International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:10, No:10, 2016

Topology-Based Character Recognition Method for Coin Date Detection

Xingyu Pan, Laure Tougne

***Abstract*—**For recognizing coins, the graved release date is important information to identify precisely its monetary type. However, reading characters in coins meets much more obstacles than traditional character recognition tasks in the other fields, such as reading scanned documents or license plates. To address this challenging issue in a numismatic context, we propose a training-free approach dedicated to detection and recognition of the release date of the coin. In the first step, the date zone is detected by comparing histogram features; in the second step, a topology-based algorithm is introduced to recognize coin numbers with various font types represented by binary gradient map. Our method obtained a recognition rate of 92% on synthetic data and of 44% on real noised data.

***Keywords*—**Coin, detection, character recognition, topology.

I. Introduction

THE image-based coin identification has become an active research topic for recent decade, but nowadays the coin identification is still a time-consuming task which requires expensive human investigation. The high level of variability within a class due to non-industrial manufacturing is the greatest problem to deal with ancient coins [1]. However, when it comes to modern machine-struck coins, the high inter-class similarity becomes the main challenge: coins from different monetary types could have almost identical visual patterns, because coin authors create a new coin type by adopting a similar, or even the same, design of the former one. A possible way to distinguish those monetary types with such high inter-class similarity is to detect subtle nuances of their visual patterns, and for doing that Pan et al. [2] has attempted in their research. Another more intuitive solution is to detect the date of those similar monetary types because in most cases they are released during different periods and the release date is a key legend graved in one face of the coin (see Fig. 1).

Fig. 1 Release period differentiates similar monetary types. DrapDolSE_rev (blue): 1795-1798; DrapDolLE_rev (red): 1798-1803

Xingyu Pan is with the LIRIS Laboratory, Univ Lyon, Lyon 2, F-69676, Lyon, France (phone: +33-478773942, e-mail: xingyu.pan@liris.cnrs.fr).
Laure Tougne is with the LIRIS Laboratory, Univ Lyon, Lyon 2, F-69676, Lyon, France (e-mail: laure.tougne@liris.cnrs.fr).

Detection of coin legends, including release date, is very challenging since the properties of coin characters could have extremely high variations (font, size, tilt, etc.), and all training-based methods will meet insufficient data in numismatic contexts. In addition, conventional character recognition methods based on segmentation of character region, such as OCR for scanned document, are inappropriate in this context, because characters of coins have very often the same color as its background, and the edge due to the relief could be subtle and discontinued in degraded coins. Moreover, other salient patterns in the coin sometimes own similar features as the legend to be detected. Despite all constraints, in this paper we present a novel method to detect and recognize the date of coins based on two points: 1) in general, the graved date in a modern coin is always composed of four numbers; 2) the position of the date is relatively predictable when compared with other legends. It is worth noting that there are surely exceptions for those assumptions, but a relative "regularity" of coin date has been verified for most modern coins.

The rest of this paper is organized as follows: Section II is the review of related works about coin identification and legend detection. Section III explains the proposed date detection step. Section IV describes the method developed to recognize numbers cropped from the extracted date. Section V presents the results of our experiments. In Section VI, conclusions are drawn.

II. Related Work

Conventional approaches of image-based coin identification are firstly based on training global features. For example, Fukumi et al. [3] use a neural network on edge images of coins as the first attempt, various feature extractions of coins are proposed, such as binary gradient features [4], edge-angle-distance distribution [5], Fast Fourier Transform on discretized gradient directions [6], multiple eigenspace gradient features [7], etc. Recently, template-based approaches have been adopted for ancient coins when there is no enough data for training. Kampel and Zaharieva [8] matched local feature points of coins by using SIFT descriptor, and then Zambanini and Kampel [9] extended this work by using SIFT flow for image matching. A coarse-to-fine scheme is also presented to reduce the computational time linear to the number of templates [10]. Also on ancient coins, Arandjelovic [11] introduced a locally-biased directional histogram which can represent geometric relationships between local feature points. Then, Anwar et al. [12], [13] proposed a modified bag of visual words model containing spatial information. In regards to modern coins, a template-based approach by using a graph-based

similarity score proposed by Pan et al. [2] can be considered as an attempt to deal with coins with a high inter-class similarity.

