ORIGINAL RESEARCH
published: 09 May 2017
doi: 10.3389/fict.2017.00009

A New Database of Digits Extracted from Coins with Hard-to-Segment Foreground for Optical Character Recognition Evaluation

Xingyu Pan and Laure Tougne*

CNRS, Lyon 2, LIRIS, Univ Lyon, Lyon, France

Since the release date struck on a coin is important information of its monetary type, recognition of extracted digits may assist in identification of monetary types. However, digit images extracted from coins are challenging for conventional optical character recognition methods because the foreground of such digits has very often the same color as their background. In addition, other noises, including the wear of coin metal, make it more difficult to obtain a correct segmentation of the character shape. To address those challenges, this article presents the CoinNUMS database for automatic digit recognition. The database CoinNUMS, containing 3,006 digit images, is divided into three subsets. The first subset CoinNUMS_geni consists of 606 digit images manually cropped from high-resolution photographs of well-conserved coins from GENI coin photographs; the second subset CoinNUMS_pcgs_a consists of 1,200 digit images automatically extracted from a subset of the USA_Grading numismatic database containing coins in different quality; the last subset CoinNUMS_pcgs_m consists of 1,200 digit images manually extracted from the same coin photographs as CoinNUMS_pcgs_a. In CoinNUMS_pcgs_a and CoinNUMS_pcgs_m, the digit images are extracted from the release date. In CoinNUMS_geni, the digit images can come from the cropped date, the face value, or any other legends containing digits in the coin. To show the difficulty of these databases, we have tested recognition algorithms of the literature. The database and the results of the tested algorithms will be freely available on a dedicated website.[1]

Keywords: test database, character recognition, coin, date, digits

OPEN ACCESS

Edited by:
Robin Strand,
Uppsala University, Sweden

Reviewed by:
Thanh Duc Ngo,
Vietnam National University, Vietnam
K. C. Santosh,
University of South Dakota, USA
Sebastian Zambanini,
Vienna University of Technology, Austria

****Correspondence:***
Xingyu Pan
xingyu.pan@liris.cnrs.fr

Specialty section:
This article was submitted to Computer Image Analysis, a section of the journal Frontiers in ICT

Received: *30 September 2016*
Accepted: *13 April 2017*
Published: *09 May 2017*

Citation:
Pan X and Tougne L (2017) A New Database of Digits Extracted from Coins with Hard-to-Segment Foreground for Optical Character Recognition Evaluation. Front. ICT 4:9. doi: 10.3389/fict.2017.00009

INTRODUCTION

Nowadays, character recognition methods have been widely implemented into various applications in our daily life like license plate recognition system, mail sorting machine, and *Google Instant Camera Translation*. For each category of application, appropriate databases are essential requirements to develop adapted algorithms. Even a large amount of applications are developed on a small set of private data collected by researchers, public databases play more and more important roles for developing generalized methods and launching competitions of different approaches. In a brief

[1] http://liris.univ-lyon2.fr/Coins/

view of available databases of character recognition, there are MINST (LeCun et al., 1998), IAM (Marti and Bunke, 2002), and IRESTE (Viard-Gaudin et al., 1999) for handwriting recognition; Chars74K (de Campos et al., 2009), NEOCR (Nagy et al., 2011), KAIST (Lee et al., 2010), SVT (Wang and Belongie, 2010), MSRA (Hua et al., 2004), and ICDAR datasets (Lucas et al., 2003; Shahab et al., 2011; Karatzas et al., 2013, 2015) for texts on natural scenes. In a recent survey about text detection and recognition, Ye and Doermann (2015) summarize and compare features such as sources, orientations, languages, and information of training/test samples for those commonly used databases. In studying these different databases as we were searching one to test an algorithm dedicated to digit recognition extracted from coins, we noticed the following elements: there is no such database and the existing databases do not present the same difficulties. For example, first, concerning the handwritten characters, although there is great intra-class variability, a correct segmentation of the character foreground is always considered as a prerequisite for most recognition methods. Second, concerning the texts detected in natural scenes, despite the complexity of background, in most cases the foreground has quasi-uniform color or texture and such characters are still assumed to be extremal regions differentiated from the background (Sun et al., 2015). Furthermore, the challenge in the context of natural scenes is rather the localization of the text itself then the recognition. However, on natural scenes, some rare samples having "hard-to-segment" foreground are often reported as acceptable failure cases (see **Figure 1**). Those characters having the same color in foreground as in background, like characters on coins, draw our attention. As a matter of fact, such characters are seldom studied in character recognition issues but they are largely presented in sculptures, industrial products, coins, artworks, etc. Let us call "hollow-type font" such characters have the same color or texture in foreground as in background. Up to now, there is no specific database of characters in hollow-type font.

However, a special type of characters called drop cap letters is not so far from hollow-type font. In fact, with a growing interest in digitally perverting historical documents, recognition of drop cap letters has become a new research topic. A database of drop cap letters out of BVH database is proposed in the work of Landré et al. (2009), and other researchers built their own databases of drop cap letters based on online search. A drop cap letter is a graphic object composed of three principle elements: the letter, the pattern, and the background (Coustaty et al., 2011). Although such special characters can be considered without leading foreground color, rich high frequency patterns around the central letter encourage researchers to extract the foreground of letter from the textured background. Thus, many methods for drop-letter recognition are still based on a complicated foreground extraction and then followed by classic optical character recognition (OCR)-based character recognition methods (Coustaty et al., 2011) or other approaches based on segmented regions (Jouili et al., 2010). Unlike drop cap letters with patterned background, our purpose is to propose a database of characters with "hard-to-segment" foreground in a more general way: the foreground and the background have the same properties in terms of color and texture.

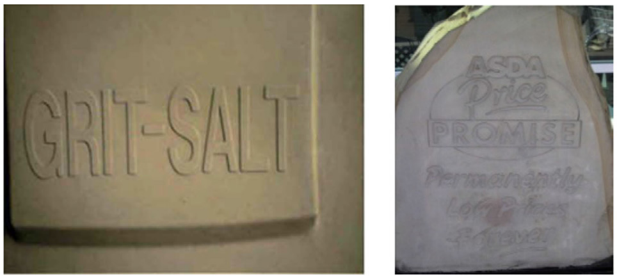


FIGURE 1 | **Failed text detection cases in natural scenes with "hard-to-segment" foreground, reported in the previous works by Pan et al. (2011) (left) and Sun et al. (2015) (right).**

As image-based coin recognition has become an active research topic for recent decade, various algorithms based on training global features (Fukumi et al., 1992; Huber et al., 2005; Van Der Maaten and Poon, 2006; Reisert et al., 2007) or matching local features (Kampel and Zaharieva, 2008; Arandjelovic, 2010; Pan et al., 2014) have been proposed. At the beginning, studies were limited in classification of modern coins where legends were treated no differently as other patterns in relief. In our opinion, there are two reasons why researchers were not interested in reading coin legends. First, detection and recognition of coin characters in hollow-type font is a real challenge to obtain good results because of their hard-to-segment foreground, not to mention their extremely high variations in terms of font, size, tilt, etc. and, the possible overlap with other decorative patterns (see **Figure 2**). Second, most legends may stay unchanged among different coin types of one country at the same period, for example, "LIBERTE EGALITE FRATERNITE" on different French franc coins. Thus, the differences in terms of overall appearance are easier to differentiate different coin classes than local legends. First attempts to deal with characters in hollow-type font were made by researchers trying to classify ancient coins. To classify manual struck ancient coins, the overall appearance is sometimes confusing due to a high intra-class difference. For example, in some cases for ancient coins with an emperor portrait as the main pattern, the different portraits can be the same emperor and the similar portraits can be different emperors. Therefore, reading legends indicating the emperor's name can assist a lot to classify those ancient coins. To solve such challenges within characters extracted from ancient coins, researchers have opted for solutions rooted in object recognition rather than character recognition methods. Arandjelovic (2012) describes the character appearance by using a HoG-like descriptor, but this step is implemented in the whole coin recognition pipeline and he neither reported individually his character recognition results nor published the

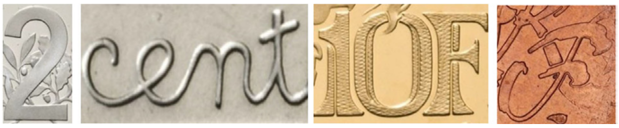


FIGURE 2 | **High variations of characters on coins.**

database of the extracted characters. Kavelar et al. (2012) and Zambanini and Kampel (2013) used a single SIFT descriptor to describe the patch that contains the character, and they reported a recognition rate of 25% to 68% on 18 classes using different training sizes per class (15–50). But, once again, the authors did not publish the databases of characters extracted from coins.

