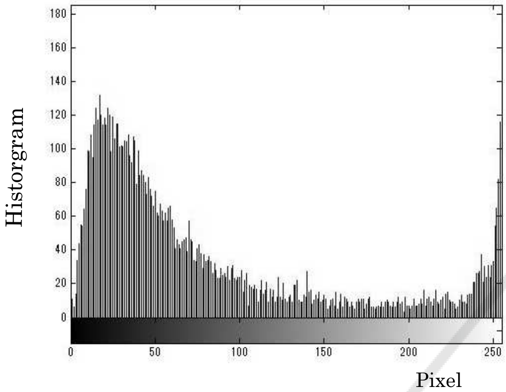
3.1 Noise Reduction Processing

Due to aging, many noises both big and small exist on OBI rubbings. Noise reduction processing is therefore an important part of the recognition process. Figure 3a) is the original image, the character means the period of ”zi”, which is a rubbing image cut from (Pu and Xie, 2009). As shown, both smaller noises such as fog and some bigger noises exist in the image.

We use Gaussian filtering and binarization for reducing the smaller noises. Formula (1) shows the Gaussian filter used for blurred images. Figure 3b) shows the Gaussian filtering results and Fig. 3e) shows the histogram of the Gaussian filtering results divided into two peaks. The Otsu method(Sezgin and Sankur, 2004) was used to decide the threshold for binarization and reduce the smaller noise. Figure 3c) shows the results of binarization, where the smaller noises (such as fog) are reduced successfully and the



a) Original image of "zi" b) Gaussian filtering c) Binarization d) Labeling



e) Historgram of Gaussian filtering

Figure 3: Noise reduction result.

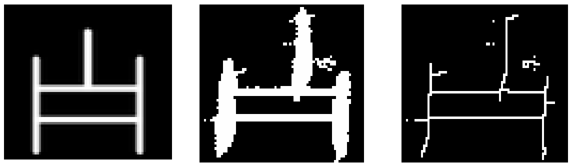
character becomes clear. However, the bigger noises remain.

$$f(x,y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \quad (1)$$

To reduce the bigger noise, labeling (L.F. He, 2008) is used. Labeling is a method that last scans the binarization image and counts the pixel numbers of each connected object. We know that bigger noises have only a few pixels while the connected characters have a lot of pixels, so pixel number represents a big change between noisy objects and character objects. We use a histogram method to detect big changes in the histogram of objects for detecting the threshold. If an objectfs pixel number is more than a threshold, the object will be left alone, and otherwise, the object is treated for noise reduction.

Figure 3d) shows the result of labeling. As shown, the bigger noises are reduced successfully and the character becomes more clear.

In Fig. 3d), there are some noises that we were not able to reduce due to the noise being closely connected with the characters. However, the experimental results discussed in section 5 demonstrate that these few remaining noises do not have a serious effect on the recognition.



a) Template image of "zi" b) Affine transformation c) Thinning

Figure 4: Feature extraction pre-processing.

3.2 Feature Extraction Pre-processing

Feature extraction pre-processing includes Affine transformation (Schneider and Eberly, 2003) and Thinning thinning (L. Lam and Suen, 1992).

For comparison with the template image in the database, the original image needs to be normalized into the database image space. Affine transformation is a map that transforms points and vectors in the original image space into points and vectors in the database image space. In other words, for the example of "zi", the labeling results space of Fig. 3d) need to be changed into to the template space of Fig. 4a). Formula (2) show the changing method, where (x_i, y_i) is the pixel axis in the original image, (x_c, y_c) is the character center axis of the template image, θ is the angle to which the original image will be changed as a result of the rubbing, and M is the size of the extension from the original image size and to the template image size.

Figure 4b) shows the affine transformation results, namely, the correct changing of the labeling result space of Fig. 3d) into the template space of Fig. 4 a).

$$\begin{pmatrix} x_i^* \\ y_i^* \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} M\times x_j \\ M\times y_j \end{pmatrix} + \begin{pmatrix} x_c \\ y_c \end{pmatrix} \quad (2)$$

After the affine transformation, we extract the skeleton from the original image using Hilditch's algorithm. The method considers each of the eight neighborhoods (p2,p3...P9) of the target pixel as one pixel (p1) and decides whether to peel it off or keep it a skeleton. Figure 4c) shows the thinning results of Fig. 4b).

3.3 Line Feature Extraction by Hough Transform

Hough transform is wildly widely used for extracting lines from images, by transforming the (x,y) space to the (r,θ) space.