As mentioned before, characters of coins have totally different properties from those in traditional fields of character recognition. Therefore, in the previous study on ancient coins, detection of coin legends is considered as a particular object recognition task rather than conventional character recognition. Arandjelovic [14] described the character appearance by using a HoG-like descriptor, but this step is implemented in the whole coin recognition pipeline and he did not report individually his character recognition results. Zambanini and Kampel [1] and Kavelar et al. [15] used a single SIFT descriptor to describe the patch which contains the character and they reported a recognition rate of 25% to 68% according to different training size per class (15 to 50). Both their methods require off-line training so it is only applicable when sufficient and representative training data are available. In addition, their studies on Roman coins cannot be easily extended to worldwide modern coins with the same size of training data due to variations of fonts and patch appearances (see Fig. 2).

(a) Roman coins (b) Worldwide coins

Fig. 2 Variations of appearances on cropped character patches

Unlike legends of coins as a specific domain, character detection and recognition in general natural scenes receives significant attention [16]-[20]. Text detection methods can be grouped into three categories: sliding window based methods; connected components based methods, and hybrid methods [16]. In general, sliding window based methods are for cases in which character or word candidates have fixed size; component based methods are for cases in which regions of character candidates have distinct properties (color, texture, stoke width [17], etc.) from the background. As long as character candidates are correctly segmented and binarized, OCR-like methods [21] based on feature extraction and training are able to complete the character recognition task. However, in most of above studies a distinction between character foreground and background is still a prerequisite for success. Rare testing samples with coin- legend-like characters are often reported as typical fail examples in recent studies (see Fig. 3). Among a huge number of studies on character detection and recognition issues applying to other fields other than coins, we are interested in some contour-based methods which could adapt better to our context. Liu and Ikenaga [18] used gradient and geometric features of region contours to construct candidate text regions, dealing well with cases in which the foreground is difficult to be segmented by using connected components or texture features. For reading license plates, Thome et al. [22] used Reeb graph to present topology of binarized character shape, preceded by a pre-classification on contour information using a neural network, performing well on classifying characters with similar shapes, like 8/B or 2/Z.

Fig. 3 Failed examples of character detection and recognition in natural scenes

It is also worth mentioning that Deep Learning in neural networks seems to be powerful for many pattern recognition issues, including character recognition. Unfortunately, it is not the case in our numismatic context where we consider also rare coins for which we have few representative data for training.

III. Coin Date Extraction

Date extraction is carried out by searching the possible date zone in the coin photo. Even though a date could appear at any position of each side of the coin, in most cases its position is relatively limited especially when we know some *a priori* knowledge such as the country of the given coin. We could also consider that a pre-selection as been done using [2] for example. Therefore, a list of candidate zones is proposed in order to reduce the searching time over the entire image. According to our observations on more than 3000 modern American coins, the bottom of the obvers and the middle of the revers have the most chance to contain the date. As sometimes the date on the bottom runs in a circular way inside the coin border, a polar coordinates transform is used to put the circular date in a horizontal way. Thus, if we have no idea which side of the coins is in the image, the candidate zones are defined at the bottom part by $z_{1c}$and z_{1p} and, in the middle part by z_2. Let us denote by $Z = \{z_{1c}, z_{1p}, z_2\}$ the set of candidate zones (see Fig. 4).

Fig. 4 Candidate zones for date retrieval

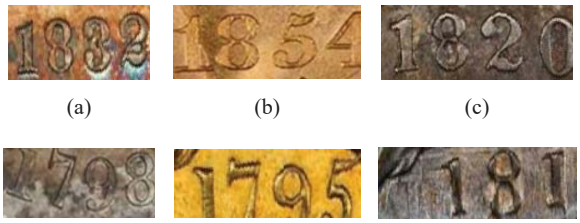
Let us define W the set of all possible sliding windows in a candidate zone $z_i \in Z$ and $l_{min}, l_{max}, h_{min}, h_{max}, r_{min}$and r_{max} the six extrema of width, height and aspect ratio for a possible date zone $w_{date} \in W$. For each sliding window w_i, we compute a normalized histogram $H(w_i)$ obtained by scanning vertically then horizontally the gradient map of the image inside the window. $H(w_i)$ will be compared with a mean histogram H_M obtained by using a large number of synthetic date models (see Fig. 4). The window w_{date} is the one with a histogram having the highest correlation to H_M.

