

Zone Classification Using Texture Features

Dmitry Chetverikov*

Jisheng Liang†

József Kömüves*

Robert M. Haralick†

Abstract

We consider the problem of zone classification in document image processing. Document blocks are labelled as text or non-text using texture features derived from a feature based interaction map (FBIM), a recently introduced general tool for texture analysis [3, 4]. The zone classification procedure proposed is tested on the comprehensive document image database UW-I created at the University of Washington in Seattle. Different classification procedures are considered. The performance ranges from 96 % to 98 % using 6 FBIM texture features only.

1. Introduction

Document image understanding involves determining the geometric page layout, labeling blocks as text or non-text, determining the read order for text blocks, recognizing the text of text blocks through an OCR system, determining the logical page layout, and formatting the data and information of the document in a suitable way for use by a word processing system or by an information retrieval system [5]. The zone classification or labeling is an important step in the document image understanding process.

A geometric page layout of a document image page is a specification of the geometry of the maximal homogeneous regions and the spatial relations of these regions. A region is homogeneous if all its area is of one type: text, or figure etc. and each text line of the page lies entirely within some text region of the layout. Many of the page segmentation algorithms for determining ge-

ometric layout have the zone labeling modules embedded in their systems. Wahl et. al. [14] extract features of the blocks including the area of the connected component of the block, the number of black pixels in the block on the original document image, the mean horizontal black run lengths of the original image within the blocks, and the height and width of the bounding rectangle of the block. Text areas are classified into text, horizontal solid black lines, graphic and halftone images, and vertical solid black lines. Fisher et. al. [8] extract connected component features of the run length smoothed image, such as component height, width, aspect ratio, density, perimeter, and area for classifying each block as text or non-text. Saitoh and Pavlidis [11] classify each component into text, text or noise, diagram or table, halftone image, horizontal separator, or vertical separator, using block attributes such as block height, height to width ratio, and connectivity features of the line adjacency graph, and whether there are vertical or horizontal rulings. Pavlidis and Zhou [10] label each block as text or non-text using features such as ratio of the mean length of black intervals to the mean length of white intervals, the number of black intervals over a certain length, and the total number of intervals. Amamoto et. al. [13] decide a block is a text block if the length of the longest black run length in the vertical and horizontal directions is smaller than a given threshold. Each block is then assigned a class label from the set: text, figure, image, table, and separation line. Belaid and Akindele [1] label the blocks' contents as small letter text, medium letter text, large letter text, graphics or photographs, based on the connected component analysis and rules which are determined before hand during a learning stage. Sivaramakrishnan et. al. [12] extract features for each zone such as run length mean and variance, spatial mean and variance, fraction of the total number of black pixels in the zone, and the zone width ratio, and use a decision tree clas-

*Computer and Automation Research Institute, Budapest, Kende u.13-17, H-1111 Hungary. E-mail: mitya@leader.ipan.sztaki.hu.

†University of Washington, Seattle, USA.

sifier to assign a zone class on the basis of its feature vector. Jain and Zhong [9] utilize a neural network to train a set of masks for discriminating three main texture classes: halftone, background, and text and line drawing regions. The text and line-drawing regions are further discriminated based on connectivity analysis.

However, except for Sivaramakrishnan et.al. [12] who tested their module on a significantly large data set (UW-I document image database), there was no systematic evaluation of the performance of the zone classification modules. In the present study, we also use the UW-I database to evaluate the performance of the FBIM texture features in document zone classification. The FBIM has been recently introduced in [3, 4] as a new general tool for texture analysis. As a pilot proof of the efficiency of this tool, we show that the FBIM texture features can be applied to document image analysis. Our zone classification approach is based on textural characteristics only. This imposes certain lower limit on the size of the zone but at the same time speeds up the procedure as the original resolution of the document image can be drastically reduced.

In section 2 we briefly describe the main aspects of the FBIM approach, then introduce those FBIM texture features that are proposed for text/non-text separation. The classification procedures, the experimental protocol and the results of the tests are presented in section 3.