In a recent study of identification of modern coins under numismatic context, Pan et al. (2014) highlighted high inter-class similarities between subclasses of certain coin series: extremely similar but non-identical patterns can be found on different monetary types, especially when coin authors use the design of the former type to create the new one without modifying it too much, like examples shown in **Figure 3**. In the previous benchmark of modern coin sorting competitions, those subclasses were not required to be distinguished (Nölle et al., 2004). However, it is important to differentiate those coin subclasses for coin dealers and collectors. To address this problem, the most reliable and intuitive solution is to detect the release date because in most cases coins belonging to visually similar monetary types are released during different periods, marked by the release date stuck on one of the coin face. Therefore, it is required to detect and read coin date composed of digit characters in hollow-type font. For doing that, a training free topology-based digit recognition method has been proposed to address this problem (Pan and Tougne, 2016). In this work, digit characters of various fonts are extracted from the background of coin metal using contours determined by relief surface. Thus, based on the work of Pan and Tougne (2016), we construct a more comprehensive database of digits extracted from coins and propose to make it available to the scientific community. This database will be publicly available under Creative Commons license to encourage other researchers to propose algorithms to deal with such special characters and to report their results on the associated website that will be available. Moreover, we built this database based on professional coin photographs, for the sake that this database can largely assist image-based numismatic studies.

The rest of the article is organized as follows: Coin database section introduces the available databases of coin photographs from which we select to build our databases of digits in hollow-type font. Digit database: CoinNUMS section describes the detailed properties of the proposed database and how we build it. Experiments and results section shows experiments carried out on the proposed database. In the final section, conclusions and future works are drawn.

COIN DATABASE

Recently, several databases of coin photographs have been built mainly for image-based coin recognition issues. Establishing such a database with a decent data scale is much more difficult than building one for face recognition, for example, due to availability of rare coins and highly required photograph quality. Therefore, most online amateur photographs taken by coin dealers and collectors cannot be used as eligible data. It requires that museums, numismatic companies, and research institutions make effort to collect coins and take photographs under strictly controlled conditions. Meanwhile, how to shoot quality coin photographs through a structured and systematic approach is still an open field for professional photographers. Interestingly, even among professional coin photographs, criteria of quality could differ in different application contexts. To display an esthetic and shiny coin appearance to coin dealers and collectors, one or more angled lighting could be applied; however, to analyze small details on coins, a quasi-uniform light condition is expected. In general, a quasi-uniform background, a minimal shadow, and constant controlled light conditions are main characteristics of numismatic databases used in computer vision-assisted applications. In this section, we will introduce coin photograph databases in the literature of computer vision, including those from which we build our database of digit characters.

MUSCLE CIS

MUSCLE CIS is considered as the most used numismatic database with a significant data scale. It has been first introduced in 2003 for assisting to sort pre-Euro coins from 12 European countries by an image-based coin sorting system called Dagobert (Nölle et al., 2003). Afterward, the ARC Seibersdorf research center who developed Dagobert sorting system defined MUSCLE CIS benchmark for coin recognition tasks, followed by two benchmark competitions in 2006 and 2007. The MUSCLE CIS 06 contains in total 120,000 coin face images, unequally divided into 692 types (country + facial value) with 2,270 different coin face classes. In the benchmark, coin faces with similar but different patterns are considered as similar subclasses belonging to the same class, which need not be distinguished. Images in MUSCLE CIS 07 are based on MUSCLE CIS 06 but with synthetic occultation. All coin photographs, taken on black convey belt, are preprocessed into grayscale images with a normalized size of 640 × 576 pixels, composed of a upper part containing uncropped coin photograph and a lower part containing printed

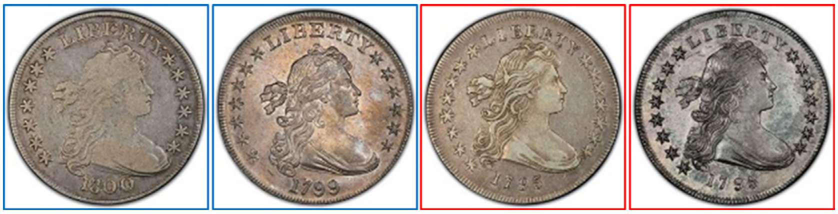


FIGURE 3 | **Release period differentiates similar monetary types: DrapDolSE_rev (blue): 1795–1798; DrapDolLE_rev (red): 1798–1803**.

information of its diameter, size, and ground truth class. Finally, rotation of coins in the photograph is totally random. However, this database, although very interesting, does not correspond to a real context of our study. The images are over preprocessed, and the resolution of the coin image is too low to extract small legends like the release date. Examples of coin images of MUSCLE CIS are shown in **Figure 4A**.

USA_Grading

USA_Grading is a coin photograph database collected from PCGS Photograde. Its main features are its equal class distribution, large intra-class variability, and high inter-class similarity. PCGS Photograde, by the American numismatic company PCGS, provides free online coin images for rough coin grading: coin dealers and collectors can fast estimate the grade of their coins by comparing them to graded image references. It includes major USA coin series in 18th, 19th and 20th centuries, and each coin face class contains more than 10 samples with different grades from worn circulated coins to uncirculated mint state coins. Moreover, some coins that have extremely similar patterns are labeled by different classes to be distinguished. As data in PCGS Photograde show great advantages as it can be used for developing robust image-based coin recognition algorithm, Pan et al. (2014) used coin images with grades above 10 (to remove extremely worn coins with almost no visible patterns) to construct USA_Grading database, which consists of 2,598 coin photographs divided into 148 different coin face classes. All data are cropped coins on a white background, normalized in 500 × 500 pixels. In this database, rotation difference of coins is less than 20° because during photographing coins are intentionally put straight with human error. Such database will be used in the following to construct the proposed digits database. Examples of coins images of USA_Grading are shown in **Figure 4B**.

Roman Republican Coins

A database of 180 Roman Republican coins divided into 60 classes was published by Zambanini and Kampel (2013). To our knowledge, it is the first database used to read characters on coins. Photographs are taken on a quasi-uniform background in a light color, and their sizes varies from about 1,000 × 1,000 pixels to about 2,000 × 2,000 pixels. Rotation of coins is also random in this database. In some studies about ancient coins, other databases of Roman coins or Macedonian coins have been collected from museums or numismatic websites, but they are not available. However, such coins contain few digit characters, and furthermore, specific characters on ancient coins do not correspond to our study. Examples of Roman Republican coins are shown in **Figure 4C**.

GENI Coin Photographs

GENI is a French numismatic startup, which develops new techniques to obtain high-quality professional coin photographs, especially for coin grading service. For this purpose, they try to minimize human-caused visual elements on the photograph, such as shadows or highlights, at the same time to enhance the intrinsic elements, such as contours, patina, and scratches. They provided us 217 unlabeled coin photographs mainly of French Franc coins for research use, and this database is to be expanded. The size of cropped coin images is about 2,000 × 2,000 pixels, and the rotation of coins is negligible. Such database will be used in the following. Examples of GENI coin photographs are shown in **Figure 4D**.

