2002

Color Image Segmentation in the Color and Spatial Domains

Tie Qi Chen[1], Yi L. Murphey[1], Robert Karlsen[2], and Grant Gerhart[2]
[1]VisiTek, Inc. 1409 HarbrookeAnn Arbor, Michigan 48103, Yilu@umich.edu

[2]U. S. Army - TACOM, Warren, MI 48397-5000

Please address all correspondence to:
Professor Yi Lu
Department of Electrical and Computer Engineering
The University of Michigan-Dearborn
Dearborn Michigan 48128-1491, U.S.A.
Voice: 313-593-5028, Fax: 313-593-9967
yilu@umich.edu

ABSTRACT

In this paper we describe a color image segmentation system that performs color clustering in a color space followed by color region segmentation in the image domain. In color space, we describe two different algorithms that cluster similar colors using different criteria and present our evaluation results on these two algorithms in comparison with three well-known color segmentation algorithms. The region segmentation algorithm merges clusters in the image domain based on color similarity and spatial adjacency. We developed three different methods for merging regions in the image domain. The color image segmentation system has been implemented and tested on a variety of color images, including satellite images and moving car images. The system has shown to be both effective and efficient.

Keywords: Color image, image segmentation, fuzzy logic

Color Image Segmentation in the Color and Spatial Domains

1. Introduction

In the past decade, color imaging has become popular in many applications including object classification and recognition, video surveillance, image indexing and retrieval in image databases and video databases, feature based video compression, etc. [1, 2]. In some applications, image contents can be better described in terms of color features combined with spatial relations such as enclosure and adjacency due to the irregularity of object shapes in images. This paper describes our research in color image segmentation, which is often a necessary computational process for color-based image retrieval and object recognition [3].

Image segmentation is a process of partitioning image pixels based on selected image features. The pixels that belong to the same region must be spatially connected and have similar image features. If the selected segmentation feature is color, an image segmentation process would separate pixels that have distinct color features into different regions and, simultaneously, group pixels that are spatially connected and have similar color into the same region. In color imagery, image pixels can be represented in a number of different color spaces, such as RGB, XYZ, or LUV [4, 5, 6]. One major concern in color image segmentation is that the computational complexity increases significantly in comparison with gray scale image segmentation. The process of image segmentation can be considered as unsupervised clustering, if a priori knowledge about the number and type of regions present in the image is not available [7, 8]. Image clustering procedures often use the individual image pixels as units and compare each pixel value with every other neighboring pixel value, which requires excessively long computation times for images of high resolutions. A high quality representation of color requires 8 bits, using PCM (Pulse Code Modulation) quantization for each of the three color-components, Red (R), Green (G) and Blue (B). For each image pixel, 24 bits of amplitude quantization is required, resulting in 2^{24} = 16,777,216 distinguishable colors. This leads to high computational cost in image clustering.

In this paper we describe an efficient color image segmentation system, depicted in Figure 1. The color image segmentation system consists of two stages of computation. At the first stage, we use a color clustering algorithm to generate clusters of similar colors in the color histogram space of an image. The histogram of a color image in a selected color space is a three dimensional(3D) discrete feature space that provides the color distribution of the image. The output of the color clustering algorithm is a set of non-overlapping color clusters, CL_1. Each cluster in CL_1 contains similar colors and all colors in the same cluster are assigned with the same color label. The labeling of the clusters in CL_1 results in a multi-threshold image in which pixels of the same region have the same color label and are spatially connected. These regions form the second cluster set CL_2 to be used in the second stage, where a region segmentation algorithm agglomerates the initial clusters in CL_2, based on the spatial connection and the color distances between the adjacent regions. The second stage merges the selected adjacent regions and does not split any regions. Therefore, the first design criterion for the fuzzy clustering algorithm is that different color regions should be in different clusters in CL_1. However, we should prevent the generation of too large a set of clusters. For example, one extreme is that each cluster in CL_1 contains only one color, which certainly satisfies the first criterion but misses the purpose of the histogram based clustering. Therefore, the second design criterion is that CL_1 must be compact. These two criteria have been used to guide the development of the clustering algorithms.

The second set of clusters, CL_2, is obtained by labeling image pixels with the corresponding color clusters in CL_1. Because there is no spatial information used at the first stage, the pixels in the same cluster in CL_1 can be scattered over the entire image, resulting, in many cases, $|CL_2| >> |CL_1|$. The segmentation algorithm at the second stage iteratively merges the regions in CL_2 based on the color distances between the neighboring regions, the region sizes and the maximum number of clusters in CL_3, the system output that contains the meaningful color regions with respect to objects in the image.

