IEEE Int. Workshop VLSI Design & Video Tech.
Suzhou, China, May 28~30, 2005

# Meaningful Regions Segmentation in CBIR*

*Shikui Wei, Yao Zhao, Zhenfeng Zhu*
(Institute of Information Science, Beijing Jiaotong University, Beijing, 100044, China)
*E-mail:mysnowdays@126.com*

## Abstract

In this paper, a new approach to fully automatic image segmentation is proposed to get the meaningful regions of general-purpose image. In order to avoid image over segmenting, the original input image is first smoothed by Gaussian filters with different scales. Then an improved ISODATA clustering algorithm with parameters selecting dynamically is proposed to cluster the image pixels into different regions. To eliminate those fragmentary regions, a region merging strategy is also presented. The final experimental results show that the proposed approach can effectively separate the objects from background of general-purpose image.

## 1. Introduction

The increasing amount of images has motivated the research on content-based image retrieval (CBIR) during the last decade. Because general-purpose image segmentation is thought of as a helpless goal by the computer vision community [2], researchers of CBIR generally take much time on the study of global image properties, e.g. color and texture, which avoid the problem of image segmentation. However, from the segmentation results, even imperfect, it's possible to identify meaningful regions which are very beneficial to the retrieval of images [15].

In this paper, an automatic image segmentation technique, which allows us to get approximate object-regions and background-regions, is presented. Fig.1 shows a schematic diagram of the proposed regions segmentation algorithm. As a classical clustering algorithm, parameters of ISODATA must be given in advance, which is the main fault for automatic segmentation of image. In order to implement unsupervised initial segmentation, the standard ISODATA algorithm [6, 7, 14] is improved. As an important step, a postprocessing operation is performed, using to merge small regions into adjacent big regions. Related approaches such as stochastic model based approaches [2, 3], region growing [4] and clustering analysis approaches [1, 6, 7, 9, 11], have been applied to segmentation with low-level features.

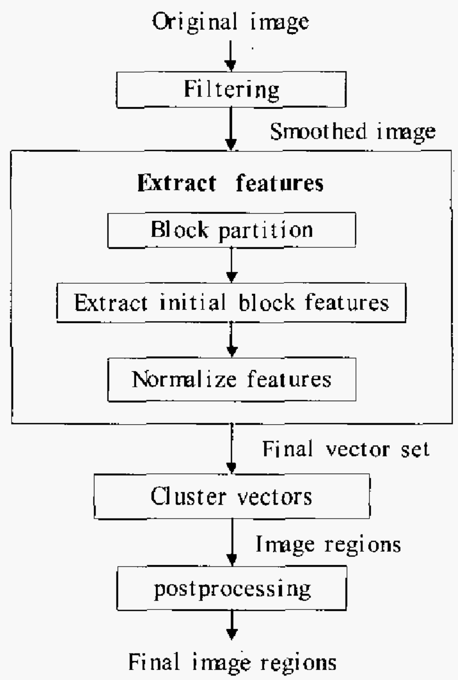


Fig.1 Flowchart of regions segmentation

The rest of the paper is organized as follows: In Section 2, preprocessing of original image and feature extraction are described. In Section 3, we review the ISODATA algorithm and describe emphatically the method used to obtain automatically parameters of the algorithm. In Section 4, a method to postprocess the segmentation image is given. The experiments and results are discussed in Section 5. Finally, we conclude with a brief discussion about our algorithm and future work.

* This work is partially supported by National Natural Science Foundation of China (No. 60373028) and the Program for New Century Excellent Talents in University.

## 2. Preprocessing and Feature Extraction

### 2.1 Original Image Preprocessing

We know that many noise points scattered about the whole image, which affect badly the accuracy of clustering analysis. Moreover, oversegmentation of image is not an ignorable problem; for example, zebra stripe maybe become its own region. In order to reduce noise and avoid oversegmenting image, Gaussian filter is used to smooth original image. Gaussian filter has a number of good properties, amounting to "well-behavedness" [5]. Furthermore, it has been proved that Gaussian is indeed an optimal filter for estimating ideal data from incomplete discrete data [8, 12]. Multi-scale Gaussian filter has been applied in many works [2, 8, 9, 10].

