2011 International Conference on Document Analysis and Recognition

A contour-based method for logo detection

The Anh Pham, Mathieu Delalandre and Sabine Barrat
Laboratoire d'Informatique
64, Avenue Jean Portalis, 37200 Tours - France.
the-anh.pham@etu.univ-tours.fr, {mathieu.delalandre, sabine.barrat}@univ-tours.fr

***Abstract*—This paper presents a new approach for logo detection exploiting contour based features. At first stage, preprocessing, contour detection and line segmentation are done. These processes result in set of Outer Contour Strings (OCSs) describing each graphics and text parts of the documents. Then, the logo detection problem is defined as a region scoring problem. Two types of features, coarse and finer ones, are computed from each OCS. Coarse features catch graphical and domain information about OCSs, such as logo positions and aspect ratios. Finer features characterize the contour regions using a gradient based representation. Using these features, we employ regression fitting to score how likely an OCS takes part of a logo region. A final step of correction helps with the wrong segmentation cases. We present experiments done on the Tobacco-800 dataset, and compare our results with the literature. We obtain interesting results compared to the best systems.**

***Keywords*-logo detection, contour detection, regression fitting**

I. Introduction

The Document Image Analysis and Recognition (DIAR) field is of high importance nowadays in the information system of companies and institutions. It has been concerned by intensive researches since beginning 90's, resulting in a large scientific literature and several commercial applications. Most of this work has been focused on the analysis of text content in document images, using some Optical Character Recognition (OCR) techniques. However, beside text documents also contain graphical information such as figures, diagrams, logos, etc. Text-based tools cannot access this graphical information correctly, and miss then important content of documents.

In this paper, we are interested with the automatic processing of logo entities in documents. The processing of logos can support traditional OCR methodologies in several ways. In a context of mass-digitalization, logo processing can offer an efficient way of document categorization. It can also help in precise document classification, by mixing text and logo recognition methods. It can also provide a good priori knowledge about organization of documents, by identifying some initial document models and then supporting the stages of page segmentation and OCR.

In the earlier works about logo processing [14], research has been focused on logo recognition only. In these works, it was assumed that the segmentation of the logos has been done by a previous stage of page segmentation. Thereafter, several works have been proposed on the problem of logo detection [3], [4], [5], [9], [10], [13]. Such examples of research are complimentary regarding their features extraction and detection stages.

The extracted features aim at discriminating between logo and text parts of a document. They could be either domain or graphics based. In [14], [9], the authors proposed to exploit domain-based features such as logo positions and aspect ratios (i.e. text vs. logo sizes). Based on their experiments, these features look simple to extract and efficient for a first level of text/logo discrimination. The system presented in [9] used similar features, to drive an extension process of a rectangle and perform initial segmentation of logos. Geometric features are used in [10], [13] to describe the segmented regions (e.g. connected components, XY cuts) such as surface, orientation, width and height of bounding box, contour length, pixel density, etc. In [5], the author employs a complementary feature which is compactness. In all the cases, the key goal is to discriminate text from logo parts by identifying some *rough* graphics properties. Only work of [4] considered more complex descriptors such as SIFT and SC, based on a previous key-point detection step.

Final detection of logos is ensured by various techniques coming from the classification, spotting and information retrieval fields. The logo detection task in [14] was shifted to a n-classes classification problem. These classes represent standard positions of logos in documents, obtained following training and clustering steps from the domain based features. The authors in [4] applied the bag of words concept to the logo detection problem by joining all the feature vectors of the training set and the codeword dictionary is obtained. Given a query document, its feature vectors are matched against the codeword of the dictionary to perform the retrieval. In [10], a complex process of detection was presented based on geometrical matching. Anchor lines are computed from each connected components. These lines are used to reconstruct logo prototypes and conduct the final verification exploiting geometric features extracted previously.

This paper proposes a new system for logo detection. Our approach relies on conclusions given in [14], [9] about importance of the use of domain and discriminative features for logo detection. We combine two feature levels in our system, the coarse and finer ones. At the coarse level, logo candidate regions are detected from a domain and graphical

1520-5363/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDAR.2011.150

points of view. Our finer features are gradient-based computed locally. Our motivation to employ finer features is driven by the graphical complexity of logo entities, making difficult their segmentation/recognition with geometric & standard features only. At the two levels, our features are extracted following a segmentation stage based on outer contour detection and line segmentation techniques. This process results in a set of Outer Contour Strings (OCSs) which represent different text lines as well as graphic objects in the document. In a final step, a process of logo location correcting is presented to refine the results.