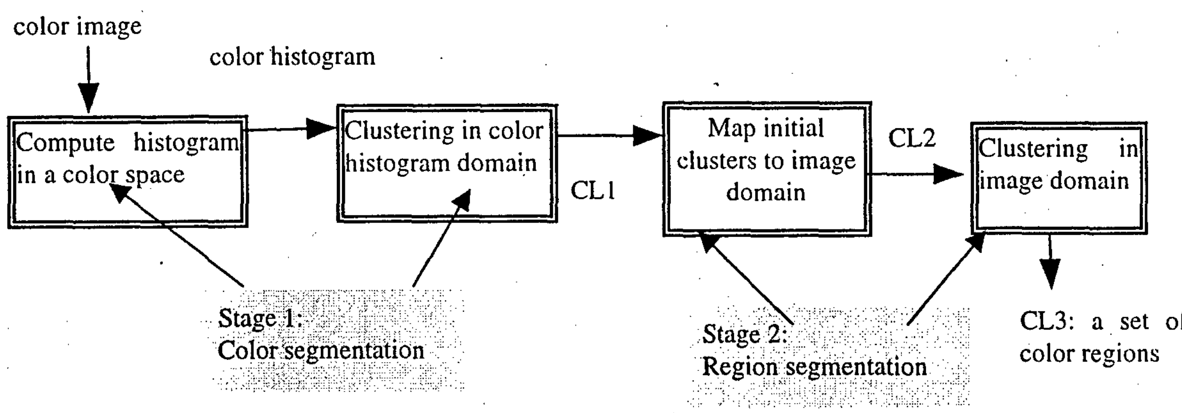


Figure 1. An overview of the color image segmentation system.

This paper is organized as follows. Section 2 will describe two different algorithms that perform the clustering computation in color space and their performance evaluation. Section 3 will present a region segmentation algorithm that groups clusters in the image domain based on color similarity and spatial adjacency. Section 4 presents our experimental results and the performance analysis of each algorithm in the segmentation system.

2. Clustering in color space

Color is the most popular feature used in image retrieval and object recognition applications for color imagery. Color is often described in terms of intensity, luminance, lightness, hue and saturation. Intensity is a measure, over some interval of the electromagnetic spectrum, of the flow of power that is radiated from, or incident upon, a surface. Intensity is what is called a linear-light measure, expressed in units such as watts per square meter. There are a number of color spaces that are defined based on different application criteria and most of these are three-dimensional [3]. In general, for a given color space, we can calculate a color histogram function $F(C_i)$ from a color image I, where C_i is a color in the color space and $F(C_i)$ denotes the number of pixels in *I* that have color C. The dimensions of a color histogram are determined by the color space used to represent the image. For example, in the L*u*v* space, C_i is represented by a vector (l, u, v) and $F(C_i)$ is the number of pixels that have L*u*v* values equal to (l,u,v); in RGB color space, C_i is represented by a vector (r, g, b) and $F(C_i)$ is the number of pixels in *I* that have color equal to (r, g, b). Considering the uncertain nature of classifying similar colors into clusters, we developed two color-clustering algorithms, Max Quantization Error, and Optimal-Cut.

Max Quantization Error

This is a greedy algorithm that aggressively reduces the global color quantization error by creating new clusters at the place where the maximum quantization error is. This algorithm has the following steps:

1. *CL1={} and j=0.*

2. Find the color bin in the histogram with the maximum color frequency, and create a color cluster centered in that bin. Mathematically, we are trying to find a color, $C_{cluster}$, such that

$$F(C_{cluster}) \geq F(C_i) \text{ for all color } C_i \text{ in the image domain.}$$
$$C_j = C_{cluster}, \text{ and } CL1 = CL1 \cup \{C_j\}$$

3. Calculate color quantization error for every bin in the histogram using the following equation:

$$E_i = F(C_i)\|C_i - C_j\|^2$$

4. Find $C_{cluster}$, such that

$$E_{cluster} = F(C_{cluster})\|C_{cluster} - C_j\|^2 \geq E_i \text{ for all i.}$$

   set j=j+1, $C_j = C_{cluster}$, and $CL1 = CL1 \cup \{C_j\}$

5. Recalculate the color quantization error for every color C_i in the histogram using the following equation:

$$E_i = F(C_i)\|C_i - C_{nearest\_cluster}\|^2,$$

   where $C_{nearest\_cluster} \in CL1$ is the cluster that is closest to C_i.