2. FBIM features for text separation

A texture feature-based interaction map [3, 4] displays the structure of statistical pairwise pixel interactions evaluated through the spatial dependence of a gray-level difference histogram (GLDH) feature. The FBIM approach uses the extended GLDH (EGLDH) introduced earlier in [2]. This extension was necessary to provide, at arbitrary spacings, the angular resolution required for accurate anisotropy analysis. The EGLDH overcomes the problem of interdependence of the angle and the magnitude of the spacing vector arising in a digital image when the conventional GLDH is used. Here, we only give a brief partial description of the FBIM approach sufficient for understanding its use in zone classification. Other functions and major algorithms of the method are described elsewhere [3, 4].

The polar interaction map—the basic entity of the method—is an intensity coded polar representation of an EGLDH feature, with the columns enumerating the magnitude, the rows the angle of the varying spacing vector. It is obtained as follows:

For a discrete set of spacing vectors $\vec{d}_{ij} = (\alpha_i, d_j)$, $\alpha_i \in [0, 2\pi]$, $i = 0, \dots, M_1$, $d_j \in [1, d_{max}]$,

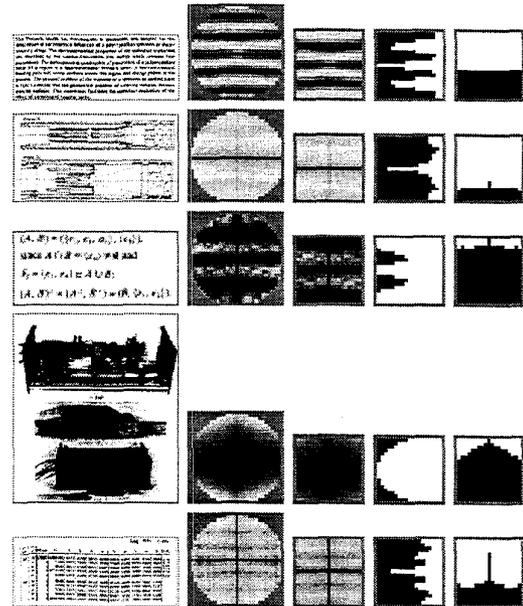


Figure 1. Computing the FBIM features for different types of zones. Each row shows a zone image, the interaction map, the central part of the map, its row projections and the column projections of the negated map. Row 1: text. Row 2: drawing. Row 3: math. Row 4: halftone. Row 5: table.

$j = 0, \dots, M_2$, compute the extended histogram $H(k; \alpha_i, d_j)$ and an EGLDH feature $F(\alpha_i, d_j)$. Define *polar feature based interaction map* as $M_{pl}(i, j) = F(\alpha_i, d_j)$.

$M_{pl}(i, j)$ is then transformed to the Cartesian coordinates to obtain the *Cartesian (XY) interaction map* $M_{xy}(m, n)$ which is used in our zone classification procedure.

In this study, we use as the EGLDH feature the *median* of the absolute gray-level differences. Alternative EGLDH features can also be applied. They are similar to the standard GLDH features proposed in [6].

Examples of interaction maps for various types of zones are shown in figure 1. In these examples, the size of the polar interaction map matrix is 72×15 , which corresponds to the angular resolution of 5 degrees and the maximum spacing of 15 pixels. The original resolution of the zone images has been reduced by a factor of 8.

In the text separation procedure, we assume that the document image has been partitioned into homogenous zones, i.e. blocks of certain type: text, table, math,

etc. Our task is to classify the zones into text or non-text using the FBIM texture features designed for this purpose. This is possible because text has a characteristic textural structure which exhibits itself in the interaction map. The FBIM features proposed are selective to the typical layout of a text map as a periodic arrangement of horizontal lines corresponding to the lines of the text. If a zone includes at least two lines of text, there are two or more distinct maxima in the row projections of the map and no distinct maxima in column projections of the negated map. This is because texts are normally dense, with a moderate number of background-to-background transitions in the vertical direction. The letters occupy about half of the zone, i.e. the difference in area between the dark and the light runs of the projections is relatively low.

The following FBIM features were used for zone classification:

$$RMAX = \begin{cases} 1 & \text{if } MAPROW \text{ has at least 2 maxima} \\ 0 & \text{otherwise} \end{cases}$$

RHT = average height of maxima in $MAPROW$

RWD = average width of maxima in $MAPROW$

$$DLD = \left| 1 - \frac{2 \cdot \text{darkarea}}{\text{totalarea}} \right|$$

CHT = average height of maxima in $MAPCOL$

CWD = average width of maxima in $MAPCOL$

Here $MAPROW$ and $MAPCOL$ are the row and the column projection arrays, respectively, $\text{darkarea}/\text{totalarea}$ the fraction of the total area of $MAPROW$ occupied by the dark runs. (See figure 1.) Only distinct maxima are considered. If there is no distinct maxima in $MAPROW$ ($MAPCOL$), by default $RHT = RWD = 0$ ($CHT = CWD = 0$).

