



Segmentation system and its evaluation for gray scale coin documents

Vortana Say, Mickaël Coustaty, Joseph Chazalon, Jean-Marc Ogier

► To cite this version:

Vortana Say, Mickaël Coustaty, Joseph Chazalon, Jean-Marc Ogier. Segmentation system and its evaluation for gray scale coin documents. 4th International Conference on Image Processing Theory, Tools and Applications (IPTA), 2014, Oct 2014, Paris, France. pp.1-6, 10.1109/IPTA.2014.7001960 . hal-01247954

HAL Id: hal-01247954

<https://hal.science/hal-01247954>

Submitted on 19 Sep 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Segmentation System and its Evaluation for Gray Scale Coin Documents

Vortana SAY¹, Mickaël COUSTATY¹, Jean-Christophe BURIE¹, Joseph CHAZALON¹ and Jean-Marc OGIER¹

¹L3i, La Rochelle, France,

E-mail: {vortana.say, mickael.coustaty, jean-christophe.burie, joseph.chazalon, jean-marc.ogier}@univ-lr.fr

Abstract—The numismatics community undergoes an important change toward a digital archiving of coin trades. Historic transactions of coins are tracked; many catalogues and the digitization of such content is of great interest for augmenting experts work in estimating a coin value and preventing illegal sales. In order to recognize and index coin pictures in transactions archives, a segmentation step is required. We propose a system which enables the automatic segmentation of coins and labels in the documents. As opposed to state of the art approaches, our system can extract coin images of any shape and handle multiple coins per image, as well as text labels. We also present in detail the evaluation metric, pixel correspondence graph, and the adaptation of the metric that we used to evaluate the performance of the proposed system.

The evaluation of the proposed system on seven books (230 pages) shows promising results: over 99% of coins and 90% of labels are correctly segmented.

Keywords—image segmentation, connected components, pixel correspondence graph, numismatic, coins.

I. INTRODUCTION

Coin authentication, which refers to estimate a coin's value and to prevent illicit coin trade, is a major concern of the numismatics markets [1]. To identify and recognize a coin, numismatics experts involve a deep knowledge of ancient and new coins, and rely on many official sales catalogues which have vital information such as nature of coins, prices and so on. These documents serve as references for experts to analyze and compare a new coin to the known ones. There are millions of coin images stored in such catalogues. These books are fragile, and keeping track of coin sales history is complex. Recently, numismatic has shifted from manually study by coin experts to hi-tech expertise in which some automated systems are used to assist the experts during their work [1].

Classification and recognition are major challenges for numismatic experts. While many works in Document Analysis and Recognition propose to extract, recognize and index document image contents, very few have discussed the segmentation processes that could be applied on real digitized documents composed of coins and other metadata. Segmentation step plays a fundamental role in a large number of image processing and computer vision applications and in the case of coin images, it has a serious impact on the later processes such as indexing, classification and recognition [2]: the outer border of coin images usually contains most of the information used of its recognition.

This paper has two contributions. We first present an adaptive segmentation system which automatically extracts and separates coins and labels images from numismatics catalogues. Then, we propose a full evaluation system which is generic enough to be used in other domains or application fields.

We validated the proposed system with an adapted version of a state of the art evaluation method, and used it to evaluate efficiently the extracted contents of seven catalogues (230 pages) containing various coin shapes, mixed image and text content, as well as difficult configurations with touching elements. Figure 1 shows a few pages

from the books, and a detail of a possible layout configuration for both sides of a coin.

This paper is organized as follows: section 2 describes the state of the art of the segmentation methods. The proposed algorithms will be described in section 3. The experiments protocol and the evaluation process are presented in section 4 followed by result and evaluations in section 5. Finally, we give a conclusion and ideas for further improvement in section 6.