6. Find the color, $C_{cluster}$ with the maximum color quantization error defined in step 4, set j=j+1, $C_j = C_{cluster}$, and $CL1 = CL1 \cup \{C_j\}$

7. Repeat steps 5 to 6 until one of the following conditions is met then go to step 8:
   a. $|CL1| = M$, where M is the desired number of clusters set by the user
   b. the maximum quantization error, $E_{cluster}$ is below a threshold.

8. At the end the loop we have *CL1={ C_j|i=0, 1, ..., M-1}, which* is a set of colors representing M cluster centers. For every color C in the image, C is assigned a color label p, $0 \le p \le M-1$, if

$$D_{color}(C, C_p) = \|C - C_p\|^2 \le D_{color}(C, C_j) = \|C - C_j\|^2,$$

for all j such that $0 \le j \le M-1$.

Optimal-Cut

The *optimal-cut* algorithm adopts a splitting process that generates a number of color cubes in a color space that minimize the global color quantization error. The algorithm is based on the searching of 3D boxes, i.e. a right rectangular prisms, in the color space that give minimized global color quantization error.

We define the quantization error function E_{box} within a 3D box in the color space as follows:

$$E_{box} = \sum_{i \in box} F(c_i)\|c_i - \bar{c}_{box}\|^2$$

where c_i is the 3-dimensional *ith* color vector within *box*, $F(c_i)$ is the color frequency (or the number of pixels that have the color) of c_i, and $\bar{c}_{box}$ is the central color of the *box*, which is calculated as follows:

$$\bar{c}_{box} = \frac{\sum_{c_i \in box} F(c_i)c_i}{\sum_{c_i \in box} F(c_i)}.$$

The optimal-cut algorithm has the following steps

1. Define a minimum 3D box, i.e. a right rectangular prism, box_0 =(C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8) in the color space that encompasses all the colors in the image *I*, where C_i, i = 1, 2, ..., 8 are the 3D coordinates of the eight vertices that uniquely define the color box. Add it to the cluster list CL1 = { box_0 }.

2. For each color box in CL1, search along each of the three coordinate axes in the color space to find a split point that can partition the box into two sub-boxes and where the reduction of the quantization error is maximum in comparison to all the other possible splits within all current color boxes. Assume a color box has coordinates box={(x_1, y_1, z_1), (x_2, y_1, z_1), (x_1, y_1, z_2), (x_2, y_1, z_2), (x_1, y_2, z_1), (x_2, y_2, z_1),(x_1, y_2, z_2), (x_2, y_2, z_2)}. If a split occurred along the x-axis at point x*, where $x_1 < x^* < x_2$, then the two sub-boxes have coordinates, sub-box1={(x_1, y_1, z_1), (x_2, y_1, z_1), (x_1, y_1, z_2), (x*, y_1, z_2), (x_1, y_2, z_1), (x*, y_2, z_1),(x_1, y_2, z_2), (x*, y_2, z_2)}, and sub-box2 = {(x*, y_1, z_1), (x_2, y_1, z_1), (x*, y_1, z_2), (x_2, y_1, z_2), (x*, y_2, z_1), (x_2, y_2, z_1), (x*, y_2, z_2), (x_2, y_2, z_2)}. Splits along the y-axis and the z-axis can be similarly calculated. The reduction of quantization error resulted from the split is calculated as follows:

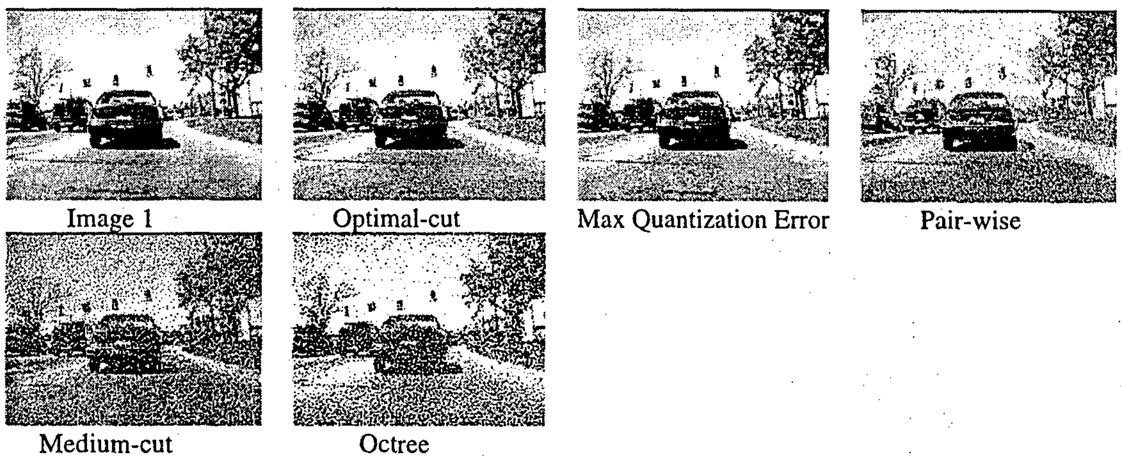
$$E_{box} - E_{sub-box1} - E_{sub-box2}.$$

Based on the definition of the quantization error given above, it can be easily shown that $E_{box} > (E_{sub\text{-}box1} + E_{sub\text{-}box2})$

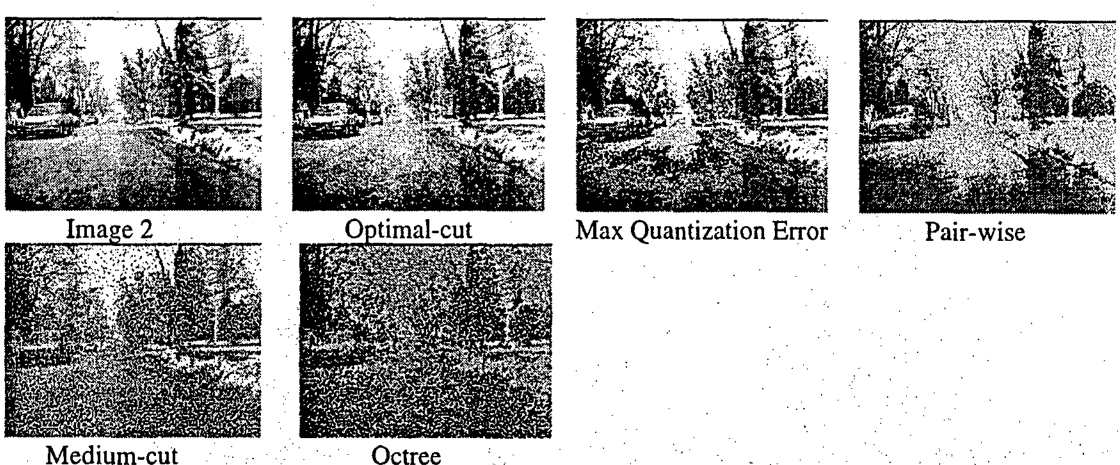
3. Readjust the two sub-boxes so that there are no empty color bins along their boundaries and then add them to the color cluster set CL1.

4. Repeat steps 2 and 3 until the desired number of color clusters is obtained, or the maximum quantization error is below the threshold.

We compared the two color clustering algorithms introduced above with three other algorithms published in the literature, Pair-wise clustering, Medium-cut and Octree [9, 10, 11]. Figure 2 shows the experiment results on three different images generated by the proposed two methods as well as the other three. All algorithms were set to generate 16 different color clusters. From the resulting images we can see that the proposed optimal-cut and max-quantization generated better results than the other three algorithms on all three example images.



(a) Color clustering experiments conducted on image 1.

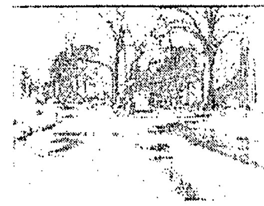


(b) Color clustering experiments conducted on image 2.

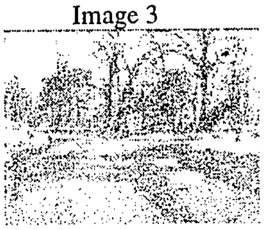
Image 3

Optimal-cut

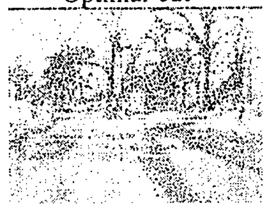
Max Quantization Error



Pair-wise



Medium-cut



Octree

(c) Color clustering experiments conducted on image 3.

