2009 10th International Conference on Document Analysis and Recognition

Graph b-Coloring for Automatic Recognition of Documents

Djamel GACEB Véronique EGLIN Frank LEBOURGEOIS Hubert EMPTOZ
LIRIS UMR 5205CNRS, INSA de Lyon 69621 Villeurbanne Cedex
{djamel.gaceb1,veronique.eglin,flebourg,hubert.emptoz}@insa-lyon.fr

Abstract

In order to reduce the rejection rate of our automatic reading system, we propose to pre-classify the business documents by introducing an Automatic Recognition of Documents stage (ARD) as a pre-processing step. This important step will guide the other stages involved in the recognition process of the documents contents. Once the document class identified, the reading system will use correct information from the ARD stage to improve the segmentation of the layout, the recognition of the document structure, the parameterization of the OCR, and the final decision for the rejection. We propose in this paper an original method for the classification of business documents suited for complex layouts having great variability. We introduce the graph coloring approach for both layout analysis and document classification. The proposed method is reliable, robust to various constraints and guarantees a real-time answer to the sorting of business documents.

1. Introduction

The automatic processing of documents is a considerable added value to the companies. It makes the documentary heritage more accessible and allows new services which can improve the organization of companies. In particular, the automatic sorting of documents save time and reduce the costs of manual handling. In order to break the actual limits of the OCR, the solution consists to improve the overall organization of the computer vision system by introducing feedback loops and other processes which bring new information about documents contents at each stage of the processing. Any recognition system of documents requires the introduction of prior knowledge related to the type of document to be recognized [1]. Most of these recognition systems embed this knowledge into the program directly which becomes difficult to adapt for new documents.

The Automatic Recognition of Documents (ARD) system is used for documents classification, which brings information to various stages like the OCR, the layout analysis stage, the decisions stage and the selection of the dictionaries (figure 1).

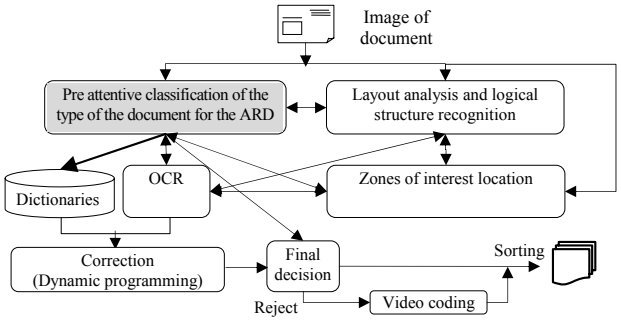

Figure 1. Location of the ARD stage into the general scheme of the documents sorting system

The introduction of an ARD stage in the overall scheme of a documents sorting system remains an unsolved problem, which must respect several constraints: 1) a large variety of documents; 2) a real-time processing; 3) A high resolution of the image (300 dpi); 4) the superposition of different information layers (marks, logo). To satisfy all these constraints, we propose a flexible ARD architecture based on a new approach, which uses the hierarchical coloring of graphs. Until now, this powerful approach has never been used in document image analysis.

The paper is organized as follows: The next section describes the various existing methods of classification of documents and their limits. In the third section, we present the graph b-coloring method and its application in an ARD stage. At the end of the paper, we will present our results obtained with a large database.

2. Classification method

2.1. Different strategies

A document can be considered as a complex organization of various objects (text, graphic, notes

978-0-7695-3725-2/09 $25.00 © 2009 IEEE
DOI 10.1109/ICDAR.2009.72

and other symbols of all kinds) located randomly and having an unpredictable arrangement. The recognition of document consists to cluster documents having similar structure and text contents into the same class. Numerous unsupervised and supervised classifications can be used to classify documents images like the K-Means, Markov chains, decision trees, graphs isomorphism, SVM, Neuronal Networks, and various statistical approaches. These methods can use different features: 1) image features only and/or, 2) layout features and/or, 3) structure features and/or, 4) text content features.

ARD systems based on documents structures [2] or text contents [3] are difficult to use for a real-time application. Moreover, the layout of the documents is important to the recognition of the documents having different forms [4]. We need simple and representative features which allow to discriminate quickly a great variety of documents into separate classes. In order to answer to the constraints imposed by our application, the approach proposed in this paper is based on the documents layouts.