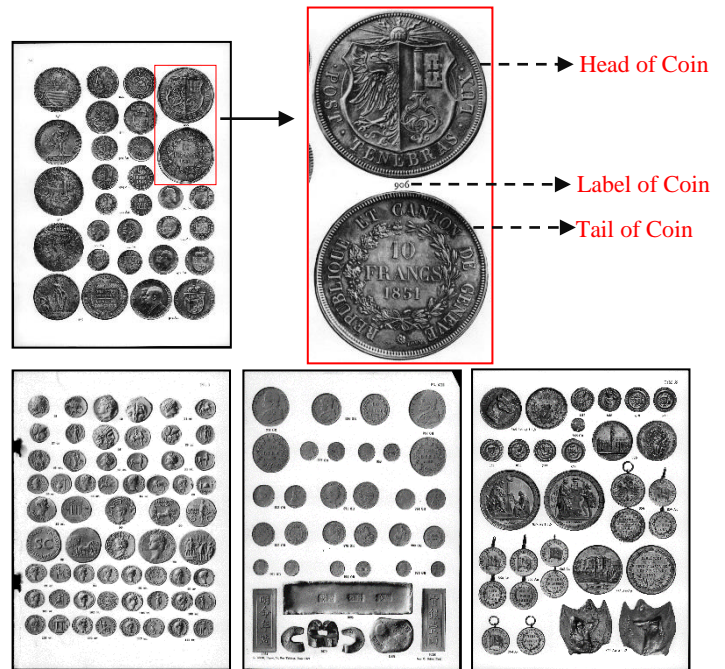


Fig. 1. Different samples of coin documents in the dataset

II. RELATED WORK

Many segmentation methods for document images can be found in the literature, divided into three categories: top-down approaches, bottom-up approaches and hybrid approaches [3].

Projection profiles [4], X-Y cuts [5] or white streams [6] are considered as top-down approaches. These approaches are fast and useful only if some knowledge about documents layout is available. In the specific case of our dataset, the unpredictability of document layout prevent from using them.

The second category, bottom-up approaches, mainly relies on the use of unitary elements to build bigger ones. These unitary elements can be the connected components [20], text lines or text blocks. These methods are more flexible, but they are generally slower than the top-down approaches [3]. The common techniques used in bottom-approaches consists of run-length smearing [13], region growing [7] [8] and mathematical morphological [9].

The last category, the hybrid approaches, attempts to combine the robustness of bottom up approaches with the fast speed of the top-down approaches [3]. They are based often on texture-based approaches like Gabor filters, multi-scale wavelet analysis [10], or fractal signatures [11].

Concerning coin image segmentation, existing works [13 - 17] have been proposed, but these techniques do not address the combination of ancient coin images with multiple coins per page. However, they provide interesting insights on coin segmentation issues.

In [14], a gray levels thresholding process was used to segment the coins in the assumption that the coins themselves are brighter than their background. A more advanced separation of coins and background in used in [13]: the combination of local entropy and local range of gray values of an image were used to segment the coins. Furthermore, coins whose forms are close to circle are extracted. A form factor is used.

Shape information is also used in [15]: Generalized Hough Transform (GHT) and shape based segmentation were combined to detect the coins, with the assumption that only one coin is present in an image. In [16-17], the approach is based on edge detection. Hough Transform was used for modern coins; while a modified version of Hough Transform with a measure of compactness was used for ancient coins.

While approaches in [13-17] cannot handle multiple images per page, the way they characterize coin images thanks to texture and shape information is highly relevant for our scenario. However, a major challenge to address is the risk of damaging coin borders in the segmentation process, as such approaches do. For instance, in [13], out of ten evaluated images, the approach produced border error of 0.29% for the best segmentation result, while border error of 15.81% was produced for the worst segmentation result. In our dataset, the shape of the ancient coins can be far off circle shape (see Fig. 1).

Thus, we propose an adaptive segmentation system capable of handling a dataset with a large amount of coins and text labels, while minimizing border segmentation error.

III. PROPOSED ALGORITHM

In order to be able to deal with the specificities of our images, and taking into account the advantages and drawbacks of the existing approaches, we proposed a system composed of four modules, as illustrated in Fig. 2: pre-processing to enhance the quality and gray level conversion of the original image, coins and labels detection, coins and labels extraction and output.