Figure 2 Experimental evaluation of various color-clustering algorithms.

We further evaluate the five algorithms by measuring the quantization errors on the resulting images. Table 1 shows the quantization error generated by the five color clustering algorithms on the three images shown in Figure 2.

Table 1: Evaluation of quantization errors of five color clustering algorithms

	Optimal-cut	Max quantization	Medium-cut	Octree	Pair-wise
image 1	14.9	15.0	26.2	36.4	37.9
image 2	31.1	15.8	31.8	37.5	46.4
image 3	15.5	15.4	33.9	32.9	44.0

Both the Max Quantization Error and Optimal-cut algorithms gave much better performance in terms of minimizing the quantization error on all three images than the other three algorithms. Both algorithms gave similar performance in reducing the global color quantization error for image 1 and image 3. However the Max Quantization Error algorithm performed much better on image 2 than Optimal-cut.

According to [9] the Pair-wise clustering algorithm should produce less quantization error than other algorithms such as *uniform*, *popularity*, *median-cut*, *local K-means*, *variance-based*, and *octree*. However, the Pair-wise algorithm needs to maintain a quantization error matrix that can be overwhelmingly large, which makes this algorithm impractical. For example, for a 24-bit color image, every color component has 256 levels. If we use 32 levels to generate the histogram (as most algorithms do), we will have $32^3 = 32{,}768$ bins. Therefore, initially, the quantization error matrix can have $32{,}768^2 = 1{,}073{,}741{,}824$ elements. In these experiments, when we used clustering level 16, the Pair-wise clustering algorithm took significantly longer than the other four algorithms and it did not give better performance.

3. Region segmentation in image domain

Region segmentation is implemented by a spatial clustering algorithm that groups the clusters generated by the color clustering algorithms mentioned in the previous section using various measurements.

When we map the color clusters in CL_1 to the image domain, we obtain a color cluster set CL_2, in which each cluster contains image pixels that are both spatially connected and within the same color cluster in CL_1. In general, one cluster in CL_1 can be decomposed into more than one cluster in CL_2, and therefore CL_2 is much larger than CL_1. This is evidenced by the color clustered images shown in Figure 2. Pixels of the same color can be scattered all over the image domain, which results in many different clusters in CL_2. In general, $|CL_1| << |CL_2|$.

The image segmentation algorithm in the image domain is an agglomerative process that uses the following three parameters:

- color distances among neighboring clusters in the spatial domain,
- cluster sizes and
- maximum number of clusters in CL_3.

After every merge of two clusters, the center of the new cluster is calculated, and the size and the neighbors of the merged cluster are updated. We define the distance between two clusters as the color distance between the centers of the two clusters. During the sequential merging process, an important issue is the order of merging clusters, which can significantly affect the region segmentation result. We have investigated three clustering merging methods. All of the three methods use a common parameter, *max_cls,* to control the maximum number of clusters in CL_3. The three methods differ in the priority of selecting clusters to merge at each iterative step and will depend on the requirements of various applications.

Method 1 attempts to merge the adjacent clusters that are similar in colors. In implementation, we use a control parameter, *cl_diff_th* to denote the color difference threshold. At the first step, the algorithm attempts to merge the neighboring clusters whose color distances are below cl_diff_th. The order of merging is not considered at the this step. If the number of clusters at the end of the first step is greater than *max_cls*, then the algorithm begins the second step. At the second step, the algorithm selects the smallest cluster and merges the cluster with one of its neighbors to which it has the smallest color distance. This merging process repeats until the number of clusters in CL_3 is no more than *max_cls*.

Method 2 considers the size of clusters as the only selection criterion. It selects the smallest cluster and merges the cluster with one of its neighbors to which it has the smallest color distance. The process is repeated until the number of clusters in CL_3 is no more than *max_cls.*

Method 3 considers the color distance as the most important criterion in cluster merging in the image domain. However, the computation required in finding the minimum color distance between two adjacent clusters is quite time consuming if the number of clusters in CL_2 is large. To alleviate the computational burden, the algorithm consists of three passes of merging. At the first pass, it repeatedly merges the smallest clusters with their neighbors that have the closest color distance until the total number of clusters is reduced to a reasonable number. At the second pass, the algorithm selects a pair of two adjacent clusters that has the smallest color distance within the entire image to merge. This process repeats until the top max_cls clusters in size contain a large percentage of the image pixels. Since the largest max_cl clusters already cover the majority of the entire image pixels, the small clusters below the top max_cl should not affect too much the final segmentation result. Therefore at the third pass, the algorithm repeatedly merges the smallest cluster with its closest neighbor in color distance until the total number of clusters in CL_3 is no more than max_cls.