2.2. Document layout based approaches

Most of methods for documents classifications based on the layout use a hierarchical representation of blocks (word, text lines, graphics, checked box, tables…). This representation simplifies the comparison between each element of the layout. Heroux [4] describes a document with a tree, where each node describes an element of the layout. A comparison of trees allows the classification of documents. Esposito [5] uses a simple language to describe the elements of the layouts and their relation. Cesarini [6] compares X-Y trees to classify documents. [7][8] proposes to modify the X-Y tree into XYM tree. Baldi compare with a K-NN rule the distance edition between XYM trees and Diligenti uses the tree to build an Hidden Tree Markov Model. [9] proposes a document classification based on Graph theory and uses a First Order Gaussian Graph (FOGGs) where both nodes and branches are described by probabilities learned on a training set.

2.3. The need of new approach

All methods described previously use complex data structures for both the classification and the description of the layout. They require the extraction of knowledge from a large training set, which must contain representative documents with all possible layouts. Because of the great variability of the layouts, systems described previously are difficult to control. To answer to the industrial needs, we propose an efficient tool, which guarantees stable and coherent results and respects real time constraints. We propose a new architecture based on graph coloring.

3. Formal aspect of the graph coloring

The graph coloring provides the minimal number of classes necessary to decompose n objects (connected components, text lines, and blocks of text) into homogeneous subsets. Each object i is represented by a node v_i and each pair of objects supposed different are connected with an arc $E(v_i,v_j)$. The finite graph $G=(V,E)$ is defined by the finite set $V=\{v_1,...,v_n\}$ $(|V|=n)$ of nodes, and by a finite set $E=\{e_1,..., e_m\}$ $(|E|=m)$ of arcs.

3.1 Coloring

The coloring of the graph $G(V,E)$ consists to affect one color to all nodes so that two adjacent nodes (dissimilar) do not have the same color. These colors correspond to the classes. The number of colors used to colorize the graph G is called the chromatic number $\chi(G)\leq n$. This number represent the smallest integer k of partitions of V into k homogeneous subsets [10]. The graph G from figure 2, represents a set of 11 different patterns V described by their nodes $\{x_1,...,x_{11}\}$. We use four different colors to colorize the 11 nodes so that two adjacent nodes have different colors.

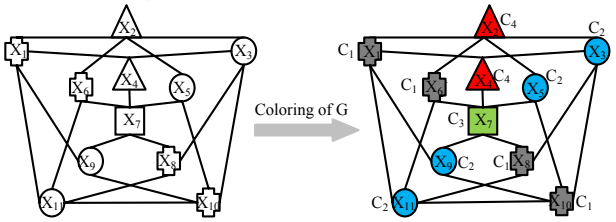

Figure 2. Coloring of graph G with 4 colors.

3.2. The b-coloring principle

The coloring is called b-coloring, if for each color C_i, it exists at least one node v_i colored C_i having nodes in its neighborhood of different colors. The node v_i is called dominating node for the color C_i.

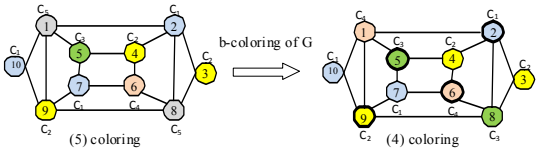

Figure 3. Example of b-coloring.

Figure 3 gives an example of b-coloring of G. The nodes 2, 5, 6 and 9 are the dominating nodes of four colors. The b-chromatic number of a graph G, noted $b(G)$, is the maximal number of colors k_b such that G is b-colorized by the k_b colors.

3.3. Implementation

Numerous techniques of coloring exists in the literature and most of them are described and compared by Paschos [10]. In our study, we are particularly interested to distributed algorithms of graphs coloring and b-coloring proposed by Effantin in [11]. All these algorithms have been introduced by d'Elghazel [12] who proposes a new unsupervised classification method of medical data based on the b-coloring of graph where the number of classes is not known in advance. In comparison with the other classification techniques of the literature, the b-coloring provides a correct representation of classes and guarantees a better disparity between classes.

4. Graph coloring in ARD system

We present in this section, the different steps of our ARD system.

4.1. Layout analysis

4.1.1. Binarization and detection of connected components

The binarization is the first step of the process in our ARD system and has a great influence on the overall performance of the sorting system. The separation between the binarization stage and the Connected Components (CCs) segmentation stage increases the computation cost and provides an over-segmentation of the noise in empty areas and around the textured background of the paper. We optimize this critical step by combining the binarization and the segmentation of CCs in the same stage. To save computation, we apply a local thresholding only around text areas which are detected by the "cumulated gradients algorithm" using multi-resolution and mathematical morphology [13].

