THE PUBLISHING HOUSE
OF THE ROMANIAN ACADEMY

PROCEEDINGS OF THE ROMANIAN ACADEMY, Series A,
Volume 4, Number 2/2003, pp.000-000

A PATTERN RECOGNITION APPROACH TO IMAGE SEGMENTATION

Tudor BARBU

Institute of Computer Science, Iaşi, Romania
E-mail: tudcont@iit.iit.tuiasi.ro

The objective of the present paper is to describe a pattern recognition approach for image segmentation. First, in the introduction, we describe the general aspects of uniformity and texture recognition. Then we provide a mean-based feature extraction approach for uniformity analysis and a moment-based one for texture analysis. In the classification stage we propose both supervised and also unsupervised clustering algorithms. The graphical results of our implementations are also presented.

Key words: pattern recognition, classification, image segmentation, uniformity analysis, texture analysis

1. INTRODUCTION

Image pattern recognition represents an important computer vision domain, consisting of classification of the patterns of a given image, based on various similarity criterions. In this paper we consider the image regions as patterns.

The image segmentation task consists of dividing the input image in a number of different objects called *image segments* or *clusters*, such that all the pixels from a segment have a common property called *similarity criterion*. There are various segmentation methods, such as the boundary-based approaches ([1],[2],[7],[8]), which use edge-detection for segments' extraction, or amplitude thresholding approaches ([1],[2]). We have used a region-based approach ([1],[2],[3]) with an uniformity or texture based similarity criterion.

A color image, or a grayscale one, contains a finite set of uniform and texture regions. If there are no textures in the image, it is called *nontextured*. An uniform region is composed from pixels with closed values. For example, a connected set of pixels with the same RGB value represents a uniform blue, yelow (or other color) region. A textured region represents a special case of uniformity, based on the repetition of basic texture units called *texels*.

First our aproach decides on the input image's type: textured or nontextured. This can be done interactively, the user deciding the type of the image displayed on computer screen. Depending on the image's character and state several enhancement filtering actions ([1],[2],[3],[4]) are applied in the preprocessing state: image smoothing, edge enhancement or contrast adjustment. For the resulted enhanced image, as the one in Figure 1 (b), we have to solve a texture (or uniformity) recognition problem: for each texture or uniform pattern of the image to find all its occurences, which means the clustering of all image regions in a proper number of categories, depending on their uniformity or texture. In the filtered image we can distinguish two textured regions and nine uniform regions which should be grouped in a number of classes.

This work has contributions both in the feature extraction stage and also in the classification stage. Our proposed moment-based approach states that a given input texture can be uniquely described by its up to order three moments. In the classification step we have developed a variant of VQ *k - means* supervised algorithm and a variant of *region-growing* unsupervised method. We have performed a MATLAB implementation for the algorithms of the proposed approaches and have obtained some graphical results. In

Recommended by Horia-Nicolai TEODORESCU,
Member of Romanian Academy

the next sections we present the two main steps of the pattern recognition method ([1],[3],[10],[12]): feature detection and pattern classification.

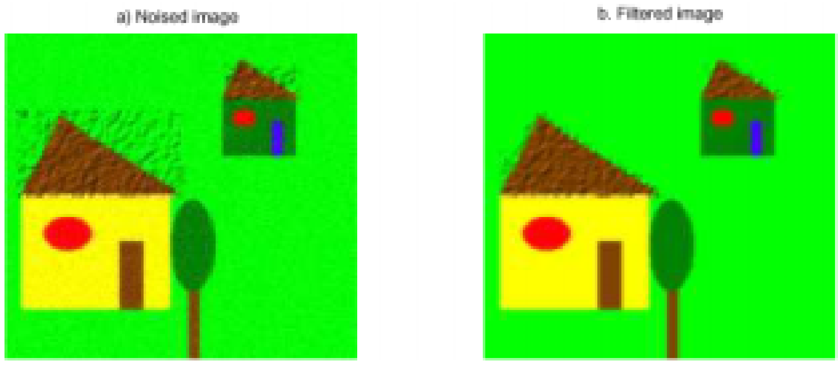

Fig.1

2. SOME FEATURES EXTRACTION APPROACHES

For the uniform regions case we have considered a single characteristic value in the features vector. A region R of a given image $I\,(R \subset I)$ is uniform if the difference between the greatest pixel value and the smallest pixel value is below a small enough chosen threshold, which is equivalent with:

$$\forall p \in R, \ \left| p - \text{mean}(R) \right| < T, \tag{1}$$

where p is a pixel and T the chosen threshold. Therefore we use a mean-based uniformity analysis approach. For each pattern (pixel) we have computed the mean of its $n \times n$ neighborhood and consider the obtained value as a *feature vector*. The block dimension n must have an odd value such that the current pixel can be its center. We choose a greater or a smaller value for n depending on the image noise's amount and the uniform regions' sizes. If a certain amount of noise is still present in the image, one should choose an n value greater than 1 $(n \geq 3)$. Otherwise, in the smooth image case, n could be 1 (the feature set for a given pixel is the pixel's value). The value of n could be set interactively by the user, but it should be chosen such that all the regions of interest be preserved (a large n could delete the small uniform regions). The patterns with their feature values are then passed to the classification stage.

Feature detection is more difficult for textured regions. Texture analysis has a long history, many approaches being developed for texture identification, classification and segmentation. There are many texture retrieval methods based on Gabor filtering([5],[9]) or multiresolution filtering techniques such as the wavelet transform ([11]). There are also histogram-based and boundary-based approaches ([1],[2],[3]) for texture analysis. We have used a moment-based approach ([1]) for texture features extraction. In the discrete case f represents an $M \times N$ image region and its $(p+q)$th order moment will be:

$$m_{p,q} = \sum_{x=1}^{M} \sum_{y=1}^{N} x^p y^q f(x,y), \ p,q = 0,1,... \tag{2}$$

We have an important result given by the Moment Representation Theorem: Image function f is determined uniquely by its set of moments $m_{p,q}$, therefore a texture region of input image is determined by its moments up to a given order.

For each pixel an $n \times n$ neighborhood is considered, as in the uniform regions case, $n \geq 1$ being an odd number and the current pixel being the center of the neighborhood. For this $n \times n$ region a set of moments is computed and considered as a feature vector. We have computed for each pixel its neighborhood's moments up to order 3 :

$$V = [m_{00}, m_{01}, m_{02}, m_{10}, m_{11}, m_{12}, m_{20}, m_{21}, m_{22}] \quad (3)$$

We have used V as the features vector for texture clustering but we have also tested the mean value of V which gives similar results and is more advantageous because of an easier implementation in the classification stage and a shorter processing time.

3. CLASSIFICATION METHODS AND SEGMENTATION RESULTS

Having now a feature vector for each pixel of the image (textured or not), pattern classification can be done in a supervised manner or in a nonsupervised one. A supervised learning can be applied to a classification problem if the number of classes is known and also the set of classes' prototypes or a training set are available. If no information about classes is available or only a little is known about them (for example we know their number only and do not have any training set), the problem requires an unsupervised learning.

First we present our distribution-free supervised classification approach based on a VQ (*Vector Quantization*) method. We have proposed a VQ classifier and have developed a variant of the k - means algorithm for it. Let $C_1, C_2, ..., C_K$ be the classes, K being the only information we know about them, and $x_1, ..., x_N$ be the objects we want to cluster. Their feature vectors, $V(x_i)$, $i = \overline{1, N}$ are sorted in ascending order. Let $v_1 = \min_i V(x_i)$, $v_K = \max_i V(x_i)$, $i = \overline{1, N}$, and $d(v_1, v_2)$ the distance between them. Dividing this distance to $K - 1$ we obtain K equally spaced vectors $v_1, v_2, ..., v_K$, $\frac{d(v_1, v_K)}{k-1}$ being the distance between 2 succesive vectors. For each i, $\min_j d(v_i, V(x_j))$ is computed and $V(x_j)$, for which this minimum Euclidean distance is obtained, is included in the training set. The feature vector $V(x_j)$ is then eliminated from the available feature space and the reasoning is repeated until training set will contain K vectors.