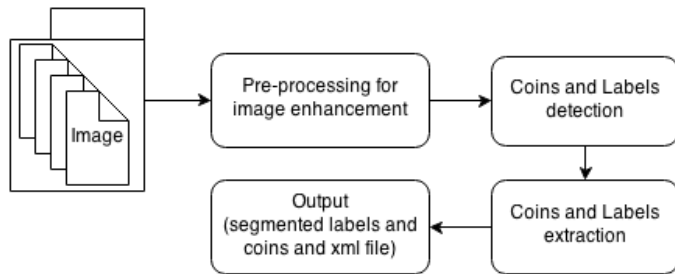


Fig. 2. Workflow of the adaptive segmentation system

A. Pre-processing

Grayscale images are the targets of the system due to its simplicity (single channel) and data reduction (reduction in time processing). Thus, the original image is converted to grayscale image. Then Otsu's method [18] is used to convert grayscale image to binary ones [12]. The idea is to find the threshold that minimizes the weighted and within class variance [12]. The objective of Otsu's method is to select a threshold automatically from a gray level histogram so that foreground and background could be distinguished [18]. This method is well known used and fast for simple documents. We use a global thresholding process instead of a local one because the result of global thresholding is acceptable and the process is faster than local thresholding.

After binarizing the images, some pixels along the border of the coins are disconnected; thus we apply a classical morphological operation, dilation [19], with small size of structure element in order to add pixels to the boundaries of the coins. The reason for using small size of structure element is to avoid connection of the nearest coins. The final operations of the pre-processing is filling the hole so that we could reduce the number of connected components candidates.

B. Coins and Labels detection

In this process we aim to identify coins and labels in the image documents. We use connected components algorithm [20] to label each blob. However, both coins and labels need to be divided; thus, K-mean clustering [21] is used to distinguish coin blobs and label blobs. K-mean clustering is fast, robust, much simpler to implement, and optimal solution for the system. Furthermore, we know the number of cluster beforehand, $k = 2$, because we want to clusters two main clusters, coins and labels.

We use the properties of the connected components such as area, height, width in order to identify the nature of coins and labels; the properties of coins and labels are used to group together all the similar properties into classes, which are coin and label.

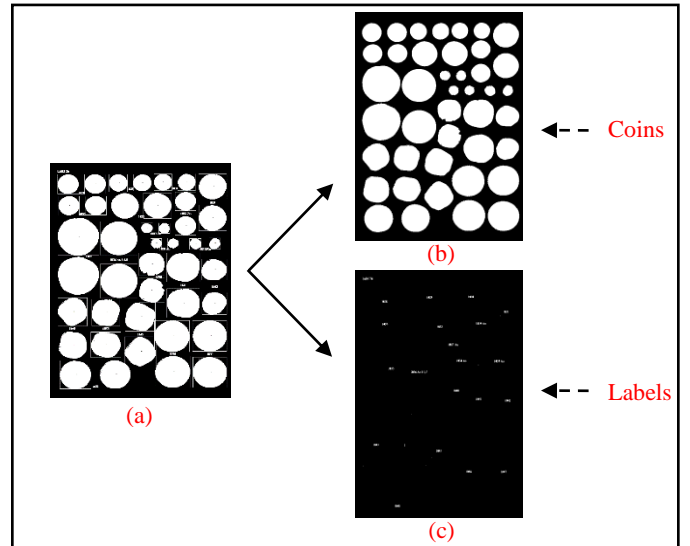


Fig. 3. Coins and labels detections and separations

In Fig. 3, image (a) is the results of the pre-processing module. By using a connected components algorithm, we can extract number of shapes that can be classified. Image (b) and (c) presents the results of the classification process with respectively the coins and the labels.

C. Coins and Labels extraction

After the detection process, each character of labels is identified. However, they are not connected, so we use the Running Length Smoothing Algorithm (RLSA) [22] to connect the nearest characters in order to form each label. Since the labels of the dataset are oriented horizontally, we apply the RLSA in this direction.