This transformation is shown in Formula 3, where (x,y) is the size of the original image, r is the distance from the origin to the closest point on the straight line,

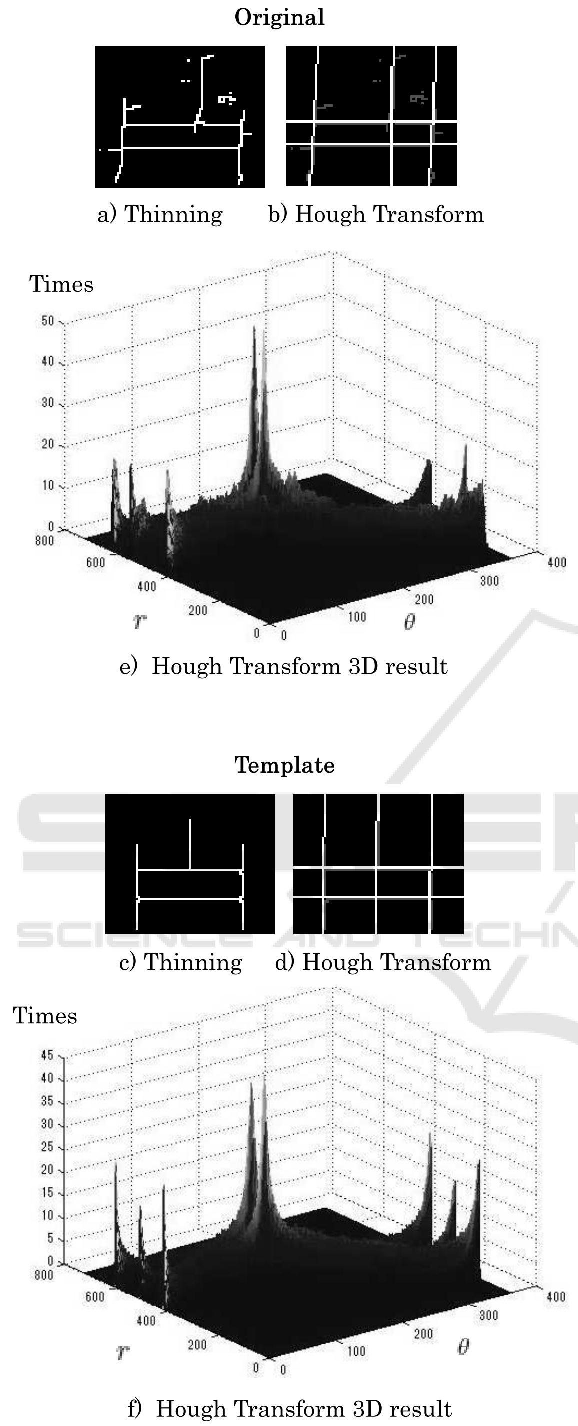


Figure 5: Hough transform.

and θ is the angle between the x axis and the line connecting the origin with that closest point.

$$r = x\cos\theta + y\sin\theta \qquad (3)$$

Every point in the (x,y) space will be transformed into a curve in the (r,θ) space by changing the θ from 0 to 2π. The method records the time the curve is passed in every pixel of the (r,θ) space. Finally, the recording time will be used for deciding the line.

Figure 5 e) and f) shows the three dimensions of Hough transform results on the (r,θ) space for the thinning result of the original (Fig. 5 a)) and template (Fig. 5 c). We found there are eight largest points that make up the feature line point in Fig. 5 e),f), especially in the template of 5 f). The three points on the left and right are the same line. This is the case of (θ) being 0° and 360°. Therefore, the feature line points are five.

If we catch the eight largest points correctly and transform the (r,θ) space into (x,y) space again, it will be possible to generate the results of the Hough transform in Fig. 5 b) d). The times at which we find the largest points of the template and original are the same in Fig. 5 e),f). Hence, deciding on the largest points and calculating the distance between the original image and the temp image is helpful for recognizing the OBIs.

Below are definitions pertaining to the line feature point decision. Algorithm 1 shows the line feature decision.

$C_{(r,\theta)}$ is all points of the (r,θ) space.

$LC_{(r,\theta)}$ is the largest point in (r,θ) space that still does not be checked.

$LPs_{(r,\theta)}$ is a set of larger points that is decided as the line feature point. It keeps the area of every point by radius.

$SDis(LP,LC)$ are the distances between $LC_{(r,\theta)}$ and all of the $LPs_{(r,\theta)}$.

$MinDis_{(r,\theta)}$ is the smallest distance of $SDis(LP,LC)LP_{(r,\theta)}$.

$SLP_{(r,\theta)}$ is a point of $LP_{(r,\theta)}$ that generates the $MinDis$ with LC.

Algorithm 1: Line feature point decision.