This method for obtaining the training feature vectors works not only in the image segmentation case but also in a general pattern classification problem. For the image clustering problem the $x_i, i = \overline{1, N}$ objects are the image pixels, N being the image area, and $V(x_i)$ are their feature vectors. The number of classes K is set interactively by displaying the image on the screen. We can obtain the training vectors via the described approach but also we have tested an user interactively approach. The user can select K pixels right--clicking K times on appropriate different uniform or texture regions of the image I (Matlab facility), their feature vectors working as a training set. With this set our k - means algorithm computes successfully the classes' prototypes, as can be seen below:

For $i = 1$ to K do
1. $C_i := \{t_i\}, i = \overline{1, K}$ being the training vectors
End
Repeat
 For $i = 1$ to K do
 2. Find j for which $\min_j d(V(x_i), \text{mean}(C_j))$ is obtained
  If $V(x_i) \notin C_j$ then {position change}
  3. $C_j := C_j \cup \{V(x_i)\}$;
  4. Find $C_l, l \neq j$, such that $V(x_i) \subset C_l$;
  5. $C_l := C_l \setminus \{V(x_i)\}$;
  End
 End

Until no position change {*optimization criterion*}

The prototype feature vectors are the centroids of the C_i sets, $i = \overline{1, N}$. The proposed optimization criterion is the condition that all the feature vectors are in the right place and no more position changes are needed. The algorithm realizes a supervised classification, the sets C_i being the obtained classes. For our image segmentation problem, $C_1, C_2, ..., C_K$ are classes of pixels. Displaying the result of this classification, the classes C_i appear as disconnected regions, each of them represented by a different color, as one can see below. We have interactively chosen $K = 7$ and a neighborhood dimension $n = 3$.

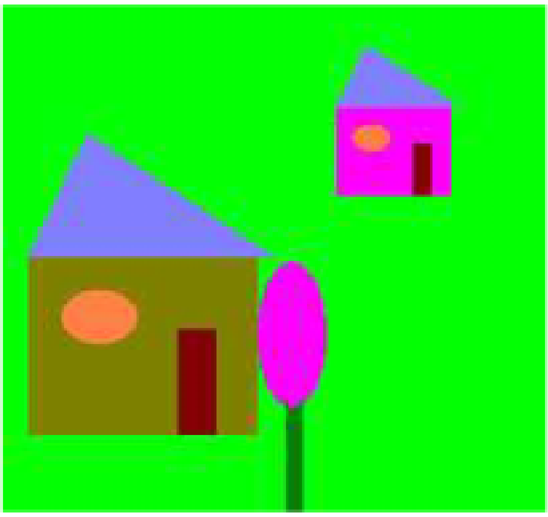

Fig. 2

As shown by Figure 2, each class contains one or more image objects having the same uniformity or texture and being displayed with the same color. The uniformity recognition and texture recognition problems are therefore solved. For solving the segmentation problem too, the component objects of each class have to be extracted. This can be done considering the pixels of each class, at a time, as being the image foreground (setting them to 1 - white) and the others as being the background (setting them on 0 - black). The connected regions of each binary image built this way are then determined and each of them considered as a segment (cluster) in the initial image. We present the segmentation results in Figure 3: 11 image clusters have been obtained, each of them marked by a different color.

Unsupervised learning (classification) consists of a natural grouping in the feature space. In unsupervised clustering the classes prototypes are not known and no training sets are used.

Fig.3

There are many unsupervised learning approaches for pattern recognition such as SOM (Self--Organizing feature Maps) methods developed by Kohonen ([10]). We propose a region - growing algorithm for classification and provide two forms of it. In the first case the number of classes is introduced interactively by the user, as in the supervised case. Let consider K the classes' number and V the feature space. The next pseudocode presents the algorithm:

For $i = 1$ to K
1. $C_i = \{V(x_i)\}$;
2. $C = \{\ \}$; index $= 1$; $\{C$ - helping set$\}$,
End
Repeat
2. $m :=$ number of C_i's (initially N);
  For $i = 1$ to $m - 1$
    For $j = 1$ to m
      If $d(\text{mean}(C_i), \text{mean}(C_j)) = \min_{i_1, j_1}(\text{mean}(C_{i_1}), \text{mean}(C_{j_1}))$ {smallest overall dist.} then
        3. $C(\text{index}) := C_i \cup C_j$; {region merging}
        4. inc(index);
      End
    End
  End
  For $i = 1$ to index $- 1$
  5. $C_i := C(i)$;
  End
Until $m = K$ {optimization criterion}

For our segmentation problem the regions are the pixels at the beginning. At each loop the most closed regions (which have closest mean values) are merged in a new region. The process continues until the optimization criterion is satisfied: regions' number equals the classes' number.