After this detection process, the coins can be extracted from the image since each blob corresponds to a coin. However, some coins have unwanted background so that we have a post processing to clean the background of the coin, and ensure that the segmented images are free from noise when sent to later processes such as classification and indexation.

The procedure for the cleaning process is:

1. Find the center of the segmented image
2. Use the connected components algorithm [20] to find the blobs of each objects in the segmented image
3. Compare the distance between center each blob to the center point of the segmented image, and keep only the blob with the nearest distance to the center point

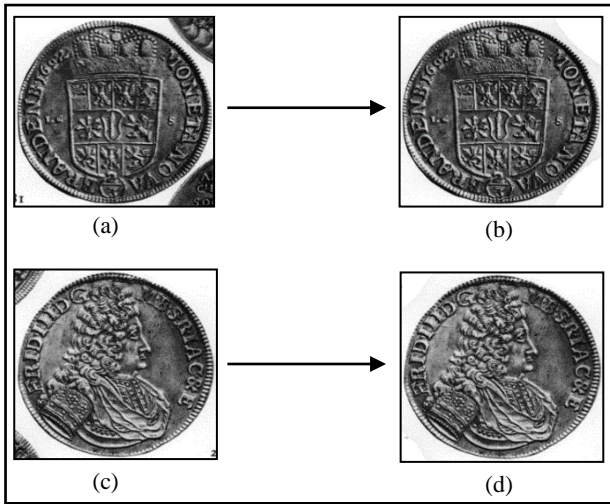


Fig. 4. Post processing for segmented coins

In Fig. 4, image (a) and (c) present the results obtained after the extraction process. As the image may contain not only the coin but also some small part of other coins or labels, we use a cleaning process to deal with this problem. After applying the cleaning phase, we can obtain the results presented in Fig. 4 in which image (b) and (d) consist only the coin.

D. Output

The main goal of this module is to propose a dedicated and as useful as possible storing of the segmented images in order to be able to link the further information for later processes, such as classification, indexing or recognition. In contrast with the mentioned work in state of the art, the catalogues contain not only coins but they also have labels. In an image document of catalogue, one coin consists of two sides, head and tail; these two sides are connected by a unique label. Thus, we need to save some properties such as width, height coordinate x and y, of the coins and labels in the digital documents. In this system, we use XML file as a mean to store those properties for each document image.

IV. EXPERIMENTAL PROTOCOL

We conducted experiments on real dataset which consists of seven scanned books. The dataset are the sample which represent the large amount of existing books. First, we present the dataset that we are working on. Second, we explain how the ground truth is built. Then the choice of the evaluation metric and its adaptation are justified. Finally, results and evaluation of the proposed system is analyzed.

A. Presentation of the dataset

Some examples of the dataset used for this work were presented in Fig. 1. It is composed of books that consist of coin images and text labels, or sometimes more complex descriptions of the coins.

The dataset contains three main challenges for a digitization process. First, page layout is not constant, as is presents in some esthetical way a composition of historical coins images (with different shapes) and metadata that can be referred as the labels (a number near the coin image). Second, each coin is represented by both its head and its tail, which are linked by the label and a semantic rule; the position of a coins are generally left (head)-right (tail) or top (head)-down (tail), and they are between a unique label. Third, some coins are located too near to each other and that the segmentation system needs to take these criteria in to considerations.

B. Building Ground-truth

Ground-truth corresponds to correctly segmented relevant data (coin and label images in our case), which is manually or user-guided determined, in a given set of data.

In order to build our ground-truth, we use a flexible ground truth editor, Gotham (<http://l3i.univ-larochelle.fr/gotham.html>). The editor offers a powerful and friendly user platform to build segmentation ground truth. Seven digitized books at 300 DPI were used, and they are composed of 230 pages.

We then build out the ground-truth of these data and we identified 6414 coins and 3600 labels. Due to the large amount of segmentation data in the dataset, a semi-automatic process was adopted, where the user only have to confirm the proposed extractions and to add the missing coins or labels. Fortunately, the proposed system performs reliably on this task.