Other Coin Databases

Some researchers in computer science prefer to take their own photographs to constitute a small test base. However, most of

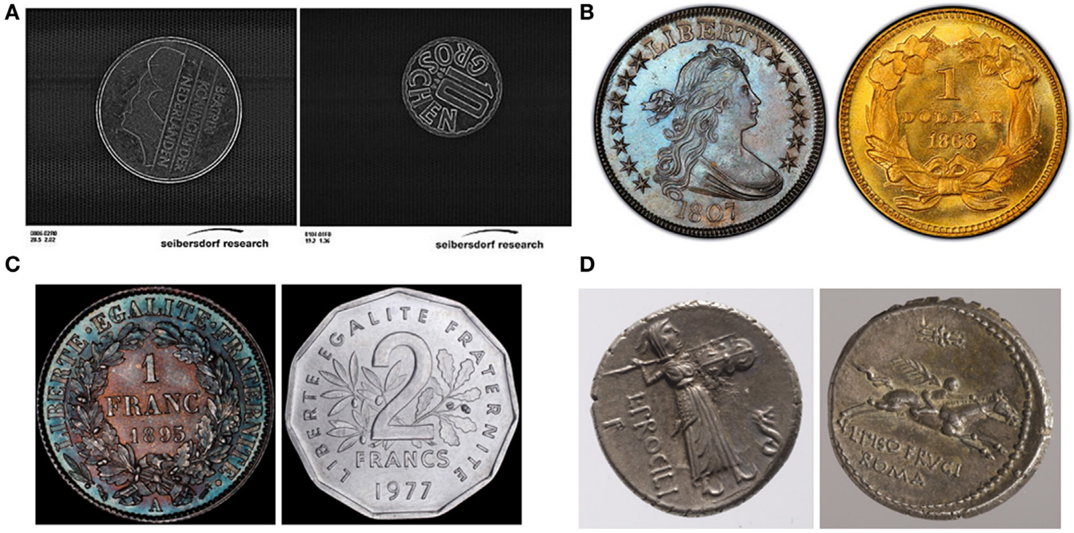


FIGURE 4 | **Examples of coin photograph databases: (A) MUSCLE CIS, (B) USA_Grading, (C) GENI coin photographs, and (D) Roman Republican coins**.

those small databases are amateur coin photographs and not published.

To sum up, to build a proper professional coin photograph database is quite challenging because large coin collections and advanced photographing techniques are necessary to accomplish this task. The quality of coin photographs depends also on the application context. For simple coin sorting systems, low-resolution images in grayscale are preferred; for analyzing coin details, images are kept in high resolution. **Table 1** shows a brief comparison between above databases, where "data quality" is a subject visual evaluation about how easy to read coin relief patterns. To construct our database of digit images, we chose USA_Grading and GENI coin photographs as image sources based on two criteria: (1) digit characters largely present in coin photographs and (2) the resolution of cropped digit characters is acceptable.

DIGIT DATABASE: CoinNUMS

Subsets of CoinNUMS

CoinNUMS is constructed from professional coin photographs taken in numismatic industrial conditions. It contains in total 3,006 cropped digit images, divided into 10 digit classes from 0 to 9. According to the original numismatic databases and the digit image extraction approaches, CoinNUMS is composed of the following three subsets:

– CoinNUMS_geni, built from GENI coin photographs, contains manually cropped digit images;
– CoinNUMS_pcgs_a, built from USA_Grading, contains automatically cropped digit images;
– CoinNUMS_pcgs_m, built from USA_Grading, contains manually cropped digit images.

The goal of using different coin photograph sources is to make our database have a large variety of character font types and different noise levels. USA_Grading contains a large number of worn coins while GENI coin photographs contain only well-conserved coins. In GENI coin photographs, digit characters could be the facial value, the release date or found in any other legends. However, in USA_Grading, digit characters are mainly found in the release date rather than other legends because the facial value of American coins are often written in letters, such as "QUARTER DOLLAR" or "TWENTY CENTS." As the release date is always composed of four digits and the position of the release date on American coins is relatively predictable, Pan and Tougne (2016) proposed an approach to extract efficiently digit characters on USA_Grading, but their automatic method cannot be applied on GENI coin photographs and may cause cropping errors such as partial cropping. Although the main difficulty that we want to present in our database is its intrinsic noise rather than the cropping errors, minor cropping errors due to automatic process are interesting to test the robustness of recognition methods. In the following part of this section, we will describe in detail each subset and how we obtained them by different approaches.

TABLE 1 | Comparison of current databases of coin photographs.

	Data size	Image size	Cropped	Rotation	Quality
MUSCLE CIS	120,000	640 × 576	No	Random	+ (gray scale)
USA_Grading	2,598	500 × 500	Yes	Straight	++
Roman Republican coins	180	>1,000 × 1,000	No	Random	+
GENI coin photographs	217	>2,000 × 2,000	No	Straight	+++

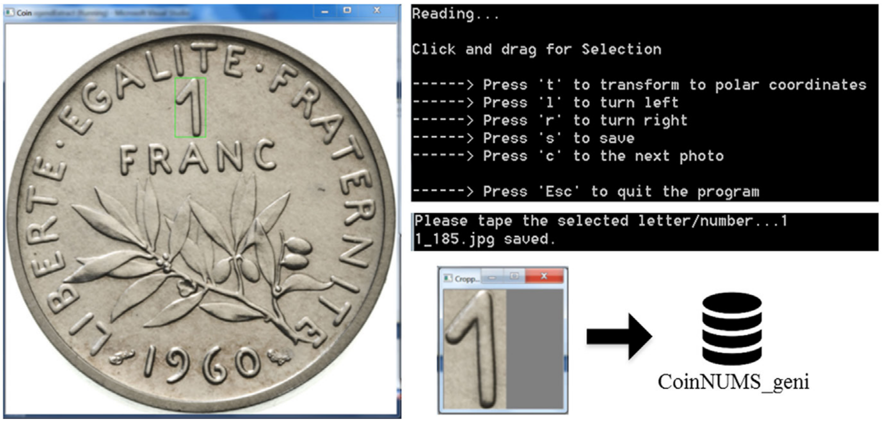
CoinNUMS_geni Subset

CoinNUMS_geni subset consists of 606 digit characters manually extracted from different legends (release date, facial value, etc.) of GENI coin photographs. Since images in GENI coin photographs are unlabeled, data in this dataset are simply labeled by *Class_Index,* where *Class* is the ground truth class labeled during the cropping process and *Index* is to indicate that the current image is the *Index*th data added into this class. For instance, 0_2.jpg means the second image of digit 0 added to CoinNUM_geni. As images provided by GENI are usually well-conserved coins with high resolution, we consider human eye has no difficulty to annotate correctly the ground truth class of those digit characters.

To facilitate character extraction from coin photographs, we developed a small tool called *CoinLegendExtract*, which allows users to extract subimages of character by drawing a rectangular bounding box on original photographs. For each coin photograph, *CoinLegendExtract* displays the original image and asks the user to draw a bounding box that contains the character. A zoomed window for cropped image preview will be displayed instantly for checking if the character is well cropped. After confirming the bounding box, the user will be asked to input the corresponding ground truth class of the character to be extracted. Then, the subimage inside the bounding box will be automatically saved into a corresponding directory, with the index assigned according to the current total number of images in this directory. If it is the first sample of one digit class, a new directory will be created. Although the rotation of coins in professional numismatic database is small, to extract rectified characters, a manual rotation function is proposed by *CoinLegendExtract*. Moreover, as some legends are struck in a circular way inside the coin border, a conversion function from the original image into polar coordinates allows the user to extract circular characters. **Figure 5** shows a captured screen of *CoinLegendExtract.* Cropped examples are shown in **Figure 6A**.

CoinNUMS_pcgs_a Subset

CoinNUMS_pcgs_a subset consists of 1,200 digit images automatically cropped from a pre-selected database of USA_Grading. Each image is labeled by *Class_PositionInDate_CoinName,* where *Class* is the ground truth class, *PositionInDate* is the digit order in the four-digit date and *CoinName* is the coin name. For instance, *1_1_GoldDollar2-55r.jpg* means the first digit 1 cropped from the release date of the coin photograph named *GoldDollar2-55r* (monetary type: *GoldDollar2*, grade: 55, side: reverse). In this



```
Reading...

Click and drag for Selection

------> Press 't' to transform to polar coordinates
------> Press 'l' to turn left
------> Press 'r' to turn right
------> Press 's' to save
------> Press 'c' to the next photo

------> Press 'Esc' to quit the program
```

```
Please tape the selected letter/number...1
1_185.jpg saved.
```

FIGURE 5 | Construction of subset CoinNUMS_geni by manual character extraction using *CoinLegendExtract*.