The rest of this paper is organized as follows. In section II, we present the outline of our approach. Section III presents our first block consisting of preprocessing and segmentation stages. Section IV describes our second block including feature extraction, training, logo detection and logos' localization correcting. We present the experiment results in section V and give the conclusions in section VI.

II. Outline of the approach

In this paper, we deal with the problem of logo detection in real and complete documents. Our approach includes six main stages as described in Fig. 1. At first, we make some pre-processings for skew correction, morphological dilation and binarization. Our segmentation stage is based on a standard contour detection and line segmentation techniques. This process results in the set of OCSs describing each graphics and text parts of the document. Then, our logo detection approach is defined as a region (i.e. our OCS) scoring problem. Two types of features are computed for each OCS. Coarse features catch graphical information and domain about OCSs, such as OCSs' position and size. Finer features characterize contour regions using a gradient based representation. Using these features, we employ a regression fitting technique to score how likely an OCS takes part of a logo region. In a final stage, a process of region merging is carried out to deal with wrong segmentation cases. At all stages, the different processes used in our system are driven by an initial setting and training.

III. Preprocessing and segmentation

To support our features extraction and detection steps, we perform at first some pre-processings that are skew correction, morphological dilation and binarization. Skew correction works with grayscale images by using the method of [11] based on the Hough Transform (HT), as HT is a popular approach for such a distortion. To improve connectivity of contours, we apply a derivation of the classical dilation operator so that it could be selft-adapted to grayscale images and to local image structure rather than depending on a specific structured element. Binarization is a mandatory preprocessing for our contour detection step. We employ the Otsu's medthod [7] as it is well adapted to the bilevel

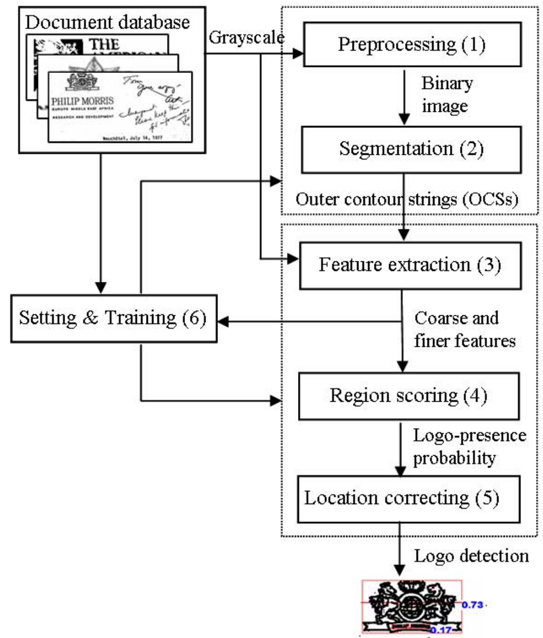

Figure 1. The flow work of our approach

thresholding problem. Fig.2(a) presents an original image and Fig.2(b) is the result image of the preprocessing stage.

Our segmentation is done with a contour detection using the Black's method [6]. It is based on a line following algorithm, and then well adapted to extract and chain outer contours. We apply then the line segmentation method of [8] to group the outer contours together and create the Outer Contour Strings (OCSs). An OCS is a sequence of outer contours that are located side by side from left to right. Fig.2(c) presents all the OCSs detected for the original image in Fig.2(a).

IV. Feature extraction and logo detection

A. Feature extraction

Starting from the segmentation results, we extract two layers of features from each OCS corresponding to the coarse and finer levels. Coarse features catch graphical information and domain about OCSs, such as OCSs' position and size. Finer features characterize contour regions (i.e. OCSs) using a gradient based representation and are computed directly from the gray-level images using the outer contours' localization information. These two levels of features constitute a vector of dimensions $4+k$ bins for each OCS, where k is the number of finer features. We will detail both of them in the rest of this section.

At the coarse level, logo candidate regions are detected from a domain and graphical points of view. We follow here the conclusions given in [3] about importance of logo position and aspect ratio as features for detection. Four useful features at the coarse level are identified. Two first

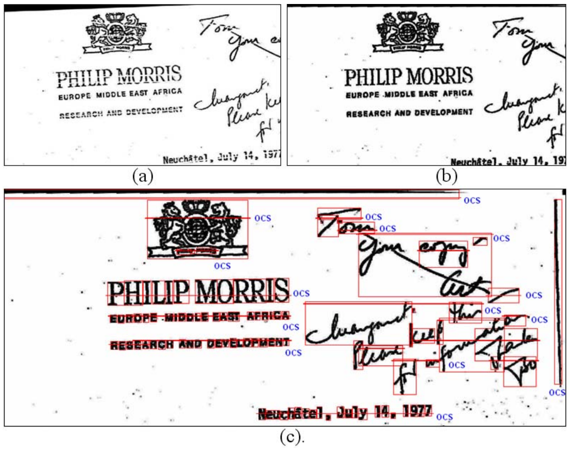

Figure 2. (a) An original image, (b) The result image after preprocessing, (c) The outer contour strings detected for the image in (a).

ones are the mean length and the standard deviation of length of the OCS. To make two those features scale-invariant, the outer contours' length of the OCS are normalized to unit length before computing two this features.