$$w_{date} = \text{argmax}\, Cor(H(w), H_M), \quad (1)$$

$$w \in W(l_{min}, l_{max}, h_{min}, h_{max}, r_{min}, r_{max}, Z)$$

It is worth noting that if the image represents a side of the coin with no date, the maximum correlation should be smaller than the other side of the coin with date (experimentally, $Cor_{max}(H(w), H_M) < 0.65$), except for rare cases in which the other patterns have very similar features as a date of four numbers. From some examples of exacted date shown in Fig. 2, we can see our date extraction method is robust even when the distinction between the date and its background is not that obvious. The second line of Fig. 5 illustrates some typical imperfect extraction: (d) non-horizontal extraction; (e) partial extraction lacking a small part of numbers; (f) partial extraction lacking a complete number. However, only the last fault is fatal but rare in our results, other small imperfections influence little on recognition of numbers.

The extracted date is then cropped into four individual numbers to recognize. The crop is simply proceeded by searching the "valleys" of the histogram $H(w_{date})$ which are closest to the supposed separations (25%, 50%, and 75% of the width of w_{date}). For individual cropped number images, we could have imperfect but workable samples including slightly skewed numbers (Fig. 4 (f)), partially incomplete numbers (Fig. 4 (e)), and numbers with other salient pattern in their bounding box. In the following section, we will propose a recognition method dealing well with those noised data. The cropped numbers remain in binary gradient map and normalized into the size for recognition. Fig. 6 demonstrates the whole process of date extraction.



(a) (b) (c)

(d) (e) (f)

Fig. 5 Samples of extracted date zones

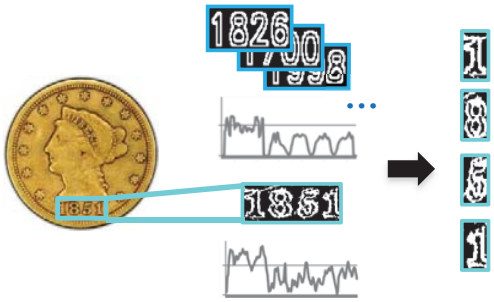


Fig. 6 Date detection and cropping by histogram of gradients

IV. Coin Number Recognition

The proposed method aims at reconstructing as far as possible the original character shape through a highly noised gradient map.

Let us denote by I_k the binary gradient map of the k^{th} image of number in our dataset to be recognized, and by S_k the shape contained in I_k. Such images contain noised characters with hollow type font (see Fig. 9 (a)). As we know we cannot directly get a binarization of S_k, it is impossible to recognize such characters in a conventional way. Moreover, due to our context that is to say, great variability of character fonts and lack of training data (less than 15 samples for certain class), we cannot implement classification-based object recognition like some methods of the literature, neither other feature training based methods requiring sophisticated classifiers with a large amount of training data. Thus, we propose a training free topology-based approach dedicated to recognition of noised numbers extracted from coins.

For a shape S_k and its associated convex hull ε_k, we define a *hole* as a connected component of background pixels inside the convex hull ε_k that does not belong to the shape S_k, and that does not touch the convex hull ε_k. For example, an "8" contains two holes: one is at top and another is at bottom. We also define an *opening* as a connected component of background pixels that does not belong to S_k but touches the convex hull ε_k. For example, a "5" contains two openings: one is at top-right and another is at bottom-left. More generally, we call such holes and openings bubbles in the image I_k. The topology of the shape S_k can be represented by its bubble map, denoted by B_k, containing all bubbles. Unwilling to describe in a complicate mathematical way, here using the term of topology simply means that we will neglect all intricacies of the shape S_k, for the sake of minimizing the influence from different sorts of noise, including a certain level of skewing and partial cropping produced during the step of date detection. Fig. 5 shows an example of such bubble map B_k where the bubbles are in white pixels.

Intuitively, we can recognize some numbers by knowing locations of all its bubbles. However, in case of a real image I_k extracted from more or less worn coins, more bubbles are extracted in its associated bubble map B_k because of the hollow type font and noise. Among all bubbles, those describing the real number shape are called *main bubbles* while those caused by noise are called *noise bubbles*. The task to identify the topology of S_k then becomes to determine if each bubble corresponds to *main* or *noise* bubble in the bubble map B_k. For doing that, several criteria are proposed to filter *noise bubbles* based on three hypotheses:

1) *Main bubbles* of the gradient map I_k should be at quasi similar locations as *main bubbles* of "ideal" binary shape of number.
2) *Noise bubbles* due to the hollow type font are in most cases discontinued in the shape S_k of the noised gradient map I_k, as shown in Fig. 7, otherwise we use the largest *noise bubble* to present S_k.
3) The total number of *main bubbles* is limited (≤4).