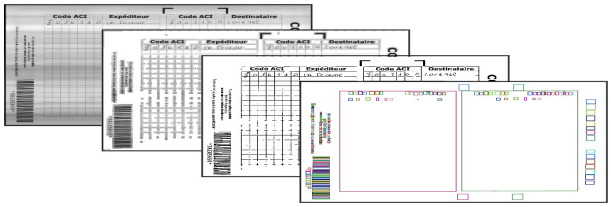

Figure 4. Local thresholding around text areas and CCs extraction

Then we detect the Connected Components (CCs) which guide the extraction of the components of the foreground. A fast CCs extraction is based on the Line Adjacency Graph (LAG) of Pavlidis [14].

4.1.2. Layout segmentation by hierarchical coloring of connected components

The key idea is to segment the layout by using a pyramidal strategy based on the graph coloring method (algorithm 1). It allows to separate relevant elements and to group them into homogeneous classes and simultaneously reject irrelevant elements. By coloring of CCs, we separate textual regions from non textual zones, then we group the CCs of textual zones into text lines. The method is detailed in [13].

Algorithm 1. Graph_Coloring (G_k)

Begin
If $col_k(i) \neq \varnothing$ *Then*
Let $M = N^k_{col}(i) \cup \{col_k(i)\}$; $q = 0$;
For every node $j \in N^k_{adj}(i)$ Such that $col_k(j) = \varnothing$ Do
q =min $\{r | r > q, r \notin M$ and $r \notin col_k(j)\}$;
If $q \leq \Delta + 1$ Then $col_k(j) := q$;
Else $col_k(j)$:=min$\{r | r \notin N^k_{col}(j)\}$;
EndIf; EndDo; EndIf; End.

With $col^k(i)$ the color of the node $i \in V_k$, $N^k_{adj}(i)$ the set of adjacent nodes of the node i. $N_{col}(i)$ describes the set of colors of nodes $N^k_{adj}(i)$, $d^k_{eg}(i) = |N^k_{adj}(i)|$ represents its degree, and : $\Delta_k = \max \{d^k_{eg}(i) | i \in V_k\}$.

4.2. Representation of documents

We describe the documents with a reduced number of features computed from their layouts : 15 global features, which describe the entire document, and 20 local features, measured on text lines. We use two type of representations. *1) Structural local representation :* each document j is described by a ranked sequence of n text lines ($Rs(j) = (L^j_1, L^j_2, ..., L^j_n)$) where the line L_t is represented by a feature vector of p dimensions $L_t = (x^t_1, x^t_2, ..., x^t_p)$. *2) Global representation :* each document j is represented by a vector of m global features $Rv(j) = (y^j_1, y^j_2, ..., y^j_m)$.

4.3. Distances measures

To compare two documents, we combine two distances (D_{Rv} over Rv^m and D_{Rs} over Rs^n) given by the following equation :

$$DT = \gamma D_{Rv} + (1-\gamma) D_{Rs} \text{ with } \gamma = \{ k = \arg\max_{0 \leq k \leq 1} (\psi_k) \}$$

The value γ must be determined to maximize the quality of the classification Ψ [13]. If two documents

are separated by a small distance *DT* then they are similar. The distance D_{Rv} between two documents, represented by the features *Rv(i)* and *Rv(j)* is given by the equation:

$$D_{Rv}[Rv(i),Rv(j)]=\left[\sum_{k=1}^{m}\left|y_k^i-y_k^j\right|^{\alpha}\right]^{\frac{1}{\alpha}} \text{ with } \alpha=2$$

The edit distance D_{RS} realizes a spatial mapping between *Rs(i)* of lines n_i and *Rs(j)* of lines *nj* by using a dynamic programming or a warping transform. The non-linear matching between *Rs(i)* and *Rs(j)* is described by the runs : $C=c_1,\ldots,c_K$ with $c_k=(i_k,j_k)$ (figure 5).