The third feature takes the position of the OCS's bounding box into account. This is based on the observation that logos appear at fixed positions in documents.

The fourth feature is the number of outer contours of each OCS. This feature is useful to discriminate the text from the graphic parts since the textlines are composed most of the time of a large number of outer contours comparing to the logos. In addition, we apply some selection rules to ensure a robustness of the feature to false positive detection. Firstly, only the including outer contours (i.e. with a bounding box not included by any other ones) are selected. Secondly, the number of outer contours of each OCS is weighted by the ratio of the bigger one and the smaller one of the width and the height of the OCS's bounding box. This is very useful to discriminate logo parts from full straight line in the documents.

Our finer features are computed locally as gradient from grayscale images. Our motivation to employ local features is driven by the graphical complexity of logo entities, and the segmentation errors that could appear during the contour detection and the line segmentation stage. Our features are computed at two levels, the outer contour level and the OCS one using a correlation analysis. Let $C_1, C_2, \ldots, C_n$ are the outer contours of some OCS. For each C_h $(h = 1, \ldots, n)$, we compute the magnitude gradient and orientation gradient for every pixel *I(x,y)* within the C_h's bounding box to build up our finer features. To this end, the orientation invariant is assumed based on the skew correction stage. The orientations are sampled into *m* bins, with *m* that can be fixed during the training step (see section *D*). After that, we build up an orientation histogram which captures the main structure of the C_h. The histogram vector is normalized and we do that for all other outer contours $(C_1, C_2, \ldots, C_n)$ to obtain the matrix *A* which is the descriptor matrix of the OCS. As the A matrix consists of *n* rows and *m* columns (*n* varied for different OCSs), it is necessary to normalize the descriptor matrix so that the feature vector of every OCS has the same number of dimensions. We use the covariance matrix from the *A* descriptor matrix to build up the feature vector. Let G is the covariance matrix of size $m \times m$ of *A*. As G is a symmetric matrix, we select only the half of G (including the diagonal line) to constitute a final feature vector of size $k = (m + 1)m/2$.

B. Region scoring

Our logo detection stage is addressed through a probability scheme, to extract confidence rates of how likely an OCS correspond to a logo or a text part. We have shifted this issue to a regression fitting by using an approach based on machine learning. For this purpose, Gentle Boost [2] is a common choice since it is known as one of the best *out of the box* supervised regression techniques. Gentle Boost combines the advantages (e.g. dealing with mixed and un-normalized data types and missing feature) from many decision trees to make final decision. All OCSs are detected and then their feature vectors are computed. The OCSs that contain logo are labeled one and the others are labeled negative one. Following is the main work flow of the training process using Gentle Boost:

- Given T examples (x_i, y_i) where x_i is a feature vector of an OCS_i and $y_i = \{1, -1\}$ with $i = 1, \ldots, T$.
- Start with weights: $w_i = \frac{1}{T}$ with $i = 1, \ldots, T$.
- Repeat for $m = 1, \ldots, M$ (M is the number of decision trees):
  - Normalize the weights w_i to unit length
  - Fit the regression function $f_m(x)$ by weighted least squares of y_i to x_i with weights w_i.
  - Update the weights: $w_i \longleftarrow w_i e^{-y_i f_m(x_i)}$
- Output the regression value:
  - Compute: $F(x) \longleftarrow \frac{1}{M} \sum_{m=1}^{M} f_m(x)$
  - Convert $F(x)$ to a value between zero and one: $F(x) \longleftarrow (F(x) + 1)/2$

The training stage results in a set of optimal decision trees so that given an input feature vector of some OCS, they output a score of how likely the OCS represents a logo part.