We would like to have each class of character represented by a distinct bubble map. However, real situations do not always meet ideal hypotheses mentioned above. On the one hand, due to variations of character fonts, certain numbers including 1, 3

and 4 could have more than one possible bubble maps and one of them could be similar to the bubble map of other classes. On the other hand, with a high level of noise, *main bubbles* are not always present while *noise bubbles* could be large enough to be recognized as false main bubbles. Thus, individual analysis of each bubble is as important as global filtering on initial bubble map.

We characterize each bubble $b_i \in B_k$ by using the following properties:

- τ_a: area ratio between b_i and ε_k;
- τ_l: length ratio between the border of b_i that touches ε_k and the perimeter of ε_k;
- θ: angle between the segment formed by geometric centers of b_i and ε_k and, the horizontal line ;
- r_1: aspect ratio of the bounding box of b_i;
- r_2: area ratio between b_i and its bounding box;
- r_3: area ratio between b_i and its own convex hull ε_i;
- r_4: length ratio between the border of b_i that touches ε_k and square root of the area of b_i.

Noise bubbles are filtered from B_k in a following way: firstly, we select the four largest bubbles according to τ_a (experimentally, $\tau_a > 2\%$) if they exist. Long curvy bubbles are removed as well according to τ_a and r_2, as they may be *noise bubbles* due to the hollow type font. Then, all retained bubbles $b_i \in B'_k$, by using τ_l and θ, can be categorized into one of the following labels:

- *CH*: center hole
- *TH*: top hole
- *BH*: bottom hole
- *TLO*: top-left opening
- *TRO*: top-right opening
- *BLO*: bottom-left opening
- *BRO*: bottom-right opening

Let us denote by l_i the label associated to the bubble $b_i \in B'_k$, $l_i \in \{CH, TH, BH, TLO, TRO, BLO, BRO\}$. Then, each labeled bubble contributes to predicting possible classes through a voting system. The class prediction is given by:

$$c_{predict} = \operatorname{argmax} V(c),\ c \in \{0,1,2,3,4,5,6,7,8,9\} \quad (2)$$

$$V(c) = \sum_{i=1}^{|B'_k|} v_c(l_i) \times \mathrm{w_c}(b_i) \quad (3)$$

where $v_c(l_i)$ corresponds to the vote of the bubble b_i for the class c according to its label l_i (see Table I), and $w_c(b_i)$ is its corresponding weight, which is an experimental parameter fixed according to noise levels as well as variations of character fonts. However, such parameters have not been set randomly. For example, a bubble predicting a class containing only one large bubble gives more weight to this class than to a class with more bubbles. Another example, for a bubble b_i with the label $l_i = TLO$, we set $w_7(b_i) > w_4(b_i)$ because we hope that the prediction of "7" will not be influenced by unfiltered *noise bubbles* and the prediction of "4" should have more votes from other bubbles. To differentiate a vote $v_c(l_i)$ for more than one class, $w_c(b_i)$ is also tuned by bubble properties r_1, r_2, r_3 and r_4: for example, a main bubble found in "1" should have a smaller r_1 than in "2", and a main bubble found in "3" should have a smaller r_3 than in "7". Generally, other bubbles with relatively normal shape and moderate size will be weighted equally by $w_c(b_i) = 1$. Although the voting system can be directly applied on the original bubble map B_k with *noise bubbles* having zero weight, the step of bubble filtering make the process more clear. Furthermore, the binary shape of number S_k can be reconstructed by the filtered bubble map B'_k, which opens access to implement other character recognition methods.

TABLE I
VOTE OF CLASS FOR DIFFERENT BUBBLE LABELS

l_i	$v_c(l_i)$									
	v_0	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9
CH	1	0	0	0	1	0	0	0	0	0
TH	0	0	0	0	0	0	0	0	1	1
BH	0	0	0	0	0	0	1	0	1	0
TLO	0	1	1	1	0	0	0	1	0	0
TRO	0	1	0	0	1	1	1	0	0	0
BLO	0	1	1	1	1	1	0	0	0	1
BRO	0	1	1	0	1	0	0	0	0	0

In conclusion, based on the bubble map detected from a binary gradient image of number, we are able to recognize the number contained in the image despite of certain level of noise. The properties of bubbles are used to filter and vote for class prediction. Fig. 7 gives the different steps of this coin number recognition.