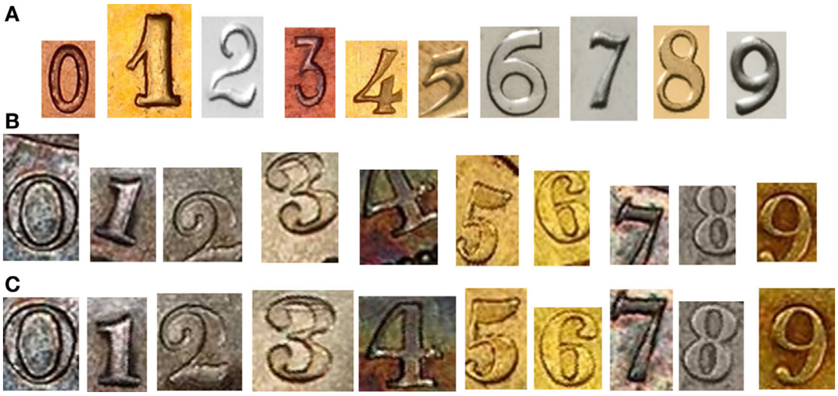


FIGURE 6 | Samples of all subsets in CoinNUMS: CoinNUMS_geni (A), CoinNUMS_pcgs_a (B), and CoinNUMS_pcgs_m (C).

dataset, the ground truth class is verified by both a priori information on the PCGS website and human eye verification.

Images of CoinNUMS_pcgs are extracted by a similar, but optimized, approach explained in the work Pan and Tougne (2016). Despite all constraints that we have mentioned on coin legend detection, the release date in coins of USA_Grading can be detected based on three assumptions: (1) the release date is always composed of four digits; (2) the position of the date is relatively limited; (3) the bounding box of the entire date has limits on its size and aspect ratio. According to our observations on coin photographs of USA_Grading, it is at the bottom of the obverse and in the middle of the reverse that we have the most chance to find the release date. Therefore, a list of candidate zones of possible date location is proposed to reduce the searching time over the entire coin image. We denote by z_1 the zone at the bottom of the coin surface and by z_2 the zone in the middle of coin surface, both with the area large enough to cover all possible positions of date. We observe also that the date in z_2 is always struck in a horizontal way, while dates in z_1 could be either put horizontally or runs in a circular way inside the coin border. Thus, to horizontalize a circular date in z_1, a polar transform is applied. Then, z_1 in original Cartesian coordinates is renamed as z_{1c} and its corresponding zone in polar coordinates is denoted by z_{1p}. After defining all candidate zones denoted by $Z = \{z_{1c}, z_{1p}, z_2\}$ (see **Figure 7**), a sliding window is used to search the date within Z. Let us denote by $w_i(l_i, h_i)$ a sliding window with the width l_i, the height h_i, and the aspect ratio $r_i = l_i/h_i$, and by W the set of sliding windows. W is defined by candidate zones Z and the six

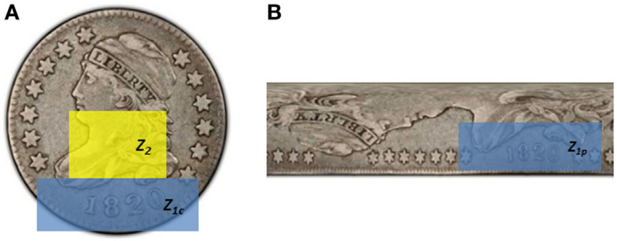


FIGURE 7 | Candidate zones for date retrieval in (A) original coordinates and (B) polar coordinates.

extrema of w_i: $l_{\min}$, $l_{\max}$, $h_{\min}$, $h_{\max}$, $r_{\min}$, and $r_{\max}$. Hence, by applying the candidate zones and the extrema of sliding windows, we can largely reduce the computational time instead of searching in the whole image with all combinations of parameters. Since a date is composed of four characters, the idea is to find on the gradient map a window that contains four groups of contours due to the relief surface. For doing that, for each sliding window $w_i \in W$, we compute a normalized histogram $H(w_i)$ by scanning vertically and horizontally the gradient map of the subimage inside w_i and comparing $H(w_i)$ to a mean template histogram of date H_M obtained on synthetic date images. The best-fitting window of date w_d is obtained by

$$w_d = \operatorname{argmax} \operatorname{Cor}(H(w_i), H_M), \tag{1}$$

where $\operatorname{Cor}(H(w_i), H_M)$ is the correlation between $H(w_i)$ and H_M.

Based on the detected date window w_d, the character cropping process is carried out by simply searching the "valleys" of the histogram $H(w_d)$, which are closest to the supposed separations $l_d/4$, $l_d/2$, and $3l_d/4$, where l_d is the width of w_d. **Figure 8** shows the whole process of digits extraction from the original image. **Figure 9** shows that our date extraction approach is robust but still prone to some cropping errors. Those examples of imperfect detection can be categorized into:

1. non-horizontal extraction (see **Figure 9E**);
2. partial extraction lacking a small part of digit (see **Figure 9F**);
3. partial extraction lacking an essential part of digit (see **Figure 9G**);
4. partial extraction lacking a complete digit (see **Figure 9H**).

As mentioned before, cropping error is a kind of noise that we are likely to add into some data in CoinNUMS_pcgs_m. Among those imperfect detection examples, the first two kinds of errors influence little on the recognition results, while the last two kinds of errors are in impossibility to recognize the digit character. Cropped digits in CoinNUMS_pcgs_a are shown in **Figure 6B**.

CoinNUMS_pcgs_m Subset

CoinNUMS_pcgs_m subset consists of 1,200 digit images manually cropped from the same coin photographs in USA_Grading as we use to build CoinNUMS_pcgs_a. Each image is labeled by *Class_PositionInDate_CoinName,* as well as CoinNUMS_pcgs_a. The ground truth class is first labeled manually by human eye and then verified by its corresponding image in the CoinNUMS_pcgs_a. The purpose is that CoinNUMS_pcgs_m and CoinNUMS_pcgs_a are the exactly same data but cropped by two different approaches.

CoinLegendExtract is used to manually crop data for CoinNUMS_pcgs_m. During the cropping process, the user is asked to input the ground truth class then the position of the current digit character in the release date. By doing that, each image in CoinNUMS_pcgs_m has the same label as its corresponding

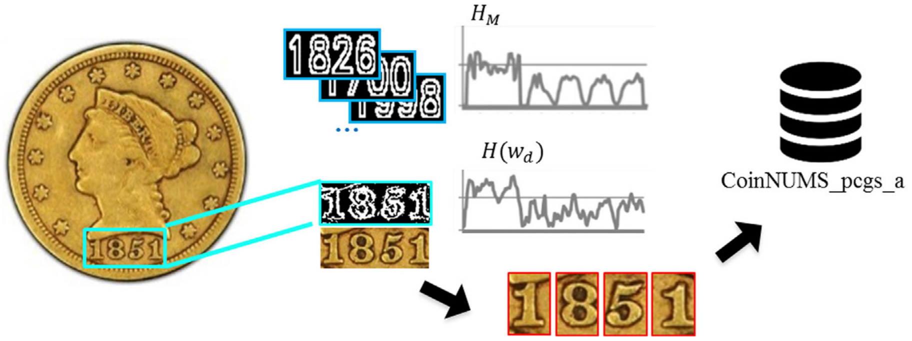


FIGURE 8 | **Construction of subset CoinNUMS_pcgs_a by automatic coin date detection and cropping**.