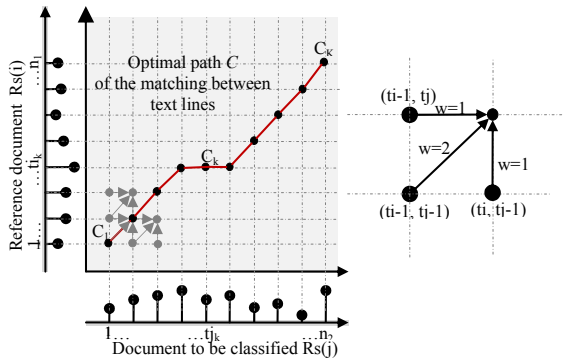

Figure 5. Dynamic matching between text lines

The weighted sum of errors along of the optimal path *C* of the matching is given by :

$$D(c)=\frac{\sum_{k=1}^{K} d(c_k)\cdot w_k}{\sum_{k=1}^{K} w_k} \text{ with } d(c_k)=d(L_{t_i}^i,L_{t_j}^j)=\sqrt{\sum_{l=1}^{p}\left[x_{t_i}^i(i)-x_{t_j}^j(j)\right]^2}$$

The weighting coefficients are:

$$w_k=ti_k-ti_{k-1}+tj_k-tj_{k-1} \text{ and } \sum_{k=1}^{K} w_k=n_i+n_j$$

In this case, the problem to solve becomes:

$$D_{RS}[Rs(i),RS(j)]=\frac{1}{n_i+n_j}\min_{C}\sum_{k=1}^{K} d(c_k)\cdot w_k$$

The number of possible paths grows exponentially with the number of text lines within the documents we have to compare. This problem can be solved efficiently by the Dynamic Programming Algorithm (DPA) which find an optimal matching between text lines. To save computation time, we do not compare all possible matching but only text lines which are spatially comparable (figure 6). We compute a limited number of costs along the diagonal in the table of the DPA (figure 5).

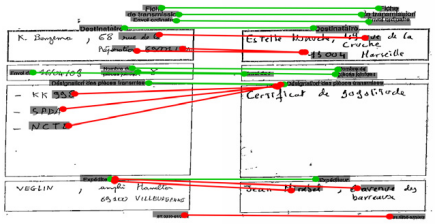

Figure 6. Dynamic comparison of documents

4.4. Classification of documents

We also use the graph coloring to classify documents. We represent a set *R* of *N* documents in a graph *G(V,E)* where each node is a document. Two different nodes v_i and v_j are adjacents if and only if the distance *DT* between the documents *i* and *j* is strictly superior to a threshold S_{DT}. The determination of this threshold is detailed in [13]. The adjacency between the nodes is given by :

$$E[v_i,v_j]=\begin{cases}1 & \text{if } DT(v_i,v_j)>S_{DT}\\ 0 & \text{otherwise}\end{cases}$$

To decompose the set *R* into homogeneous subsets, we colorize the graph *G* then we apply the algorithm of b-coloring described in [13][11].

4.5. Training step

During this step, we provide a training set *R* of *N=512* documents already classified into *14* classes. The training uses the classification algorithm detailed in the section (4.4.). The b-coloring provides automatically a set of N^* dominants nodes representing the classes which are used for a real time recognition of a unknown document.

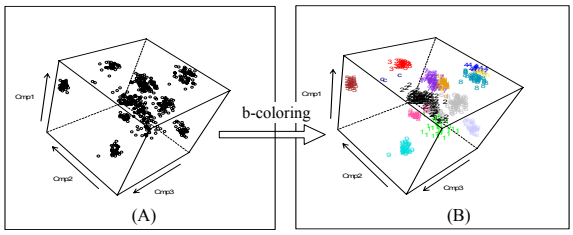

Figure 7. (A) 512 documents projected in the feature space, (B) 14 clusters found by the b-coloring.

4.6. Recognition of the document class

To classify an unknown document *T(i),* the recognition stage compares its description with all representatives documents of each class (dominating nodes) from R^* computed during the training step. The matching algorithm recognizes in real time the type of document *T(i)* from the class of the nearest documents from R^*:

$$Type[T(i)]=\begin{cases}\text{Reject if } \arg\min_{k=1\ldots N^*}\left(DT\left[T(i),R_k^*\right]\right)>S_{DT}\\ Type(R_k^* \mid \arg\min_{k=1\ldots N^*}\left(DT\left[T(i),R_k^*\right]\right) \text{ otherwise}\end{cases}$$

The adjacency threshold S_{DT} also determines the rejection rate of the recognition for documents which have not been learned by the system.

5. Experimentation

We use the KAPPA coefficient to evaluate precisely the classification of *512* documents from the training database with *3* different classification approaches (KMeans, SVM and b-coloring). A high level of the KAPPA coefficient near *100%* indicates that the classification is correct. Figure 8 shows that the b-coloring provides a better classification compared to the other methods.