In this work, two-scale Gaussian filter is used to smooth original image. In the location (x, y) of image, the convolution can be computed by:

$$V_s(x,y) = g_s(x,y) \otimes f(x,y) \tag{1}$$

where $g_s(x,y)$ is the Gaussian kernel , $s$ denotes the scale of Gaussian filter, and f(x, y) is the value of pixel in the location (x, y).

In order to select the scale dynamically, image can be filtered using the equation as follow:

$$V(x,y) = \max_{s=s_1,s_2}\left(V_s(x,y)\right) \tag{2}$$

where $V(x,y)$ represents the final value of pixel in the location (x, y). The comparison of segmentation results is show in Fig.2.

(a) (b) (c)

Fig.2. Comparison of segmentation results (a) Original image (b) Segmentation result without smooth filtering (c) Segmentation result with smooth filtering

### 2.2 Feature Extraction

For the extraction of initial feature, a similar method as in [1] is adopted, which uses six features of color and texture. Firstly, the preprocessed image is partitioned into blocks with 4×4 pixels. Then the average color components in LUV color space are extracted. In order to get the other three texture features, the Daubechies-4 Wavelet transform [13] is applied to luminance component of the image. A 4×4 block is decomposed into four frequency bands, and each band contains four coefficients. From the coefficients of high frequency bands, i.e., HL, LH and HH bands, three texture feature values are obtained. The block feature such as in HL band can be computed as:

$$f = \left( \frac{1}{4} \sum_{i=0}^{1} \sum_{j=0}^{1} c_{k+i,l+j}^{2} \right)^{\frac{1}{2}} \tag{3}$$

where $c_{k+i,l+1}$ denotes the coefficient of HL band.

Thus the final initial feature vector of a block consists of 3-D color features and 3-D texture features. Two initial blocks feature vectors acquired randomly, in which the first 3-D features denote color features and the rest describe texture features, are shown as follows

$$f_1 = (127.61, 12.23, 15.63, 0.85, 1.21, 0.32)$$

$$f_2 = (127.25, 3.74, 10.71, 0.27, 0.91, 0.12)$$

By analyzing the above two vectors, we find that the first three color feature values are generally much bigger than the texture values. That is to say, when the Euclidean distance is adopted to compute similarity between two feature points in the feature space, the significance of texture feature is inappreciable. In order to solve this problem, the normalization for feature should be considered. The equations of normalization is shown as follows

$$f_{d\max} = \max_{n}\left(f_{nd}\right), \quad \begin{cases} d = 1, 2, \cdots D \\ n = 1, 2, \cdots N \end{cases} \tag{4}$$

$$f_{nd} = f_{nd} / \mathrm{f}_{\mathrm{dmax}} \tag{5}$$

where $D$ represents the dimension of feature vector, $N$ denotes the total number of feature vectors, and $f_{nd}$ is the $j^{th}$ component of the $n^{th}$ feature vector.

## 3. Blocks clustering

To group the above-mentioned blocks into different regions, an improved ISODATA algorithm is proposed. The ISODATA algorithm [3, 6, 14] is a classical cluster technique, which is also an iterative procedure. Unlike the k-means algorithm, ISODATA can split and merge regions according to a certain criteria automatically. We will not elaborate here on the classical algorithm. Rather, we only give some important equations, which is vital to our method. The center of cluster is given by:

$$m_j = \frac{1}{N_j} \sum_{y \in \Gamma_j} y, \ j = 1, 2, \cdots C \tag{6}$$

where $N_j$ represent the number of vectors in the $j^{th}$ cluster and $\Gamma_j$ denotes the $j^{th}$ cluster.