Texture features can only be defined for those zones that are large enough to exhibit textural properties. Many of the zones in the UW-I database documents are small blocks like page numbers or very short, specific texts, e.g. headlines. These zones cannot be treated as two-dimensional textures and no characteristic structure of the interaction map can be expected for them. Such zones were rejected. The zone size rejection criterion was designed so as to accept only those zones that include at least two lines of a large font. A similar limit was set for the number of columns. In our tests, the minimum zone size was 160 by 80. This reduced the number of zones from the original 13831 to 4713. Other methods should be applied to classify the small zones.

On the other hand, the FBIM texture features do not need the original high resolution of the document

images. Textural appearance of, say, a table differs from that of a text at much lower resolution. For this reason, the resolution of the UW-I images was reduced by a factor of 8 leading to a significant gain in the processing speed.

3. Classification procedures and tests

We have computed the proposed texture features for all those zones of the UW-I database that are large enough to be treated as 2D textures. Several classification algorithms were tested with this set of feature values. A binary decision tree classifier assigns an unknown unit to one of the classes through a hierarchical decision procedure. It has the capability to break down a complex decision-making process into a collection of simpler decisions at various levels of the tree. The classification process can be described by means of a tree, in which at least one terminal node is associated with each pattern class, and the interior nodes represent various collections of mixed classes. In particular, the root node represents the entire collection of classes into which a unit may be classified [7]. Each nonterminal node is associated with a decision function, and generates two child nodes. An input pattern is classified through traversing a path from the root node till a terminal node. Only the decision functions associated with nonterminal nodes along the path are tested.

There are several techniques available for training a binary decision tree classifier. Given a set of training instances, each described by n features and labeled a class name, the general top-down growing strategy works as follows. At each nonterminal node, starting from root node, the best decision function is learned by using a criterion of optimality and a training subset the node receives. The learned decision function splits the training subset into two subsets generating two child nodes. The process is repeated at each newly generated child until a stopping condition is satisfied and the node is declared as a terminal node.

The maximum entropy reduction is used as the optimality criterion to find a decision function at each nonterminal node. Shannon's entropy is defined as,

$$E = - \sum_i p_i \log p_i,$$

where p_i is the probability of class i . At each nonterminal node t , there is a candidate decision function S that divides node t into left child t_L and right child t_R such that a proportion p_L of the cases in t go into t_L and a proportion p_R go into t_R . One could then define the goodness of the decision function S to be the

decrease in entropy:

$$\Delta E(S, t) = E(t) - E(t_L)p_L - E(t_R)p_R.$$

Then choose a decision function that minimizes $\Delta E(S, t)$ over all decision functions S . Let θ be a pre-determined threshold. If $\Delta E(S, t) < \theta$, partitioning is halted and the node t is made a terminal node.

The simplest form for a linear decision rule is a comparison of one measurement component to a threshold. This is called a thresholding decision rule. The feature space is actually partitioned using hyperplanes each perpendicular to a feature axis. Due to its simplicity, an exhaustive search is performed to find the best feature-threshold pair for a nonterminal node at the training stage. The Fisher's linear decision rule method computes the direction of the linear decision function or hyperplane at a nonterminal node through maximizing the ratio of the projected between-class scatter to the projected within-class scatter.

The hold-out method is used for the error estimation. We divide the data set into N parts, train on the first $N - 1$ parts, and then test on the N th part. Then train on the $N - 1$ parts, omitting the $N - 1$ st part, and test on the $N - 1$ st part. Continue the training and testing, each time omitting one part from the decision tree construction procedure and then testing on the omitted part. Then combine the results of the N tests together to establish an estimate of the error rate [7]. In this experiment, the value of N is chosen as 3.

The training and testing data set is drawn from the scientific document pages in the University of Washington document image database. The total number of 4713 zones are evaluated. The contingency tables for the results of the classification using thresholding decision rule and Fisher's linear decision rule are shown in Tables 1 and 2. The tables present the numbers of zones of a particular class that are identified as members of a different class. The performance of this algorithm is about 96 %.