From a computational point of view, Methods 1 and 2 are more efficient than Method 3. However, in many cases, Method 3 generates better results than the other two methods.

4. Implementation, experiments and conclusion

All the algorithms described in this paper have been implemented in C++ under the Windows 2000® operating system. In the implementation, we first converted color from RGB coordinates to L*u*v*.

We have tested the segmentation system on a large number of images from video sequences taken from a moving vehicle. Due to the limited space, we use one image example to illustrate the performance of the proposed clustering system. Figure 3(a) shows an image from a video image sequence taken in a city street scene by a video camera mounted on a moving vehicle. The objects of interest in this application are vehicles in front of the primary vehicle, which has the video camera mounted. In these experiments, we used both Max Quantization Error and Optimal-Cut in the color space clustering and Method 3 in region segmentation. It appears that Optimal-Cut gave better segmentation results in the color space. The results from region segmentation that follow both Max Quantization Error and Optimal-Cut separated the vehicles from their surrounding objects such as road, trees, etc. However, the Optimal-Cut appeared to give better details.

(a) Original image

(b) Color clusters generated by Optimal cut

(c) Color clusters generated by Max Quantization Error

(d) Image segmentation generated by Optimal-cut with 50 clusters.

(e) Image segmentation generated by Max quantization error with 50 clusters.

Figure 3 Image segmentation using optimal-cut and max quantization error.

Figure 4 shows the region segmentation results when different number of clusters were specified. In comparison to the results shown in Figure 3, where cluster number 50 was used, we can conclude that when the number of clusters increases, more details are shown but more region fragments may result. In general Optimal-Cut followed by region segmentation gives effective object segmentation using color features.