The standard deviation, $\sigma_j = \left[\sigma_{j1}, \sigma_{j2}, \cdots \sigma_{jD}\right]^T$, can be obtained as follows

$$\sigma_{jd} = \sqrt{\frac{1}{N_j} \sum_{y_k \in \Gamma_j} \left(y_{kd} - m_{jd}\right)^2} \tag{7}$$

where $y_{kd}$ denotes the $d^{th}$ component of the $k^{th}$ feature vector, and $m_{jd}$ denotes the $d^{th}$ component of the $j^{th}$ cluster center vector.

$$\sigma_{j\max} = \max_d \left[\sigma_{jd}\right] \tag{8}$$

$$\delta_{jl} = \left\| m_j - m_l \right\|, \ \ l = j+1, j+2, \cdots C \tag{9}$$

For standard ISODATA algorithm, a number of parameters should be given in advance, which is the main fault for automatic segmentation of image. According to the experiments, we find that the merging parameter, $\theta_c$, and splitting threshold, $\theta_s$, are vital to image segmentation. So, it is very important to get both of parameters dynamically. Here, we propose a method which can update $\theta_c$ and $\theta_s$ dynamically. For each iteration, we compute the $\theta_c$ and $\theta_s$ after we have got the values of $\sigma_{j\max}$ and $\delta_{jl}$. The equations computing $\theta_c$ and $\theta_s$ are given by:

$$\theta_s = \frac{1}{C} \sum_{j=1}^{C} \sigma_{j\max} \tag{10}$$

$$\theta_c = \frac{1}{C!} \sum_{j,l} \delta_{jl}; \quad C! = C(C-1)\cdots 1 \tag{11}$$

Since the other parameters are relatively stable, thus they can be given once for all.

## 4. Postprocessing

Our aim is to separate image into several meaningful and natty big regions, which is very important to image retrieval. However, after using above segmentation scheme, a lot of fragmentary regions are also created among big regions. In order to get rid of those fragments and keep consistent with people's perception, a further postprocessing procedure is performed

1. Get the location of a fragmentary region.
2. Calculate the number of adjacent pixels between the fragmentary region and all the big regions.
3. Merge the fragment into the big region which has the most adjacent pixels with the fragment.

The above three steps can be iterated until all the fragmentary regions are merged.

## 5. Experiments

It's difficult to give a fair comparison with existing segmentation algorithm such as K-mean algorithm which depends on additional information to be provided by the user during the segmentation of different images. Because the EM-based algorithm is the only other automatic segmentation algorithm we can access to, we compare this algorithm with our approach using the same images. In our experiments, we selected eight categories of images, including leopards, planes, eagles, dinosaurs, portraits, flowers, horses and tigers. Both of segmentation algorithms are tested on the images selected randomly form the eight categories. Due to the limitation of space, we compare only six images. Fig. 3 shows the segmentation results of two algorithms. In general, our algorithm performs as well as the EM-based algorithm for the images of simple background, such as eagles and planes. The eagle and plane are separated from the sky accurately. However, our algorithm can separate objects from complicated background better than EM-based algorithm, such as horses, flowers and leopards. Moreover, our approach can get more natty objects and background regions than EM-based algorithm, which is very important to the content-based image retrieval. From the results, we also can see that segmentation using our algorithm generally satisfy our requirement although several kinds of "errors" may be seen in some images.

## 6. Conclusions

In this paper, an approach for fully automatic general-purpose image segmentation is proposed. Two-scale Gaussian filter is used to avoid over segmenting. Then, an improved ISODATA algorithm with parameters selecting dynamically is presented to group the image pixels into different regions. In order to eliminate fragments, a region merging strategy is also presented. Future research work is focused on how to apply the proposed region segmentation method to region-based image retrieval.