$C \Leftarrow SortC$
while $C \neq NULL$ **do**
  Search *LC*
  Generate *SDis* by using *LC* and *LPs*, Search *MinDis*
  if *MinDis* is lower than the radius of *SLP* **then**
    do nothing
  else {*MinDis* is lower than (the radius of *SLP*+30)}
    record (the radius of *SLP* $\Leftarrow$ *MinDis*) into *LPs*
  else
    input the LC into *LP* and keep the axis
  end if
end while

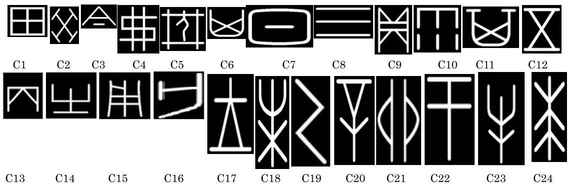


Figure 6: Template image.

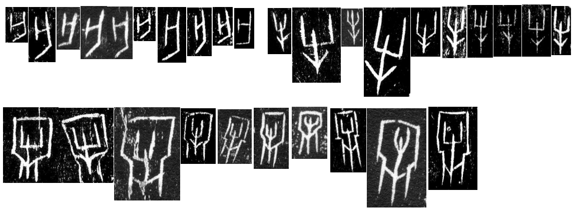


Figure 7: Original image.

3.4 Matching by Distance Calculation

After the line feature extraction processing, the line points in the (r, θ) space will be decided. Then the system calculates the minimum distance of the line points for the templates and original image. However, the line points of the template and the original are often different. Hence, The minimums are normalized by line feature point number. We defined the line points of the templates and original image into LargerNumber and SmallerNumber by comparing the numbers. We use the flown flume to normalize the minimum distance.

$$NormilzedDistance = \frac{Distance * LargerNumber}{SmallerNumber^2} \tag{4}$$

4 EXPERIMENTAL RESULTS

We used 24 templates as the dictionary and used 10 kinds original OBIs characters to do the experiment. Every kind original OBIs characters has 10 pieces OBIs, which are cutting from rubbing. Figure 6 shows the template image, with the character number shown below the images. Figure 7 shows a part of an original OBI image.

4.1 Recognition Results

Figure 8 shows the recognition rate of our method. We extract the most minimum distance time, second-most minimum distance time, and the others which and comparing them with the template. From the average, we found that almost 70% of ROIs are recognized as the most minimum distance, and 10%

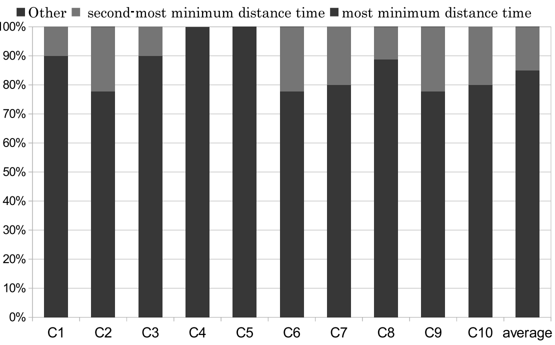


Figure 8: Recognition Rate.

are recognized as the second-most minimum distance time. The horizontal axis shows the character number of the original which that can be found in Fig.6.

4.2 Recognition Analysis

Figure 9 shows our analysis after the experiment,Figure 9 a) shows the original image. The three rankings of the character along with the template and the thinning are shown. The ranks 1 means the template which has the most most minimum distance with the original image. The original thinning is the result of the affine transformation by the top of the template in Fig. 9. About the line number, left is the template and right is the original. From the minimum distance ranking, we found the results to be correct. As for the thinning results, although noise was found, it did not have a negative effect on the recognition.

Figure 10 shows our analysis after the experiment, figure 10 a) shows the original image which have a large tilt. From the results, we found the tilt are renew which is shown in 10 b) of original. From the minimum distance ranking, we found the results to be correct. Although tilt happened in original image, it can be recognized correctly.