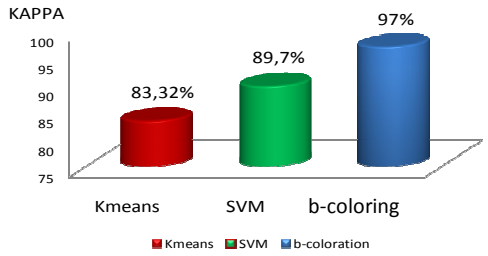

Figure 8. Comparison of results from different classification methods

We have tested three classifiers with a test set of *576* documents classified into *14* classes which have been learned and *2* new unknown classes (of reject) which have not been used during the training step. Figure 9 gives the recognition rate for the *14* known classes and the *2* unknown classes. The b-coloring gives better results in term of recognition and rejection rate.

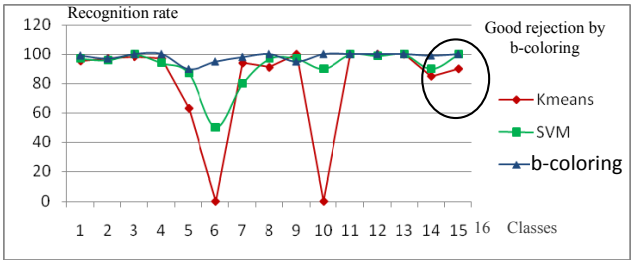

Figure 9. Comparison of results

6. Conclusion

We have presented a new method for the classification of business documents based on the hierarchical coloring of graphs. The documents are represented by their layouts. The hierarchical coloring of graph has been introduced during the layout analysis step to improve the robustness of the segmentation. The b-coloring has been also used during the training step to find the representatives documents for each class. Because of the small constraints required by the b-coloring, this new method can answer to a large variety of classification problems. It can process documents having variable layouts and provides a real representation of documents classes by using dominants documents. Moreover the b-coloring allows to increase the coherence between different phases of the ARD system and reduces the overall computation cost of the system. In future works, we propose to extend this method for the incremental training of the rejected documents. This new step will allow to reclassify documents which have been rejected by the system.

7. References

[1] R. MULLOT, *Book*: *Les documents écrits de la numérisation à l'indexation par le contenu*, Hermes science Publication, 2006, pp. 365.

[2] V. EGLIN and S. BRES, "Document page similarity based on layout visual saliency: application to query by example and document classification", *the 7th ICDAR*, Scotland, 2003, pp. 1208-1212.

[3] H.K. MOHAMED, "Automatic documents classification", *IEEE ICCES'07*, pp. 33-37.

[4] P. HÉROUX and al, "Classification method study for automatic form class identification", *the 14th ICPR*, Brisbane, Australia, 1998, pp. 926-929.

[5] F. ESPOSITO and al, "Machine learning for intelligent processing of printed documents". *J. Intell. Inf. Syst*, 2000, pp. 175-198.

[6] F. CESARINI and al, "Encoding of modified X–Y trees for document classification", *6th ICDAR'01*, pp. 1131-1136.

[7] S. BALDI and al, "Using tree-grammars for training set expansion in page classification", 7th ICDAR'03, pp. 829-833.

[8] M. DILIGENTI and al, "Hidden Tree Markov Models for document image classification", *IEEE Trans. Pattern Anal. Mach. Intell*. 25(4), 2003, pp. 519 -523.

[9] A.D. BAGDANOV and M. Worring, "First order Gaussian graphs for efficient structure classification", *Pattern Recognit*. 36(6), 2003, pp.1311-1324.

[10] V. PASCHOS, *Book*, *Optimisation combinatoire5: problèmes paradigmatiques et nouvelles problématiques*, Lavoisier, France, 2007, pp. 270.

[11] B. EFFANTIN and H. KHEDDOUCI, "A distributed algorithm for a b-coloring of a graph", *IEEE ISPA'2006*, Serrento, Italy, 2006.

[12] H. ELGHAZEL and al, "A New Clustering Approach for Symbolic Data: Algorithms and Application to Healthcare Data", *BDA 2006*, Lille, France.

[13] DJ. Gaceb et al, "Address block localization based on graph theory". *DRR XIV, SPIE, USA*. 2008, pp.12.

[14] Z. Pavlidis and J. Zhou, "A Page Segmentation and Classification", *CVGIP92*, vol.54, no. 6, pp. 484-496.