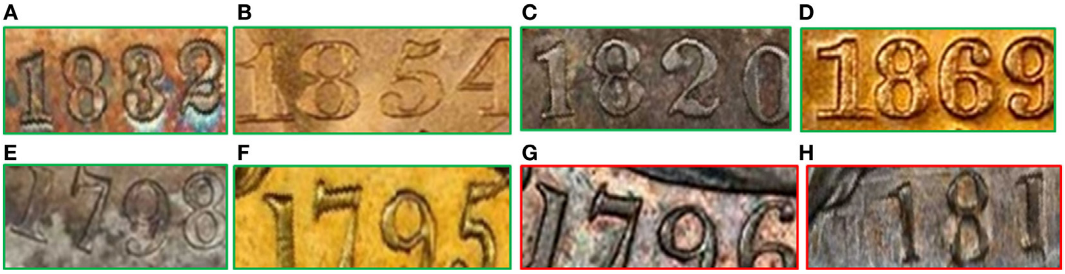


FIGURE 9 | **Samples of extracted date zones: ideal cropping (A–D) and imperfect cropping (E,F) such as non-horizontal cropping (E), incomplete digit (F,G), and incomplete date (H)**.

image in CoinNUMS_pcgs_a, but without cropping error. Cropped digits in CoinNUMS_pcgs_m are shown in **Figure 6C**.

Comparison between the Two Subsets: CoinNUMS_pcgs and CoinNUMS_geni

Here, we would like to compare them by the following criteria.

"Hard-to-segment" foreground: similar color or texture between foreground and background is the most remarkable feature that we want to present in our database. Most data in our database represent well this feature except for several data that have their foreground and background in two distinct colors. It can be caused by color of patina for used coins and by *flan bruni* with matte legends on polished metal background for new coins (see **Figure 10B**).

Neatness: we use the term of neatness to differentiate level of noise on visual quality of different subsets. Data in CoinNUMS_geni are much neater than those in CoinNUMS_pcgs_a and CoinNUMS_pcgs_m. The main reason is that CoinNUMS_geni is built from only well-conserved coins, while other subsets are built from coins in different quality from totally worn state to mint-state. Besides, GENI coin photographs have better resolution and seem to have better controlled lighting conditions than those in USA_grading, which make digit characters have more apparent relief surface.

Entirety of character: we use entirety to describe whether the image contains the entire character within. This is guaranteed by manual cropping in CoinNUMS_geni and CoinNUMS_pcgs_m, but in CoinNUMS_pcgs_a, digit characters could have a small part outside of the image (see **Figure 10C**).

Skew: CoinNUMS_geni and CoinNUMS_pcgs_m have generally unskewed digit images thanks to manual cropping, while digit images in CoinNUMS_pcgs_a are prone to slight skew since some of them are not extracted on the right coordinates (see **Figure 10D**). However, the possible skew of our data is usually limited in a small degree.

External noise: other coin relief patterns, scratches, dirtiness, or material texture may be included in the bounding box of the cropped digit as external noise patterns. Sometimes, they may attach to or overlap with the digit to be recognized (see **Figure 10E**). We find such external noise patterns in all subsets but it happens more often in those built from USA_Grading.

Internal noise: internal patterns are material texture or artificial decorations inside the digit object. Although this type of noise will not change the global shape of the digit object, it may put obstacles to some contour-based methods (see **Figure 10F**). In the current database, only CoinNUMS_geni has rare samples with artificial internal noise, but CoinNUMS_pcgs_a and CoinNUMS_pcgs_m have more data with internal caused by patina and material texture.

Table 2 recapitulates the comparison of all subsets in CoinNUMS. The ideal data of digit with "hard-to-segment" foreground are shown in **Figure 10A**. Briefly, CoinNUMS_geni data are closer to the ideal data because of better quality of source coins and careful manual operation. From **Table 3**, we can see that in CoinNUMS data are not equally distributed in different classes.

EXPERIMENTS AND RESULTS

To show the difficulty of our database, we carry out the same experiments on CoinNUMS and on two subsets of Chars74K database (de Campos et al., 2009). We select digit images from two subsets of Chars74K: Chars74K_EnglishImg and Chars74K_EnglishHnd (see **Figure 11**). Chars74K_EnglishImg contains 593 digit images cropped from natural scenes with a large variation in scale and resolution, and Chars74K_EnglishHnd contains 550 binary handwriting digits in the same bounding box but rich in various handwriting-type fonts. Compared to

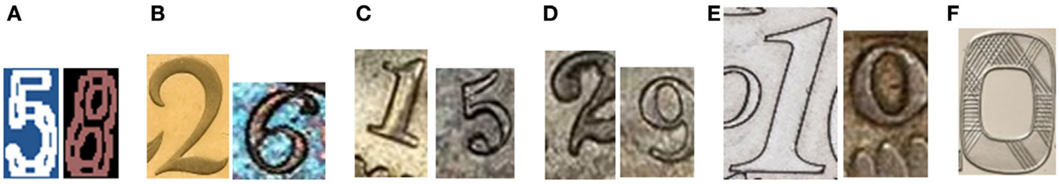


FIGURE 10 | Synthetic "ideal" digits in hollow-type font (A) and imperfect real data of CoinNUMS (B–F): distinguished foreground (B), skewed digit (C), incomplete digit (D), digit with external patterns (E), and digit with internal patterns (F).

TABLE 2 | Visual properties in different databases.

Dataset	Distinct foreground	Neatness	Entirety	Skew	External noise	Internal noise
Ideal data	No	Yes	Yes	No	No	–
CoinNUMS_geni	Yes[a]	Yes	Yes	No	Yes[a]	Yes[a]
CoinNUMS_pcgs_a	Yes[a]	No[a]	No[a]	Yes[a]	Yes[a]	Yes[a]
CoinNUMS_pcgs_m	Yes[a]	No[a]	Yes	No	Yes[a]	Yes[a]
Chars74K_EnglishImg	Yes[a]	No[a]	No[a]	Yes[a]	Yes[a]	Yes[a]
Chars74K_EnglishHnd	Yes	Yes	Yes	Yes[a]	No	No

[a] *Part of data, but not all of them, do not match the criteria of ideal data.*

our database featured by its "hard-to-segment" foreground, data in Chars74K_EnglishHnd have neat and binary foregrounds. As for data in Chars74K_EnglishImg, most foregrounds can be extracted by different intensity or color in spite of the low resolution and noise. Data distribution and general descriptive features of Chars74K_English and Chars74K_EnglishHnd are listed in **Tables 2** and **3**. We observe that in some databases, especially in Chars74K_EnglishHnd, the relative size and the position of the digit object could vary a lot. To make a fair comparison, we apply a preprocessing to make each digit object reframed in a normalized image before conducting the following experiments.

Methods

Template matching and structural analysis were first introduced into the early study of character recognition (LeCun et al., 1998). Then, with the development of machine learning techniques and the increasing number of available training data, learning-based approaches using extracted feature vectors have achieved a great success in many character recognition applications for recent two decades (Liu and Fujisawa, 2008). The performance of conventional character recognition methods mainly depends on the two steps: feature extraction step and classification step. Previous studies focused on how to extract the most pertinent and robust features to represent characters. Per an early but comprehensive survey (Trier et al., 1996), Fourier descriptors, projection histograms, contour profiles, Zernike moments, etc., can be used for feature matching or training. The selection of relevant features depends on character representation (gray-level image, binary foreground, skeleton, symbol contours, etc.) followed by a preprocessing step. Besides, to avoid compressing too much character representation into a single feature vector, DTW-based Radon features can be applied as powerful character shape descriptors (Santosh, 2011; Santosh and Wendling, 2015). Prevailing classification methods for character recognition, including statistical approaches, support vector machine, artificial neural network (ANN), and combined classifiers, are mainly based on feature vectors extracted from character images. Therefore, preprocessing technique and feature selection make difference to performance as well as training data and learning algorithm (Liu and Fujisawa, 2008). Recent studies on different classification methods, including deep neural networks which require both huge training data and computing power (Ciregan et al., 2012), have achieved recognition rates more than 99.5% on handwriting digits.