5 CONCLUSION

In this paper, we discussed our design of an OBIs recognition system and proposal proposed a recognition method by using Hough transform. The recognition includes noise reduction processing which to reduces the noise by Guaussian filtering and Labeling, feature extraction pre-processing, (which including affine transformation and Thinning) for extracting the skeleton of OBIs, and line feature processing, which extracts the line feature points by Hough transform. Then do the recognition is performed by calculating the minimum distance between the extracted line fea-

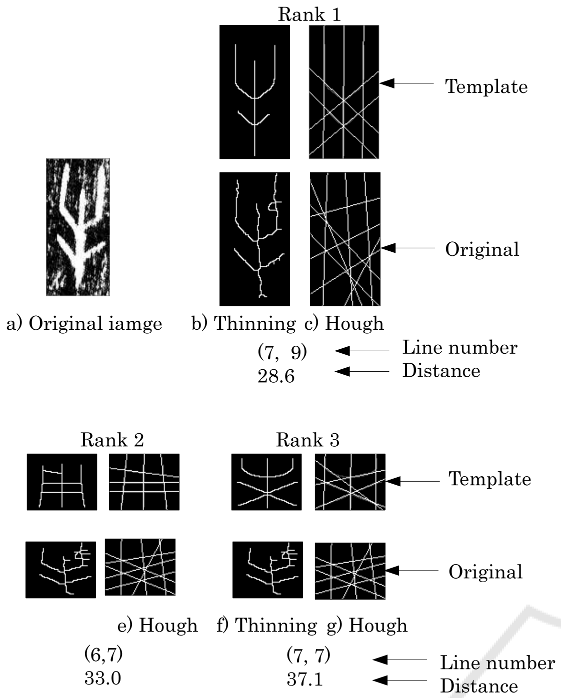


Figure 9: Recognition analysis.

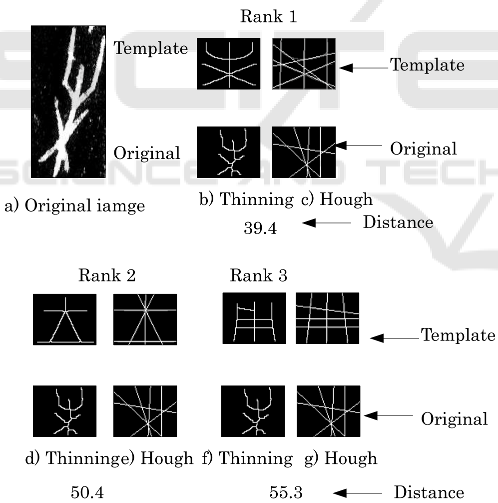


Figure 10: Recognition analysis about tilt.

ture points of the original OBI and the its corresponding template image of OBI. The method results shows showed that almost 80% of ROIs are recognized as the most minimum distance and the second-most minimum distance time. As for the thinning results, although noise was found, it did not have a negative effect on the recognition and although the tilt happened in original image, it can be recognized correctly.

ACKNOWLEDGEMENTS

This work was supported in part by a grant-in-aid for scientific research (60615938) from JSPS.

REFERENCES

Ballard, D. H. (1981). Generalizing the hough transform to detect arbitrary shapes. In *Pattern Recognition*. Elsevier.

L. Lam, S. W. L. and Suen, C. Y. (1992). Thinning methodologies: A comprehensive survey. In *IEEE Trans. on Pattern and Machine Intelligence*. IEEE.

L. Meng, e. a. (2016). Recognition of inscriptions on bones or tortoise shells based on graph isomorphism. In *Int. J. of Computers Theory and Engineering*. IJCTE.

L. Meng, T. I. and Oyanagi, S. (2015). Recognition of oracular bone inscriptions by clustering and matching on the hough space. In *J. of the Institute of Image Electronics Engineers of Japan*. IIEEJ.

L.F. He, e. a. (2008). A run-based two-scan labeling algorithm. In *IEEE Tras. on Image Processing*. IEEE.

Li, F. and Woo, P. Y. (2000). The coding principle and method for automatic recognition of jia gu wen characters. In *Int. J. of Human-Computer Studies*. ScienceDirect.

Li, F. and Woo, P. Y. (2008). Sticker dna algorithm of oracle-bone inscriptions retrieving, computer engineering and applications. In *Computer Engineering and Applications*. CEA.

Ochiai, A. (2008). *Reading History from Oracular Bone Inscriptions*. Chikuma Shobo, Japan, 1st edition.

Ochiai, A. (2014). Oracle bone inscriptions database. In *http://koukotsu.sakura.ne.jp/top.html*. No.

Pu, M. Z. and Xie, H. Y. (2009). *Shanghai Bo Wu Guan Cang Jia Gu Wen Zi*. Shanghai Bo Wu Guan, China, 1st edition.

Q. Li, e. a. (2011). Recognition of inscriptions on bones or tortoise shells based on graph isomorphism. In *Computer Engineering and Applications*. CEA.

Schneider, P. K. and Eberly, D. H. (2003). *Geometric tools for computer graphics*. Morgan Kaufmann Publishers, USA, 1st edition.

Sezgin, A. and Sankur, B. (2004). Survey over image thresholding techniques and quantitative performance evaluation. In *J. of Electronic Imaging*. SPIE.