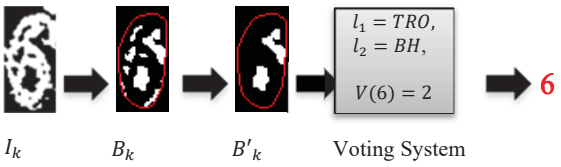


Fig. 7 Process of coin number recognition

V. EXPERIMENTS

In this section, we evaluate the performance of our approach for coin number recognition on two different datasets presented in binary gradient map. The first dataset contains images of numbers extracted from 375 coins with a high interclass similarity from the database *USA_Grading* (used in the work of Pan, et al. [2]), by the algorithm of coin date extraction presented in section III. After deleting extremely noised number images, which are impossible to be labeled even by a human (see examples in Fig. 8), we have obtained 900 images of numbers in this dataset. Notice that those samples are not equally distributed in each class and there persists very noisy images such as in Fig. 9 (b)) coming from over degraded coins. The other dataset is composed of 100 synthetic number images (10 samples per class). We simulate a similar hollow type font for those synthetic numbers with little noise as if they were extracted from well-conditioned coin photos in numismatic industries (see Fig. 9 (a)). For clarity, we name respectively the two datasets *coinNums* and *synNums*. Table II shows the distribution of data and Fig. 9 shows sample images in those

two datasets. We can observe that some highly noised numbers in *coinNums* are not easy to identify even for human eyes.

Our topology-based approach achieved a recognition rate of 44% (50% for the first three responses) on the dataset *coinNums* and 92% (95% for the first three responses) on the dataset *synNums*. Inspired by the work of Thome et al. [22], we compared our results with those obtained by an adapted template-based method using Reeb graph. Finally, we tested also a classic training-based method using a basic neural network on raw gradient images but with limited training data (10 images per class). As shown in Table III, the proposed training free method using the voting system on bubble maps over-performed on both test dataset.

Recognition of release date is not simply combining each recognized number together, because possible classes of a number at certain positions of the date are limited. For example, for modern coins the first number of the date can only be either 1 or 2. Furthermore, for the objective of recognition of monetary types, the recognized release date is used to provide supplementary information to recognize pre-classified coins, for example, as in Fig. 1, and the given release period of known monetary types will remarkably increase the recognition rate even in highly noised data. Thus, (2) could be rewritten as,

$$c_{predict} = \text{argmax}\, V(c), c \in C_{k,p} \subseteq \{0,1,2,3,4,5,6,7,8,9\} \quad (4)$$

where $C_{k,p}$ depends on the position and the release period of the pre-classified monetary types. Not having all release periods for all testing coins, in this experiment we apply only PD (Position in date) constraint on *USA_Grading* dataset with $C_{k,1} = \{1\}$, $C_{k,2} = \{7,8,9\}, C_{k,3} = C_{k,4} = \{0,1,2,3,4,5,6,7,8,9\}$. We can see the recognition rate has been boosted to 71%.

We analyzed also our results by visualizing the confusion matrix, denoted by M_c, in order to know which classes are more confused than others. In Table IV, the value at each position corresponds to the percentage of images belonging to the number i and recognized as the number j. The diagonal of M_c represents correct classification rate (green) and ideally, the values on the diagonal should be close to 100%. In the contrary, all positive values on the outside of the diagonal represent false classification rate (red). In Table IV-A, we can see that in the dataset with little noise such as *synNums* we have almost high values on the diagonals, while Table IV-B shows that for the real noised dataset *coinNums* some numbers are more easily to be recognized than other classes, and the class "3" is bad recognized because the key property r_3 of the *main bubble* that characterizes "3" is extremely sensible to noise. Interestingly, the class 4 has the lowest recognition rate on synthetic data but the highest recognition rate on real noised data, because in our synthetic database, *main bubbles* voting for 4 could be missing if they are not large enough while in noised database, unfiltered noise bubbles have more chance to compensate the missing *main bubbles*.