The objective of this part is to show the interest of the proposed database. For that purpose, we test three methods of digit recognition and highlight the difficulties which our database presents. As mentioned before, most methods in the literature are not adapted to our database. On the one hand, the "hard-to-segment" foreground prevents us from doing a proper preprocessing to obtain region-based or shape-based features; on the other hand, the scale of the current database is insufficient for training-based methods such on deep learning. For the proposed database, grayscale images and gradient maps that represent contour information are two kinds of reasonable outputs of the preprocessing step. Experiments of digit recognition are conducted by three methods based on bubble map (Pan and Tougne, 2016), Reeb graph (Thome et al., 2011), and basic ANNs. Every method is tested on images from both CoinNUMS_geni and CoinNUMS_pcgs. We also tested standard OCR engines such as the *ABBYY FineReader* and *Google Instant Camera Translation* because they can read digits in relief on bank cards, but they cannot deal with our digits extracted from coins. Obviously, digits in relief on bank cards are narrow and in standard font types but the situation of digits on coins are much more complex.

Bubble Map

To avoid sometimes a difficult learning, Pan and Tougne (2016) proposed an intuitive training free method to recognize digits on coins to recognize the release date of coins. The idea assumes that certain basic topological features remain unchanged among most variations of font types, including those in hollow-type font. For example, a digit "8" usually contains two holes: one at top and the other at bottom, and a digit "5" usually contains two openings, one at top right and the other at bottom left. Since contour information determined by the relief surface is the most detectable elements of our digits in hollow-type font, the original digit image is first converted into a binary gradient map. Morphological operations

TABLE 3 | Distribution of data in different databases.

Dataset	0	1	2	3	4	5	6	7	8	9	Total
CoinNUMS_geni	46	185	43	4	2	42	43	14	84	143	606
CoinNUMS_pcgs_a	80	331	57	81	24	66	70	149	213	139	1,200
CoinNUMS_pcgs_m	80	331	57	81	24	66	70	149	213	139	1,200
Chars74K_EnglishImg	105	79	55	50	47	64	64	48	32	34	593
Chars74K_EnglishHnd	55	55	55	55	55	55	55	55	55	55	550

Classes with insufficent data for training-based methods are marked by red.

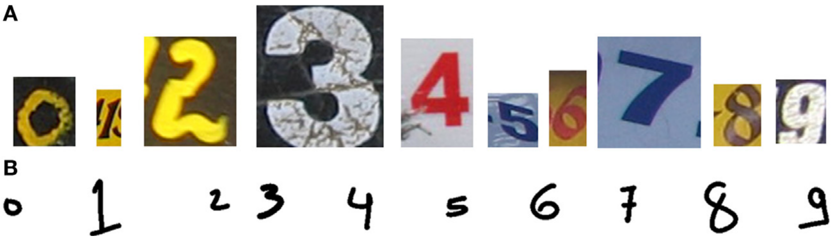


FIGURE 11 | **Samples of Chars74K_EnglishImg (A) and Chars74K_EnglishHnd (B)**.

are used to close some disconnected contours. After that, a binary map of all holes and openings, also called bubbles, is generate to predict the digit class through a voting system. This method based on topological features is robust to border noise and extra holes caused by hollow type, but sensible to the continuity of the external contours. Also, this method cannot recognize correctly some "exotic"-type fonts such as a digit "9" without the bottom hole or with two holes like "8" (see **Figure 12**).

Reeb Graph

For reading multinational license plates, Thome et al. (2011) used Reeb Graph to present the topology of character shape, preceded by a pre-classification on external and internal contour information. However, their original method applies only on the segmented foreground of characters.

Inspired by this method, we build an adapted template-based method using Reeb graph. At first, we tried to estimate and reconstruct the foreground shape of the character through a binary gradient map obtained by the same preprocessing as in the previous experiment. Then, a Reeb graph is extracted on the reconstructed binary shape and compared to template Reeb graphs representing ten classes. The advantage of Reeb graph is to detect in order the occurrences of split and merge in a character shape, especially including the curvy parts. However, it is not adapted to all "exotic"-type fonts such as a digit "3" without curvy parts, like an inversed "E." Besides, it depends heavily on the estimation of the binary character shape and it is very sensible to noise on the border.

Artificial Neural Networks

Artificial neural network is widely applied on handwriting recognition problem where exceptions and special cases prevent us from describing precisely and algorithmically the appearance of each digit class. In this experiment, we rescale all images into 32 × 32 pixels, then use a basic three-layered neural network designed for simple digit recognition with 10 output neurons and 1,024 input neurons that take each pixel as input feature.

RESULTS

From the **Table 4**, we can see that each method achieves generally higher recognition rates on Chars74K subsets than on CoinNUMS subsets. Compared with the large variability in type fonts presented in Chars74K subsets, the "hard-to-segment" foreground of our database appears to be the most challenging point that prevents from obtaining high recognition rates. In general, the highest recognition rate is obtained on Chars74k_EnglishHnd with binary foreground and the lowest recognition rate is obtained on CoinNUMS_pcgs_a built from a certain proportion of worn coins with cropping error. We can see that harder the foreground to be segmented, more difficult the digit can be recognized. We observe also that the difference of recognition rates obtained by different methods on the same database is much smaller on subsets of Chars74K than on our database.

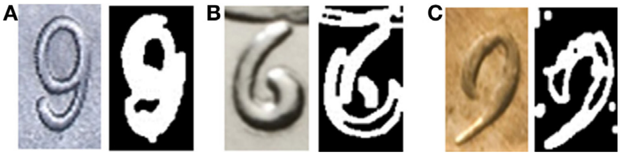


FIGURE 12 | **Difficult examples (scale normalized) for bubble map: (A) "9" with a closed bottom opening recognized as "8", (B) "6" with a broken bottom hole and a touched bar recognized as 2, and (C) "9" with a broken top hole recognized as 7**.

Since the relief contours are the only distinctive elements of the digit in our case, neatness and connectivity of the detected contours play a crucial role for Reeb graph and bubble map. From results shown in the **Table 4**, we can see that Reeb graph designed for license plates is not a good choice because it cannot resist to different kinds of noises. Although the recognition rates on CoinNUMS_pcgs_a and CoinNUMS_pcgs_m are less than 50%, bubble map obtains better results than other methods. As discussed in the work of Pan and Tougne (2016), the performance of the bubble map depends largely on the quality of the preprocessed gradient map. Obviously, data in CoinNUMS_geni, Chars74K_EnglishImg and Chars74K_EnglishHnd have better chance to have connected external contours on their gradient maps than those extracted from worn coins. Apart from noised gradient maps which may change the original digit topology, some exotic font types containing self-touching or broken holes will also cause wrong classifications (see **Figure 12**). It is worth mentioning that false recognition by bubble map in Chars74k_EnglishImg and Chars74k_EnglishHnd is mainly caused by some exotic handwriting font types, for example, "7" with a bar in the middle. Those cases up to over 5% of total data in two subsets of Chars74k are not considered in the two topology-based methods.

TABLE 4 | **Recognition rates on different tests.**

Dataset	Train size	Test size	Method	Avg. recog. rate (%)
CoinNUMS_geni	0	606	Bubble map	80
	0	606	Reeb Graph	38
	100[a]	546[a]	ANN	61
	200[a]	446[a]	ANN	63
CoinNUMS_pcgs_a	0	1,200	Bubble map	38
	0	1,200	Reeb Graph	18
	100	1,100	ANN	28
	200	1,200	ANN	30
CoinNUMS_pcgs_m	0	1,200	Bubble map	49
	0	1,200	Reeb Graph	25
	100	1,200	ANN	35
	200	1,100	ANN	42
Chars74K_EnglishImg	0	593	Bubble map	77
	0	593	Reeb Graph	72
	100	493	ANN	78
	200	393	ANN	84
Chars74K_EnglishHnd	0	550	Bubble map	79
	0	550	Reeb Graph	58
	100	450	ANN	86
	200	350	ANN	90

[a]*Synthetic images added to the classes with insufficient data.*

Otherwise, bubble map will have similar performance as ANN on both subsets of Chars74K.