A further preliminary experiment has been carried out to clarify whether it is possible to improve the accuracy by rejecting those zones whose class is judged uncertain. In the texture feature space, a distance from the center from the 'text' cluster was obtained for the training samples and then computed for the test samples as well. A decision certainty level was specified and the samples falling into the range of distances labelled as uncertain rejected. This improves the classification accuracy at the expense of a small increase in the total number of the rejected zones. Typically, additional 4 % of the evaluated zones are discarded as uncertain while the error rate falls below 2 %. However, more tests are

	Text	Non-text	Total	% Error
Text	3746	99	3845	3 %
Non-text	96	772	868	11 %

Table 1. Contingency table showing the classification results of text and non-text zones using the thresholding decision rule.

	Text	Non-text	Total	% Error
Text	3776	69	3845	2 %
Non-text	139	729	868	16 %

Table 2. Contingency table showing the classification results of text and non-text zones using Fisher's linear decision rule.

needed to finalize this result.

4. Conclusion

We have shown that FBIM texture features can be used to discriminate between text and non-text zones in document images. The proposed approach is based on textural information only. This imposes a lower limit on the size of the zone but has an advantage of much lower resolution being sufficient for operation. The performance of our approach has been systematically evaluated with a large reference document image database. The experimental results are statistically significant and can be compared to the performance of the alternative approaches once the results of their systematic evaluation become available. At the moment, the only alternative systematic study seems to be that of Sivaramakrishnan et.al. [12] who use a feature vector of much higher dimension (67) and report an accuracy of 97 % in zone classification into 9 different classes. We are now comparing the results of the two studies and exploring the ways of improving the performance of our zone classification module. This includes a detailed error analysis followed by a possible redefinition of the features.

5. Acknowledgment

This work was supported in part by the Hungarian grant OTKA T14520 and the EU COPERNICUS grant CT94 0153.

References

- [1] A. Belaid and O. Akindele. A labeling approach for mixed document blocks. In *Proc. 2nd ICDAR*, pages 749–752, 1993.
- [2] D. Chetverikov. GLDH based analysis of texture anisotropy and symmetry: an experimental study. In *Proc. International Conf. on Pattern Recognition*, pages 444–448. Vol.I, 1994.
- [3] D. Chetverikov. Pattern orientation and texture symmetry. In *Computer Analysis of Images and Patterns*, pages 222–229. Springer Lecture Notes in Computer Science vol.970, 1995.
- [4] D. Chetverikov and R. Haralick. Texture anisotropy, symmetry, regularity: Recovering structure from interaction maps. In *Proc. British Machine Vision Conference*, pages 57–66, 1995.
- [5] R. Haralick. Document image understanding: Geometric and logical layout. In *Conf. on Computer Vision and Pattern Recognition*, pages 385–390, 1994.
- [6] R. Haralick and K. Shanmugam. Textural features for image classification. *IEEE Trans. Systems, Man, and Cybernetics*, 3:610–621, 1973.
- [7] R. M. Haralick and L. G. Shapiro. *Computer and Robot Vision*, volume I. Addison-Wesley, 1992.
- [8] J. F. S. Hinds and D. D'Amato. A rule-based system for document image-segmentation. In *Proc. International Conf. on Pattern Recognition*, pages 567–572, 1990.
- [9] A. Jain and Y. Zhong. Page segmentation using texture analysis. *Pattern Recognition*, 1995.
- [10] T. Pavlidis and J. Zhou. Page segmentation by white streams. In *Proc. 1st ICDAR*, pages 945–953, 1991.
- [11] T. Saitoh and T. Pavlidis. Page segmentation without rectangle assumption. In *Proc. International Conf. on Pattern Recognition*, pages 277–280, 1992.
- [12] R. S. I. P. J. H. S. Subramaniam and R. Haralick. Zone classification in a document using the method of feature vector generation. In *Proc. 3rd ICDAR*, volume 2, pages 541–544, 1991.
- [13] N. A. S. Torigoe and Y. Hirogaki. Block segmentation and text area extraction of vertically/horizontally written document. In *Proc. 2nd ICDAR*, pages 739–742, 1993.
- [14] F. W. K. Wong and R. Casey. Block segmentation and text extraction in mixed text/image documents. *Comp. Graph. and Image Proc.*, 20:375–390, 1982.