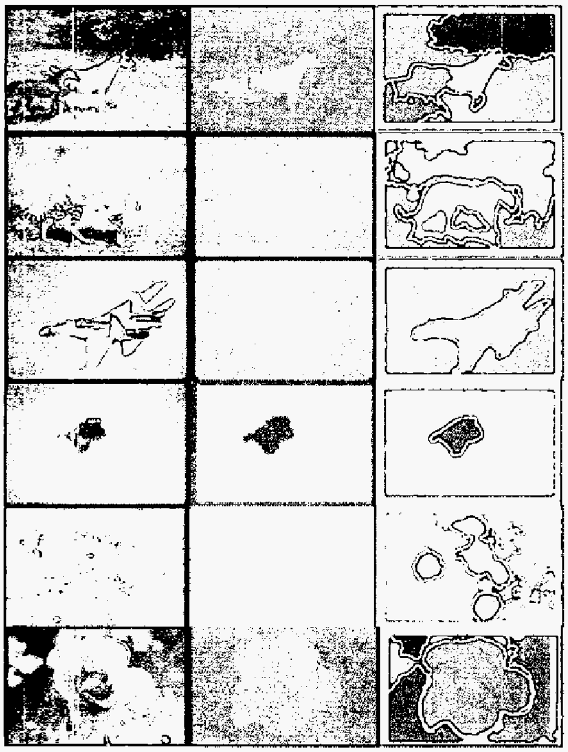

Fig.3. Comparing of segmentation results. The first column is original images, the middle column is the result of our algorithm, and the last column is the result of BlobWorld system.(copied from [ 16]).

## Acknowledgements

This work is funded by the NSFC under grant No. 60373028. .

## Reference

[1] J.Z. Wang, J. Li, and G. Wiederhold, "SMPLIcity: semantics-sensitive integrated matching for picture libraries," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 23, no. 9, pp. 947-963, Sep. 2001.

[2] C. Carson, S. Belongie, H. Greenspan, and J. Malik, "Blobworld: Image Segmentation Using Expectation-Maximization and Its Application to Image Querying," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 24, no. 8, pp. 1026-1038, Aug. 2002.

[3] A.J.M. Traina, A.G.R. Balan, L.M. Bortolotti, C.T. Jr., "Content-based Image Retrieval Using Approximate Shape of Objects," Proc. IEEE Symposium on Computer-Based Medical Systems (CBMS'04), pp. 91-96, June 2004.

[4] D. Yining, B.S. Manjunath, and H. Shin, "Color image segmentation," Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 446-451,1999.

[5] A.P. Witkin, "SCALE-SPACE FILTERING: A New Approach to Multi-Scale Description," Proc. IEEE Int'l Conf. Acoustics, Speech and Signal Processing, vol. 3, pp. 39A. 1. 1-39A. 1. 4, 1984.

[6] G. Hua, S. Guangda, and D. Cheng, "Fuzzy and ISODATA Classification of Face Contours," Proc. IEEE Int'l Conf. Machine Learning and Cybernetics, pp. 3568-3573, Aug. 2004.

[7] G.H. Ball and D.J. Hall, "A Clustering Technique for Summarizing Multivariate Data," Behavioral Science, 12(2), pp. 153-155, 1967.

[8] H. Jeong, and C.I. Kim, "Adaptive Determination of Filter Scales for Edge Detection," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 14, no. 5, pp. 579-585, May 1992.

[9] B. McCane, and T. Caelli, "Multi-scale Adaptive Segmentation using Edge and Region Based Attributes," Proc. IEEE Int'l Conf. Konwledge-Based Intelligent Electronic Systems, pp. 72-80, May 1997.

[10] S. Makrogiannis, G. Ecomomou, and N.G. Bourbakis , "Segmentation of Color Images Using Multiscale Clustering and Graph Theoretic Region Synthesis ," IEEE Trans. SYSTEMS, MAN, AND CYBERNETICS—PART A: SYSTEMS AND HUMANS , vol. 35, no. 2, pp. 224-238, Mar. 2005.

[11] H. Chih-Cheng, and K. Youngsup, "The Application of Agglomerative Clustering in Image Classification Systems," Proc. IEEE Conf. SOUTHEASTCON, pp. 23-26, 1992

[12] T.Poggio,H. Voorhees, and A. Yuille, "A regularized solution to edge detection ," *AI Memo*, no. 833, Mass. Inst. Technol., AI Lab, May 1985 .

[13] I. Daubechies, *Ten Lectures on Wavelets*. Philadelphia: SIAM, 1992.

[14] B. Zhaoqi, and Z. Xuegong, *Pattern Recognition*, Tsinhua University Press, Beijing, 1999.

[15] Z. Yujin, G. Yongying, and L. Yun, "Object-Based Techniques for Image Retrieval", In: Multimedia Systems and Content-based Image Retrieval, Idea Group, Inc, Chapter7 (156-181), 2004

[16]URL: http://elib.cs. Berkeley.edu/segmentation