Artificial neural network achieves the best results in Chars74k_EnglishImg and Chars74K_EnglishHnd. Unlike first two training free methods, the idea of neural networks is to use a huge amount of training data. Nevertheless, due to the current constraint of our data, we do not have enough data for decent data training, especially for certain classes in which we have only a few samples (see **Table 3**). For testing on each subset of CoinNUMS, we use a small training set of 100 images (10 samples per class) and 200 images (20 samples per class) randomly selected from the subsets of CoinNUMS, and the rest images are used as the test set. The average recognition rates are obtained by repeated random sub-sampling validation. For CoinNUMS_geni where the classes "3" and "4" have too few data to train, we add synthetic data that simulate digits in hollow-type font, like **Figure 10A**, into the training set to make each class have the equal training size. Of course to some databases ANN could do better if we use more training data. However, our purpose is to compare the results of different databases by using the same training size.

TABLE 5 | Avg. confusion matrix: ANN with 200 training data.

	0	1	2	3	4	5	6	7	8	9
(a) CoinNUMS_geni										
0	64%	5%	2%	6%	6%	0%	3%	10%	1%	2%
1	2%	52%	5%	5%	8%	3%	7%	13%	1%	5%
2	1%	8%	74%	2%	2%	1%	0%	7%	4%	0%
3	0%	0%	0%	25%	0%	25%	25%	0%	0%	25%
4	0%	0%	0%	0%	100%	0%	0%	0%	0%	0%
5	4%	7%	1%	4%	4%	50%	9%	8%	6%	7%
6	8%	6%	0%	1%	5%	2%	65%	5%	7%	1%
7	0%	15%	0%	8%	0%	0%	0%	78%	0%	0%
8	6%	2%	1%	7%	1%	5%	13%	2%	62%	2%
9	5%	5%	0%	2%	2%	0%	1%	4%	2%	80%
(b) CoinNUMS_pcgs_a										
0	40%	6%	6%	6%	9%	11%	11%	6%	3%	3%
1	7%	33%	6%	6%	6%	12%	6%	18%	3%	2%
2	5%	13%	21%	7%	10%	6%	9%	11%	4%	14%
3	7%	13%	8%	17%	4%	7%	8%	11%	12%	14%
4	0%	36%	0%	7%	27%	0%	27%	0%	0%	3%
5	9%	10%	4%	16%	6%	44%	2%	2%	6%	2%
6	9%	12%	6%	5%	9%	16%	17%	9%	7%	7%
7	6%	12%	8%	3%	7%	3%	5%	50%	5%	2%
8	12%	7%	16%	8%	6%	7%	8%	5%	20%	10%
9	9%	10%	16%	7%	8%	5%	8%	8%	5%	25%
(c) CoinNUMS_pcgs_m										
0	39%	9%	10%	3%	6%	9%	5%	2%	7%	10%
1	5%	47%	9%	9%	5%	6%	3%	11%	3%	3%
2	0%	7%	39%	6%	9%	6%	5%	20%	5%	4%
3	10%	7%	10%	29%	3%	7%	9%	8%	11%	7%
4	0%	26%	0%	0%	64%	0%	0%	0%	10%	0%
5	4%	6%	6%	8%	5%	50%	3%	6%	8%	4%
6	9%	6%	5%	10%	6%	8%	32%	9%	5%	10%
7	2%	7%	6%	6%	2%	3%	5%	61%	5%	4%
8	4%	6%	10%	14%	5%	4%	8%	7%	30%	11%
9	6%	4%	12%	7%	4%	9%	8%	6%	4%	41%

Classification rates of "3" and "4" in CoinNUMS_geni (in red) have no reference value since they are not trained properly due to lack of data.

To see which classes are more easily confused with others in ANN experiments, we computed the confusion matrix denoted by M_c. **Table 5** shows average confusion matrix for ANN with 20 training data per class on different subsets of the proposed database. The value at each position corresponds to the percentage of images belonging to the digit *i* and recognized as the digit *j*. Hence, 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 outside the diagonal represent false classification rate (red). For the clarity, higher classification rates are marked by colors closer to green or red, and lower classification are marked by colors closer to white. In CoinNUMS_geni, in general each class is less confused with others. It is normal that most false negative "1" and "7" are recognized as the other. The class "5" is relatively more confused than other classes due to its great variability of type fonts which cannot be largely presented by the small training data (see **Figure 13A**). The classes "3" and "4" are not considered since they have too few data and we use synthetic training data for those two classes. In CoinNUMS_pcgs_a and CoinNUMS_pcgs_m, classes are much easier get confused due to the strong presence of noise. CoinNUMS_pcgs_a is worse. Although the right class in most cases has the highest recognition rates compared to other classes, an extremely noised digit can be affected to any other classes. Except for the intrinsic confused digit shapes like "1" and "7," we observe that classes having more extremely noised data, such as those extracted from too worn coins, are worse recognized (see **Figure 13B**). If we compare CoinNUMS_pcgs_m and CoinNUMS_pcgs_a, the influence of cropping error seems to be more obvious on class "4" because an incomplete "4" with only the vertical part can be confused with "1" (see **Figure 13C**). Besides,

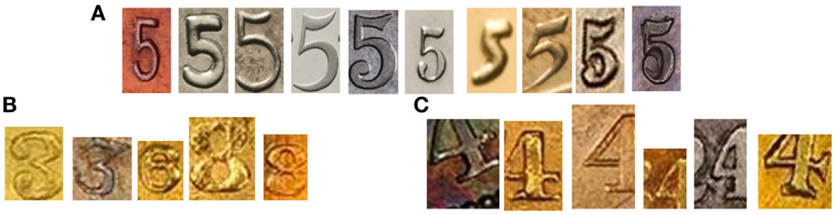


FIGURE 13 | Difficult examples for ANN in CoinNUMS: (A) type font variations of "5" (scale normalized) in CoinNUMS_geni, (B) digits cropped from too worn coins in CoinNUMS_pcgs_m, and (C) various cropping errors of "4" in CoinNUMS_pcgs_a.

the class "4" has relatively less data in CoinNUMS_pcgs_a than other classes.

Based on our experiments, the main difficulty of this challenging database is its "hard-to-segment" foreground. Besides, various noises caused by degradation of material of coins or sometimes incomplete cropping make it more difficult to get a correct foreground of digit character before recognition. If the detected contours fail to represent the correct topology of the digit, the method based on bubble map will fail but ANN still has the possibility to give the right class prediction. However, small training data with great noise can hardly represent the most distinctive features of each class. If we are able to get complete contours of relief surface, bubble map with fine-tuned parameters can be robust to border noise and variation of type fonts. For leaning-based methods, more data and better data distribution are required.

CONCLUSION AND PERSPECTIVES

A database consisting of digit characters having indistinct foreground and background has been described in this article. It is built from professional coin photographs provided by numismatic companies. We propose this database based on two motivations. First, those characters with "hard-to-segment" foreground in hollow-type font have been less studied in previous work of character recognition. Second, due to lack of effective algorithms to deal with characters in hollow-type font, legends on coins are difficult to read precisely by computer vision, which make numismatic community still relies on expensive human investigation.

Due to the availability of professional coin photographs, the current version of our database has not a huge amount of data and equal data distribution. Our future work is to continue our effort to further enlarge the database with help of numismatic companies: with the availability of more professional coin photographs in better quality, in next versions CoinNUMS will be largely increased in scale; furthermore, we aim at extending CoinNUMS little by little to a more comprehensive database, perhaps renamed as CoinLEGENDS, that will include letters, foreign characters, and short words as well. We hope CoinNUMS will become an initiative to draw attentions of researchers studying on character recognition to some complex cases which rarely happens in general situations but could be common in a specific domain. For a numismatic perspective, we hope automatic reading legends on coins will become as fast and precise as nowadays matured OCR applications.

AUTHOR NOTES

The CoinNUMS databse described in this paper was collected by the authors; the experiments were conducted by the authors.

AUTHOR CONTRIBUTIONS

All authors listed have made substantial, direct, and intellectual contribution to the work and approved it for publication.

FUNDING

This work was supported by the numismatic company GENI, and co-funded by ANRT (Convention CIFRE 2013/0807).