We could obtain a more natural grouping if we eliminate the knowing of classes' number and therefore the user interactivity. If the number of clusters is unknown our region growing algorithm has to be modified to work properly. At step 3 we introduce one more condition for the unification of two classes: C_i and C_j are merged if $d(\text{mean}(C_i), \text{mean}(C_j)) < T$, where T is a threshold depending on values from C_i and C_j. The optimization criterion is modified too: *Repeat ... until there are no more regions' unifications*. We have tested as threshold a fraction of the minimum between the compared means, therefore $T = \frac{1}{\alpha}\min(\text{mean}(C_i), \text{mean}(C_j))$. If the regions to be discriminated are very closed, α has a large value.

The results obtained with the unsupervised technique are similar with those presented in Figure 2, but the computing time is larger because region- growing algorithm has a higher complexity than the k - means algorithm. The classes $C_i, i = \overline{1,7}$ represent not only classes of pixels but also classes of objects with the same uniformity or texture. For image segmentation the reasoning continues as in supervised case and we get similar results with those in Figure 3.

4. CONCLUSIONS

We have described our pattern recognition method for the segmentation of both textured and also nontextured images and presented its results. Our main contributions are the choosing of a proper moments' set which gives a good texture description and the development and implementation of the two described clustering algorithms.

The results of this paper could be succesfully applied in another image recognition field: the image objects recognition. We will present in our next work an objects recognition approach based on composing the detected uniform and texture regions.

REFERENCES

1. ANIL K. JAIN, *Fundamentals of Digital Image Processing*, Prentice--Hall International, Inc., Englewood Cliffs New Jersey, 1989.
2. WILLIAM K. PRATT, *Digital Image Processing*, John Wiley and Sons, Inc, 1991.
3. THEO PAVLIDIS, *Algorithms for Graphics and Image Processing*, Computer Science Press, Inc, 1982
4. JAN T. YOUNG, JAN. J. GERBRONDS, LUCAS J. VAN VLIET, *Fundamentals of Image Processing*, Tu-Delpht Interactive Course, 1998.
5. B.S. MOUNJUNATH, W.Y. MA, *Texture Features for Browsing and Retrieval of Image Data*, IEEE Transactions on Pattern Analysis and machine Intelligence, vol. 18, No. 8, August 1996.
6. DENHSHENG ZHANG, GUOJUN LU, *Enhanced generic Fourier descriptors for object-based image retrieval*, International Conference on Acoustics, Speech and Signal Processing, 2002
7. TOLL, WILLIAM E., *Point Based Approaches in Graphics*, Ph. D. dissertation, Computer Science Department, University of Kentucky, Lexington, KY, 1998.
8. B.S. MOUNJUNATH, W.Y. MA, *Edge flow: a framework of boundary detection and image segmentation*, Proc. IEEE International Conference on Computer Vision and Pattern Recognition, San Juan, Puerto Rico, June 1997, pp. 744-49.
9. JAN T. YOUNG, L. J. VAN VLIET, M. VAN GINKEL, *Recursive Gabor filtering*, IEEE Transactions on Signal Processing, Vol. 50, No. 11, 2798-2805, 2002.
10. T. KOHONEN, *Self-Organizing Maps*, Springer Series in Information Sciences, Vol. 30, Springer, Berlin, Heidelberg, New York, 1995, 1997, 2001. Third Extended Edition, 501 pages. ISBN 3-540-67921-9, ISSN 0720-678X
11. ANDREW LIAN, JIAN FAN, *An Adaptive Approach for Texture Segmentation by Multichannel Wavelet Frames*, Center for Computer Vision and Visualization, August 1993.
12. RICHARD O. DUDA, *Pattern Recognition for HCI*, Department of Electrical Engineering San Jose State University, 1996-1997

Received March 11 ,2003