Fig. 8 Examples of "unreadable" images

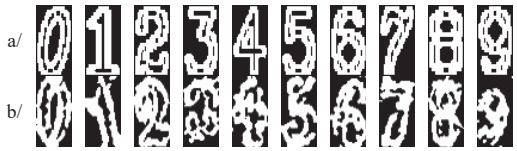


Fig. 9 Samples in two datasets (a/ *synNum*, b/ *coinNum*)

TABLE II
DISTRIBUTION OF DATA IN DIFFERENT DATASET

Dataset	0	1	2	3	4	5	6	7	8	9
coinNums	64	304	38	38	13	38	28	123	168	86
SynNums	10	10	10	10	10	10	10	10	10	10

TABLE III
RECOGNITION RATES OF DIFFERENT TESTS

Dataset	Data size	Method	Recog. rate
synNums	100	Proposed	92%
synNums	100	Reeb graph	66%
synNums	100	Neural network	75%
coinNums	900	Proposed + PD	71%
coinNums	900	Proposed	44%
coinNums	900	Reeb Graph	13%
coinNums	900	Neural network	36%

VI. CONCLUSIONS

We have presented a topology based number recognition in a particular numismatic context, preceded by detection of a possible date in the photo of the coin. Our training free recognition method is not only firstly dedicated to characters represented by edges (or hollow type font), but also adapts well to traditional number recognition applications. The results obtained by the proposed method are encouraging even on extremely noised numbers. Even though in current method the part of voting system is only for recognizing numbers, other stages can be generalized to other characters.

Compared to other contour based methods requiring clean gradients, our algorithm is robust to topological invariant noise (like small teeth) but relies strongly on a right continuity of gradient map, because we have to use the convex hull to generate the bubble map. Theoretically we cannot deal with over graded coins in which the exterior edges of extracted numbers are disconnected. Better quality of professional coin photos and more optimized gradient detection and cleaning algorithms will largely increase the recognition rate. We assume that the topology of number shape is invariant through most variations of type fonts, but we still meet special font types that cannot be tackled by the proposed method (see Fig. 10). In the future work, a graph-based analysis on external contour (plat or curved) like in [22] is considered to add into our method.

From a numismatic perspective, the motivation of this research is to distinguish different monetary types with a high inter-class similarity by their different release date. In the future

work, we will implement the proposed method into the pipeline of the work of Pan et al. [2], in order to increase the identification rate.

Fig. 10 Examples of "6" and "9" in special font with no closed hole

TABLE IV-A
CONFUSION MATRIX: *SYNNUMS*: PROPOSED METHOD

	0	1	2	3	4	5	6	7	8	9
0	100%	0%	0%	0%	0%	0%	0%	0%	0%	0%
1	0%	70%	0%	0%	0%	0%	0%	30%	0%	0%
2	0%	0%	100%	0%	0%	0%	0%	0%	0%	0%
3	0%	0%	0%	80%	0%	0%	0%	20%	0%	0%
4	0%	0%	10%	0%	70%	0%	0%	0%	0%	20%
5	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%
6	0%	0%	0%	0%	0%	0%	100%	0%	0%	0%
7	0%	0%	0%	0%	0%	0%	0%	100%	0%	0%
8	0%	0%	0%	0%	0%	0%	0%	0%	100%	0%
9	0%	0%	0%	0%	0%	0%	0%	0%	0%	100%

TABLE V-B
CONFUSION MATRIX: *COINNUMS:* PROPOSED METHOD

	0	1	2	3	4	5	6	7	8	9
0	61%	3%	8%	0%	2%	0%	17%	0%	3%	5%
1	2%	54%	32%	1%	3%	1%	1%	5%	1%	1%
2	8%	0%	61%	0%	5%	0%	3%	3%	16%	5%
3	5%	3%	45%	3%	3%	13%	5%	3%	18%	3%
4	8%	0%	8%	0%	77%	0%	8%	0%	0%	0%
5	3%	3%	16%	0%	18%	29%	26%	0%	5%	0%
6	0%	0%	29%	0%	11%	0%	50%	0%	11%	0%
7	2%	2%	30%	11%	1%	7%	1%	35%	6%	4%
8	8%	3%	21%	2%	7%	1%	15%	2%	37%	3%
9	6%	3%	29%	2%	5%	8%	3%	2%	9%	31%

ACKNOWLEDGMENT

This research is co-funded by the numismatic company G.E.N.I. and French national research association ANRT.