REFERENCES

Arandjelovic, O. (2010). "Automatic attribution of ancient Roman imperial coins," in *Computer Vision and Pattern Recognition (CVPR 2010)* (San Francisco: IEEE), 1728–1734.

Arandjelovic, O. (2012). "Reading ancient coins. Automatically identifying denarii using obverse legend seeded retrieval," in *ECCV12* (Florence: Springer), 317–330.

Ciregan, D., Meier, U., and Schmidhuber, J. (2012). "Multi-column deep neural networks for image classification." in *Computer Vision and Pattern Recogintion (CVPR 2012)* (Providence, RI: IEEE), 3642–3649.

Coustaty, M., Pareti, R., Vincent, N., and Ogier, J. M. (2011). Towards historical document indexing: extraction of drop cap letters. *Int. J. Doc. Anal. Recognit.* 14, 243–254. doi:10.1007/s10032-011-0152-x

de Campos, T. E., Babu, B. R., and Varma, M. (2009). "Character recognition in natural images," in *VISAPP*, Vol. 2, Istanbul, 273–280.

Fukumi, M., Omatu, S., Takeda, F., and Kosaka, T. (1992). Rotation-invariant neural pattern recognition system with application to coin recognition. *IEEE Trans. Neural Netw.* 3, 272–279. doi:10.1109/72.125868

Hua, X. S., Wenyin, L., and Zhang, H. J. (2004). An automatic performance evaluation protocol for video text detection algorithms. *IEEE Trans. Circuits Syst. Video Technol.* 14, 498–507. doi:10.1109/TCSVT.2004.825538

Huber, R., Ramoser, H., Mayer, K., Penz, H., and Rubik, M. (2005). Classification of coins using an eigenspace approach. *Pattern Recognit. Lett.* 26, 61–75. doi:10.1016/j.patrec.2004.09.006

Jouili, S., Coustaty, M., Tabbone, S., and Ogier, J. M. (2010). "NAVIDOMASS: structural-based approaches towards handling historical documents," in *20th International Conference on Pattern Recognition (ICPR)* (Istanbul: IEEE), 946–949.

Kampel, M., and Zaharieva, M. (2008). "Recognizing ancient coins based on local features," in *International Symposium on Visual Computing* (Berlin, Heidelberg: Springer), 11–22.

Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanov, A., Iwamura, M., et al. (2015). "Competition on robust reading," in *13th International Conference on Document Analysis and Recognition* (Tunis: IEEE), 1156–1160.

Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., i Bigorda, L. G., Mestre, S. R., et al. (2013). "Robust reading competition," in *12th International Conference on Document Analysis and Recognition* (Washington, DC: IEEE), 1484–1493.

Kavelar, A., Zambanini, S., and Kampel, M. (2012). "Word detection applied to images of ancient roman coins," in *Virtual Systems and Multimedia (VSMM 2012)* (Milan: IEEE), 577–580.

Landré, J., Morain-Nicolier, F., and Ruan, S. (2009). "Ornamental letters image classification using local dissimilarity maps," in *10th International Conference on Document Analysis and Recognition* (Barcelona: IEEE), 186–190.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 2278–2324. doi:10.1109/5.726791

Lee, S., Cho, M. S., Jung, K., and Kim, J. H. (2010). "Scene text extraction with edge constraint and text collinearity," in *20th International Conference on Pattern Recognition (ICPR)* (Istanbul: IEEE), 3983–3986.

Liu, C. L., and Fujisawa, H. (2008). "Classification and learning methods for character recognition: advances and remaining problems," in *Machine Learning in Document Analysis and Recognition* (Berlin, Heidelberg: Springer), 139–161.

Lucas, S. M., Panaretos, A., Sosa, L., Tang, A., Wong, S., and Young, R. (2003). "Robust reading competitions," in *2003 International Conference on Document Analysis and Recognition*, Vol. 2003, Edinburgh, 682.

Marti, U. V., and Bunke, H. (2002). The IAM-database: an English sentence database for offline handwriting recognition. *Int. J. Doc. Anal. Recognit.* 5, 39–46. doi:10.1007/s100320200071

Nagy, R., Dicker, A., and Meyer-Wegener, K. (2011). "NEOCR: a configurable dataset for natural image text recognition," in *International Workshop on*

Camera-Based Document Analysis and Recognition (Berlin, Heidelberg: Springer), 150–163.

Nölle, M., Jonsson, B., and Rubik, M. (2004). *Coin images Seibersdorf-Benchmark. Technical Report*. ARC Seibersdorf Research GmbH.

Nölle, M., Penz, H., Rubik, M., Mayer, K., Holländer, I., and Granec, R. (2003). "Dagobert-a new coin recognition and sorting system," in *Proceedings of the 7th International Conference on Digital Image Computing-Techniques and Applications (DICTA'03)* (Sydney, Australia).

Pan, X., Puritat, K., and Tougne, L. (2014). "A new coin segmentation and graph-based identification method for numismatic application," in *ISVC'14*, Vol. 8888 (Las Vegas: LNCS), 185–195.

Pan, X., and Tougne, L. (2016). Topology-based character recognition method for coin date detection (*World academy of science, engineering and technology, international science index 118). Int. J. Comput. Electr. Autom. Control Inf. Eng.* 10, 1725–1730.

Pan, Y., Hou, X., and Liu, C. L. (2011). A hybrid approach to detect and localize texts in natural scene images. *IEEE Trans. on Image Process* 20, 800–813. doi:10.1109/TIP.2010.2070803

Reisert, M., Ronneberger, O., and Burkhardt, H. (2007). "A fast and reliable coin recognition system," in *Joint Pattern Recognition Symposium* (Berlin, Heidelberg: Springer), 415–424.

Santosh, K. C. (2011). "Character recognition based on dtw-radon," in *International Conference on Document Analysis and Recognition (ICDAR2011)*, Beijing, 264–268.

Santosh, K. C., and Wendling, L. (2015). Character recognition based on non-linear multi-projection profiles measure. *Front. Comput. Sci.* 9:678–690. doi:10.1007/s11704-015-3400-2

Shahab, A., Shafait, F., and Dengel, A. (2011). "Robust reading competition challenge 2: reading text in scene images," in *2011 International Conference on Document Analysis and Recognition* (Beijing: IEEE), 1491–1496.

Sun, L., Huo, Q., Jia, W., and Chen, K. (2015). A robust approach for text detection from natural scene images. *Pattern Recognit.* 48, 2906–2920. doi:10.1016/j.patcog.2015.04.002

Thome, N., Vacavant, A., Robinault, L., and Miguet, S. (2011). A cognitive and video-based approach for multinational license plate recognition. *Mach. Vis. Appl.* 22, 389–407. doi:10.1007/s00138-010-0246-3

Trier, ØD., Jain, A. K., and Taxt, T. (1996). Feature extraction methods for character recognition – a survey. *Pattern Recognit.* 29, 641–662. doi:10.1016/0031-3203(95)00118-2

Van Der Maaten, L. J., and Poon, P. (2006). "Coin-o-matic: a fast system for reliable coin classification," in *Proceedings of the Muscle CIS Coin Competition Workshop* (Berlin, Germany), 7–18.

Viard-Gaudin, C., Lallican, P. M., Knerr, S., and Binter, P. (1999). "The ireste on/off (ironoff) dual handwriting database," in *ICDAR'99. Proceedings of the Fifth International Conference on Document Analysis and Recognition* (Bangalore: IEEE), 455–458.

Wang, K., and Belongie, S. (2010). "Word spotting in the wild," in *European Conference on Computer Vision* (Berlin, Heidelberg: Springer), 591–604.

Ye, Q., and Doermann, D. (2015). Text detection and recognition in imagery: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.* 37, 1480–1500. doi:10.1109/TPAMI.2014.2366765

Zambanini, S., and Kampel, M. (2013). "Improving ancient roman coin classification by fusing exemplar-based classification and legend recognition," in *New Trends in Image Analysis and Processing-ICIAP 2013* (Naples: Springer), 149–158.

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.