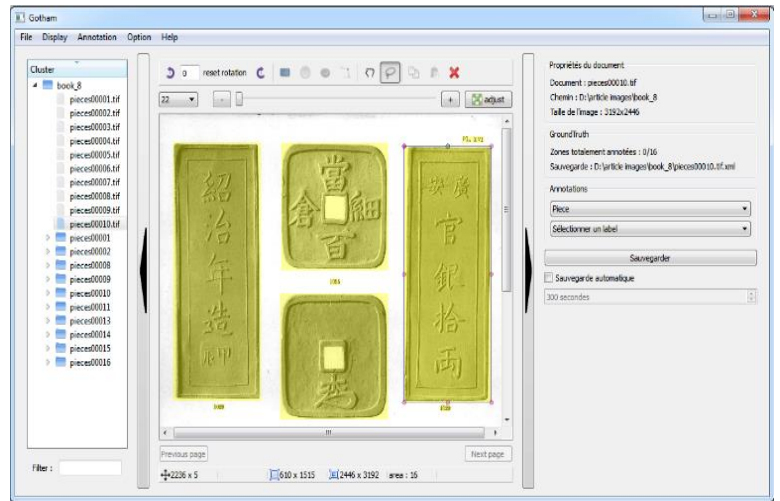


Fig. 5. Gotham editor

Figure 5 shows bounding boxes of coins and labels are carefully marked for ground truth. Some unmarked texts are not considered as labels.

Gotham use XML files to store all the information of the segmentation data, thus, we could use the information of the XML of our proposed system in order to generate the semi-automatic mechanism. The procedure is as follows:

- We use Gotham to generate candidate of segmentation data, and we verify by carefully inspect for each field if the segmentation is acceptable.
- If the segmented field is not acceptable, the segmentation can be touch up or re-create manually.

C. Evaluation Metric

In order to compare the quality of hypothesis segmentation (i.e. the segmented image from the proposed system) H and the ground truth G, we adapt the notion of pixel correspondence graph [23] and the measurements of [24]. Using the availability of the pixel correspondence of G and H, we could compute and construct the bipartite graph [23]. Furthermore, this metric provides not only the quality comparison, but it also indicates different types of errors as stated in [24]. This evaluation metric and its adaptation proposed in this paper are generic to be used to evaluate others segmentation results.

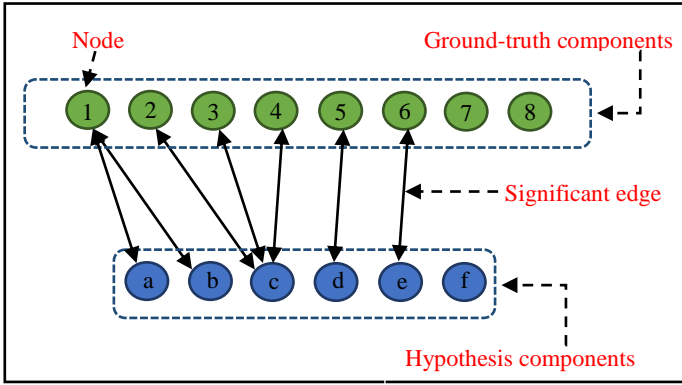


Fig. 6. The pixel correspondence graph

Figure 6 illustrates the concept of “correspondence graph” using an example in order to explain how this metric could be used to evaluate the results. Components of the ground truth and hypothesis are represented as nodes of the graph. When the components of the ground-truth and hypothesis match, we create an edge (i.e. significant edge) between the matched-components.

To simplify, the objective of the pixel correspondence graph for the needs of the proposed segmentation system is to define the metric in order to determine the relationship between the ground-truth segmentation G and the hypothesis segmentation H. Then using these relationships we could find the following indicators proposed in [24].