REFERENCES

[1] Zambanini, S., Kampel, M.: Improving ancient roman coin classification by fusing exemplar-based classification and legend recognition. In: New Trends in Image Analysis and Processing-ICIAP 2013. Springer (2013) 149-158.
[2] X. Pan, K. Puritat, L. Tougne, A New Coin segmentation and Graph-Based Identification Method for Numismatic Application. ISVC14, Las Vegas, LNCS, Vol.8888 (2014) 185-195.
[3] Fukumi, M., Omatu, S., Takeda, F., Kosaka, T.: Rotation-invariant neural pattern recognition system with application to coin recognition. IEEE Trans. Neural Network 3 (1992) 272-279.
[4] Nolle, M., Penz, H., Rubik, M., Mayer, K., Hollander, I., Granec, R.: Dagobert-a new coin recognition and sorting system. In: Proceedings of the 7th Internation Conference on Digital Image Computing-Techniques and Applications (2003).
[5] Van Der Maaten, L.J., Poon, P.: Coin-o-matic: A fast system for reliable coin classification. In: Proc. of the Muscle CIS Coin Competition Workshop. (2006) 7-18.
[6] Reisert, M., Ronneberger, O., Burkhardt, H.: A fast and reliable coin recognition system. In: Pattern Recognition. Springer (2007) 415-424.
[7] Huber, R., Ramoser, H., Mayer, K., Penz, H., Rubik, M.: Classification of coins using an eigenspace approach. PRL 26 (2005) 61-75.
[8] Kampel, M., Zaharieva, M.: Recognizing ancient coins based on local features. In: Advances in Visual Computing. Springer. (2008) 11-22.
[9] Zambanini, S., Kampel, M.: Automatic coin classification by image matching. In: Proc. of the 12th International conference on Virtual Reality, Archaeology and Cultural Heritage. (2011) 65-72.
[10] Zambanini, S., Kampel, M.: Coarse-to-fine correspondence search for classifying ancient coins. ACCV Workshops 2, 25–36 (2012).
[11] Arandjelovic, O.: Automatic attribution of ancient roman imperial coins. In:CVPR10. (2010) 1728-1734.
[12] Anwar, H., Zambanini, S., Kampel, M.: Supporting ancient coin classification by image-based reverse side symbol recognition. CAIP 2, (2013) 17–25.
[13] Anwar, H., Zambanini, S., Kampel, M.: Coarse-grained Ancient Coin Classification using Image-based Reverse Side Motif Recognition, Machine Vision and Applications, 26 (2-3): (2015) 295-304;
[14] Arandjelovic, O.: Reading ancient coins: Automatically identifying denarii using obverse legend seeded retrieval. In: ECCV12. Springer (2012) 317-330.
[15] Kavelar, A., Zambanini, S., Kampel, M.: Word detection applied to images of ancient roman coins. In Virtual Systems and Multimedia (VSMM), 2012 18th International Conference on (pp. 577-580). IEEE.
[16] Pan, Y. F., Hou, X., Liu, C. L. (2011). A hybrid approach to detect and localize texts in natural scene images. IEEE Transactions on Image Processing, 20(3), 800-813.
[17] Epshtein, B., Ofek, E., Wexler, Y. (2010). Detecting text in natural scenes with stroke width transform. In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on (pp. 2963-2970). IEEE.
[18] Liu, Y., Ikenaga, T. (2006). A contour-based robust algorithm for text detection in color images. IEICE transactions on information and systems,89(3), 1221-1230.
[19] Zhu, A., Wang, G., Dong, Y. (2015). Detecting natural scenes text via auto image partition, two-stage grouping and two-layer classification. Pattern Recognition Letters, 67, 153-162.
[20] Sun, L., Huo, Q., Jia, W., Chen, K. (2015). A robust approach for text detection from natural scene images. Pattern Recognition, 48(9), 2906-2920.
[21] Trier, Ø. D., Jain, A. K., Taxt, T. (1996). Feature extraction methods for character recognition-a survey. Pattern recognition, 29(4), 641-662.
[22] Thome, N., Vacavant, A., Robinault, L., Miguet, S. A cognitive and video-based approach for multinational license plate recognition. Machine Vision and Applications, 22(2), (2011) 389-407.