C. Logo's localization Correcting

In desirable conditions, we obtain single OCS for a logo in document image. However, this situation rarely exists due to the segmentation errors resulting of the noises and the scanning process. In practice, the complete logo can be segmented into several parts. Therefore, it is reasonable to perform a process of correcting the logo's localization. To do it, we group the OCSs using some similarity (i.e. probability scores) and neighboring criteria:

Table I
PARAMETERS AND MODEL CONCERNED BY THE TRAINING

Stage	Parameter(s)	Description
Pre-processing	*se*	Size of the structured element used by the dilatation operator
Features extraction	*m*	The number of used bins for the finer features
Region scoring	*F(x)*	The regression function
Localization correcting	*prob_thre*	The threshold on probability scores
Localization correcting	*eucli_thre*	The threshold on Euclidean distance for OCSs merging

- Given N pairs $\{OCS_i, lpp_i\}$ where lpp_i is the probability that OCS_i contains a logo and $i = 1, \ldots, N$.
- Select the OCSs which are higher than a threshold (*prob_thre*) set from the training step. This threshold corresponds to the minimum probability score computed from the logo OCSs of the training set.
- Let OCS_x and OCS_y are two outer contour strings. These OCSs are linked together if the Euclidean distance between them is less than a threshold (*euclid_thre*) obtained from the training step. This threshold corresponds to the maximum Euclidean distance obtained from two logo OCSs of the training set.

After linking some OCSs together, we re-compute their bounding boxes and update their probability scores (i.e. a new features extraction and region scoring).

D. Setting and training

At all stages, the different processes used in our system are driven by an initial setting and training. The Table I gives the list of concerned processes with their associated parameters. The parameters for pre-processing (*se*) and features extraction (*m*) are set based on our experiments. The rest of parameters (regression function and localization correcting) are trained from a representative set of document selected the test dataset.

V. EXPERIMENTS

We use the Tobacco-800 dataset [1] and the performance evaluation metrics described in [3] to evaluate our results. The Tobacco-800 dataset, and the metrics in [3], have been employed to evaluate the systems in [3], [9] and [10] , allowing like this a comparison of our results with those of the literature.

Tobacco-800 is a database of real-life documents composed of 1290 images. It is given with ground-truth describing the logos at vector graphics level (tight rectangular bounding boxes). Metrics proposed in [3] considers lower and upper bounds of bounding boxes when comparing detected and ground truthed logos, to ensure that a logo is correctly detected. Final characterization is given in terms of precision accuracy.

For our experiments, we have set our system regarding the parameters described in Table I. We have employed a 3×3 structured element for our dilatation operator. The parameter m for features extraction has been set at 8, resulting in a features vector of size $k = 40(36 + 4)$. The regression function and the localization correction parameters have been trained from two subsets of the Tobacco-800 database. These database subsets respect respectively the distributions used in [3], [10] and then [9]. Table II gives results we obtained and compare them with [3], [9] and [10].

This first subset is composed of 50 images (40 logo images and 10 non-logo images). The rest of the database (1240 images) has been used for testing. This distribution respects experiments done in [3], [10]. Table II gives our results. We obtain 91% on accuracy at the precision of 44% and 75% on accuracy at the precision of 84%. Fig.3 presents in details our results of accuracy and precision as a function of probability threshold (*prob_thre*). The mean running time on Intel Core i5 2.4 GHz 2.5 G RAM is 430 ms and the off-line training phase costs only 30 seconds.

The second subset respects distribution described in [9], where the experiments was done only on the logo images. This subset is composed of a test set of 376 logo documents using the same training set as described in the first experiment. As detailled in Table II, we obtain 90.05% on accuracy and 92.98% on precision.

Fig.4 visually presents a result image of the original image in Fig. 2(a). Enven if the input image is mixed with many hand writings, most of text lines and hand writings is identified with low probability scores and there is only one logo region correctly assigned with a high score. In Fig.5, we present some cases in which our approach does not work well because of poor results in the segmentation process.

VI. CONCLUSIONS

In this paper, we propose a new approach to address the problem of logo detection in document images. At first, preprocessing is done including skew correction, morphological filtering, and automatic binarization. A contour tracking algorithm combined to line segmentation method result in outer contour strings describing graphics and text parts of

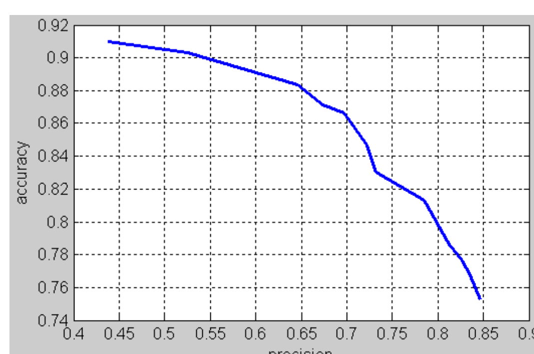

Figure 3. Our results of accuracy/precision on the test set of 1240 images

Table II
RESULTS OF CURRENT APPROACHES ON TOBACCO-800 DATABASE

ID	Approaches	Test set	Training set	Total	Accuracy	Precision	Time (ms)
1	G. Zhu and D. Doerman [3]	1240	50	1290	84.2%	73.5%	680
2	Zhe Li el al. [10]	1240	50	1290	86.5%	99.4%	328
3	Our approach	1240	50	1290	75 - 91%	44 - 85%	430
4	H. Wang and Y. Chen [9]	316	100	416	80.4%	93.3%	Absent
5	Our approach	376	50	426	90.05%	92.98%	430