- **Total of correct segmentations (T_c):** total amount of exact match (one-to-one) between the ground-truth components G and the hypothesis segmentation components H.
- **Total of over segmentations (T_o):** minimization between the total amount of significant edges (matching) that ground truth components G have and the number of ground truth components which have at least one significant edge.
- **Total of under segmentations (T_u):** minimization between the total number of significant edges that hypothesis segmentation components have and the number of hypothesis segmentation components which have at least one significant.
- **Over segmented Components (C_o):** number of ground truth components G which have more than one significant edge.

- **Under segmented Components (C_u):** number of hypothesis segmentation components H which have more than one significant edge.
- **Missed Components (C_m):** number of ground truth components G that did not match any foreground component in the hypothesized segmentation H (i.e. the ground-truth components G do not have any significant edge to hypothesis segmentation H).
- **False alarms (C_f):** number of components in the hypothesized segmentation that did not match any components in the ground truth segmentation.

In order to calculate the indicators mentioned above, the associations between components of hypothesis (H) and components of ground truth (G) need to be defined; there are two steps to define the associations. The first step is to calculate the overlapping areas between components of H and components of G. After that in the second step, the associations are determined using the defined conditions.

1) Calculate overlapping Areas $W_{(h_m, g_n)}$

To facilitate the understanding of the pixel correspondence graph, we represent the set of $H = (h_1, h_2, h_3, \dots, h_m)$ represents the coins and labels in the hypothesis segmentation, while the set of $G = (g_1, g_2, g_3, \dots, g_n)$ represents the ground truth. Between each image element from ground truth segmentation (G) and hypothesis segmentation (H), we calculate the intersection values between the two sets of elements:

$$W_{(h_m, g_n)} = \text{Area}(h_m \cap g_n)$$

$$\begin{cases} \text{Area}(h_m) = \text{Width}(h_m) \times \text{Height}(h_m) \\ \text{Area}(g_n) = \text{Width}(g_n) \times \text{Height}(g_n) \end{cases}$$

Table 1. The intersection values between ground truth components and hypothesis components

	Hypothesis components					
	h_1	h_2	h_3	h_m	
Ground-truth components	g_1	w_{11}	w_{12}	w_{13}		w_{1m} → t_{g_1}
	g_2	w_{21}	w_{22}	w_{23}		w_{2m} → t_{g_2}
	g_3	w_{31}	w_{32}	w_{33}		w_{3m} → t_{g_3}

	g_n	w_{n1}	w_{n2}	w_{n3}		w_{nm} → t_{g_n}
		↓ t_{h_1}	↓ t_{h_2}	↓ t_{h_3}		↓ t_{h_m}

After calculated the intersection values between ground truth components and hypothesis as shown in Table 1, total weight of a component (t) is calculated as a sum of the overlapping area between that component and others components. The total weight (t) plays vital role in the determining the association between ground truth components and hypothesis components; the role of total weight is to make sure that the small overlapping area occurred by components will not be considered as the association with the parent component (see Fig. 7).

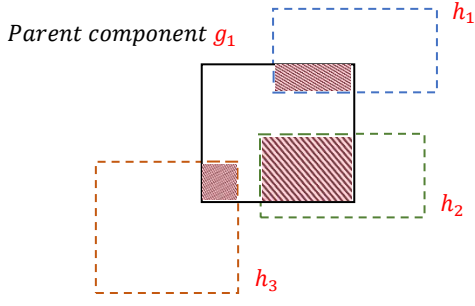


Fig. 7. Calculating total weight of component g_1

Figure 7 shows that there are three components, h_1 , h_2 and h_3 which are overlapped with parent component g_1 . Thus the total weight of component g_1 :

$$t_{g_1} = (g_1 \cap h_1) + (g_1 \cap h_2) + (g_1 \cap h_3)$$

Thus:

$$t_{g_i} = \sum_{j=1}^m w_{ij}$$

$$t_{h_j} = \sum_{i=1}^n w_{ij}$$

$$\begin{cases} m : \text{total number of hypothesis components} \\ n : \text{total number of ground truth components} \end{cases}$$

2) Determine the association value b_{nm}

Table 2. The association values between ground truth components and hypothesis components

		Hypothesis components				
		h_1	h_2	h_3	h_m
Ground-truth components	g_1	b_{11}	b_{12}	b_{13}		b_{1m}
	g_2	b_{21}	b_{22}	b_{23}		b_{2m}
	g_3	b_{31}	b_{32}	b_{33}		b_{3m}
	\cdot	\cdot	\cdot	\cdot	\cdot	\cdot
	g_n	b_{n1}	b_{n2}	b_{n3}		b_{nm}

As shown in Table 2, the pixel correspondence graph indicates the vice versa relationship between the ground truth segmentation and hypothesis segmentation. Thus, determination of value b_{nm} is done from ground truth and hypothesis perspective. Unlike [24], we deploy two steps conditions in each perspective, and If two blobs satisfied either condition (1) and (2) or condition (3) and (4), the two blobs are match (i.e. $b_{nm} = 1$) otherwise the value $b_{nm} = 0$.

Hypothesis perspective:

$$\frac{W(h_m, g_n)}{t_{h_m}} > \text{relative_threshold} \quad (1)$$

$$\frac{W(h_m, g_n)}{\text{Area}(h_m)} > \text{absolute_threshold} \quad (2)$$

Ground truth perspective:

$$\frac{W(h_m, g_n)}{t_{g_n}} > \text{relative_threshold} \quad (3)$$

$$\frac{W(h_m, g_n)}{\text{Area}(g_n)} > \text{absolute_threshold} \quad (4)$$

For the purpose of this work, the `relative_threshold` and `absolute_threshold` have been respectively set to 0.1 and 0.8. The condition (1) and (3) test if the fraction of pixels overlapping is bigger than 10 pixel. However, this condition alone cannot adapt to our system because there are possibilities that small part of two coins overlap due to the proximity of the elements. Thus, we use the condition (2) and (4) which indicate that if the overlapping areas is more than 80 percent than the two elements are considered as matched.

V. RESULT AND EVALUATION

Due to the different nature between coins and labels, we conduct the evaluation on the results of segmented images separately. We compare the segmented image coins resulting from the proposed system, hypothesis components, and the image coins produced by ground truth components. We compare the segmented image coins resulted from the proposed system, hypothesis components, and the image labels produced by ground truth components.

Table 3. The evaluation results with different measurements for the used dataset of 7 digitized books

	# region produced in ground truth	# region produced in hypothesis	T_c	T_o	T_u	C_o	C_u	C_m	C_r
Coins	6414	6429	6378 (99.43%)	15	15	13	9	0	14
Labels	3600	5014	3265 (90.69%)	1	0	1	0	334	1747

According to the Table 3, the segmentation system can achieve a satisfying number of total correct segmentations. 6378 coins out of 6414 coins (99.43%) are correctly extracted (see Fig. 9). However, we also indicate some errors occurred. These errors happen when there are connected coins in the original image, too much noises and unnecessary texts that are not considered as labels as shown in the Fig. 8 “Paris (IX°)” and “54, Rue T”, for instance, are not considered as labels; however, they are segmented from the dataset.

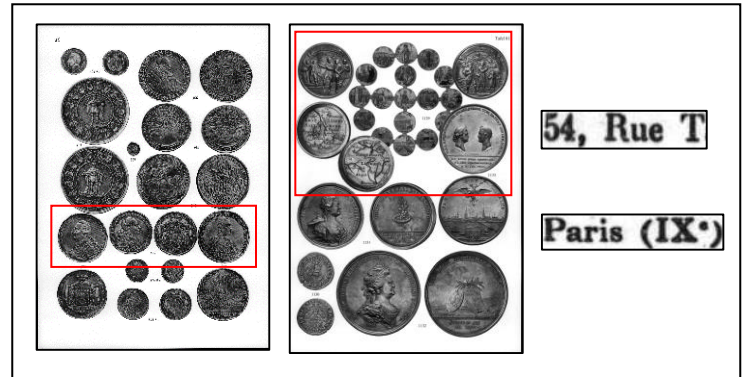


Fig. 8. The sample of connected coins and non-label texts