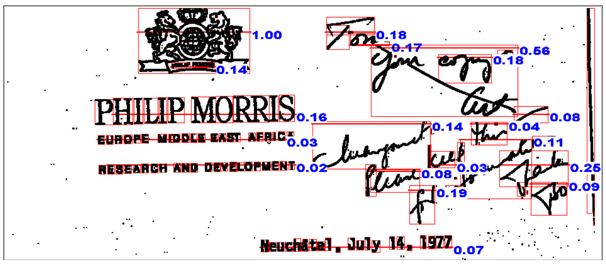

Figure 4. Final results: each OCS is assigned a logo presence probability

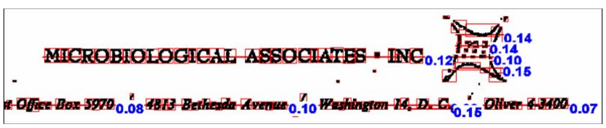

Figure 5. The segmentation errors resulting in our logo detection

the documents. A bi-levels descriptor is computed from each OCS composed of coarse features (that catch graphical and domain information) and finer ones (exploiting a gradient based representation to characterize the contour regions). The logo detection problem is considered as the regression fitting by using Gentle Boost. A final process of correcting the logo's location is performed to reconstruct the full shape of logo. Experiments done on the Tobacco-800 dataset comprehensively represent our results in terms of both detection rate and precision rate compared to the best systems in literature. Based on this work, we are going to employ recognition methods to reject false detection and improve the precision level.

ACKNOWLEDGMENT

This work has been supported by the Viet Nam International Education Development project, numbered 322-VIED.

REFERENCES

[1] HD. Lewis et al., *Building a test collection for complex document information processing*. In Proc. Annual Int. ACM SIGIR Conference, 2006, pp. 665-666.

[2] Y. Freund and R. E. Schapire, *Experiments with a New Boosting Algorithm*. In Proc. the thirteenth International Conference (Morgan Kauman, San Francisco, 1996), 148-156.

[3] G. Zhu and D. Doerman, *Automatic Document Logo Detection*. In Proc. ICDAR07, pp.864-868.

[4] Marcal Rusiol and Josep Llads, *Logo Spotting by a Bag-of-words Approach for Document Categorization*. In Proc. ICDAR09, pp.111-115, 2009.

[5] T. Pham, *Unconstrained logo detection in document images*. Pattern Recognition, 36(12):3023-3025, 2003.

[6] Black W. et al., *A general purpose follower for line structured data*. Pattern Recognition 198, vol 14, pp.33-42.

[7] N. Otsu, *A threshold selection method from gray-level histograms*, IEEE Trans. 9(1) (1979)62-66.

[8] O' Gorman L., *The document spectrum for page analysis layout*. IEEETrans. PatternAnal. 15(11), 1162- 1173 (1993).

[9] H. Wang and Y.Chen, *Logo detection in document images based on boundary extention of features rectangles*. In Proc. ICDAR09, pp. 1335-1339, 2009.

[10] Zhe Li, Schulte-Austum M. and Neschen. M, *Fast Logo Detection and Recognition in Document Images*, International Conference on Pattern Recognition (ICPR), 2010, 2716-2719.

[11] A. Amin and S. Fischer, *A Document Skew Detection Method Using the Hough Transform*. In Pattern Analysis & Applications Volume 3, Number 3, pp.243-253, 2000.

[12] Loncaric, S. *A Survey of Shape Analysis Techniques*. In Pattern Recognition (PR), 1998, 31, pp.983-1001.

[13] Seiden, S.; Dillencourt, M.; Irani, S.; Borrey, R. & T.Murphy, *Logo detection in document images*. Conference on Imaging Science, Systems and Technology (CISST), 1997, pp.446-449.

[14] Doermann, D.; Rivlin, E. & Weiss. I, *Logo Recognition Using Geometric Invariants*. International Conference on Document Analysis and Recognition (ICDAR), 1993 , pp.894 - 897.