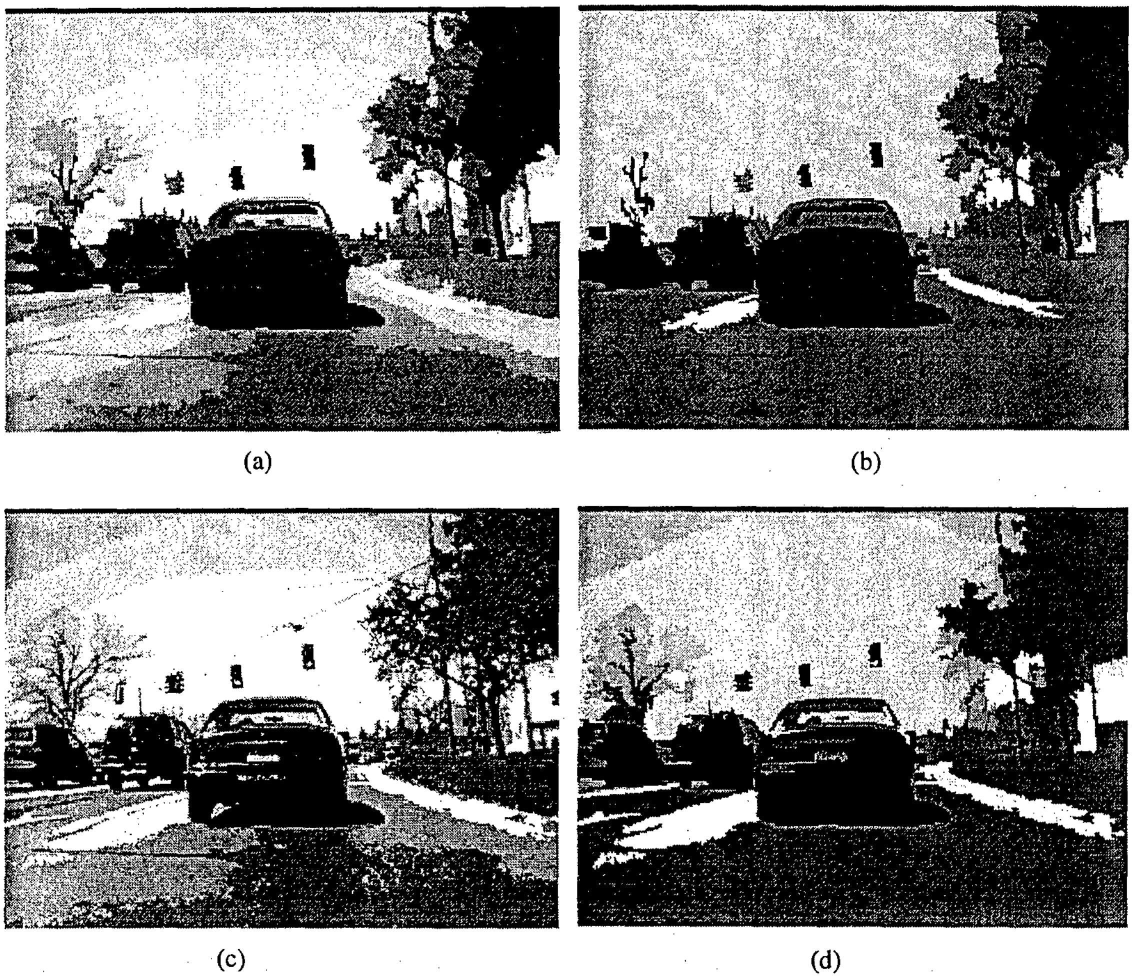

(a) (b) (c) (d)

Figure 4 Image segmentation using different number of clusters in image domain. Max Quantization Error followed by region segmentation with (a) cluster number 100 and (b) cluster number 25. Optimal-Cut followed by region segmentation with (c) cluster number 100 and (d) cluster number 25.

Unlike many existing clustering algorithms, the image segmentation system does not require the knowledge about the number of the color clusters to be generated at each stage and the resolution of the color regions can be controlled by one single parameter, the radius of a cluster. The color image segmentation system has been implemented and tested on a variety of color images including satellite images, car and face images. The system has shown to be both effective and efficient.

5. Acknowledgment

The work is support in part by a SBIR contract from the U. S. Army TACOM.

6. References

[1] J. Hafner, H. S. Sawhney, W. Equitz, M. Flickner and W. Niblack, "Efficient Color Histogram Indexing for Quadratic Form Distance Functions," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 17, No. 7, PP. 729-736, July 1995.

[2] E. Binaghi, I. Gagliardi, R. Schettini, "Indexing and Fuzzy Logic-Based Retrieval of Color Images," *Visual Databse Systems, II*, edited by E. Knuth and L. M. Wegner, Wlsevier Science Publishers B. V., 1992.

[3] Yi Lu Murphey "Image Search and Retrieval Strategies" a chapter in Encyclopedia of Imaging Science and Technology, edited by Dr. Joseph P. Hornak to be published by John Wiley & sons, Inc. 2002

[4] Ewald Hering, "Outlines of a Theory of the Light Sense," Harvard University Press, Cambridge, MA 1964

[5] C. L. Novak and S. A. Shafer, "Color Vision," *Physics-Based Vision COLOR,* pp. 1-10, Jones an Bartlett, 1992

[6] G. D. Finlayson, "Color in Perspective," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 18, No. 10, PP. 1034-1035, October, 1996.

[7] Jean-Michel Jolion, Peter Meer, and Samira Bataouche, "Robust Clustering with Applications in Computer Vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* pp 791-802, August, 1991.

[8] L. Hall, A. Bensaid, L. Clarke, R. Velthuizen, M. Silbiger, nd J. Bezdek, "A Comparison of Neural Network and Fuzzy Clustering Techniques in Segmenting Magnetic Resonance Images of the Brain," IEEE Transactions on Neural Networks, Vol. 3, PP. 672-582, September, 1992.

[9] Velho, L.; Gomes, J.; Sobreiro, M.V.R. , "Color image quantization by pairwise clustering " X Brazilian Symposium on Computer Graphics and Image Processing, 1997, Page(s): 203 -210, 233

[10] P. Heckbert, "Color image quantization for frame buffer display," Computer Graphics, 16(3), pp297-307, July, 1982.

[11] M. Gervautz and W. Purgathofer, "A simple method for color quantization: octree quantization," Proceedings of CGI 88, pp219-231, 1988.

16161

OPSEC REVIEW CERTIFICATION

(AR 530-1, Operations Security)

I am aware that there is foreign intelligence interest in open source publications. I have sufficient technical expertise in the subject matter of this paper to make a determination that the net benefit of this public release outweighs any potential damage.

Reviewer: J. Overholt GS-15 RESEARCH MECH. ENGINEER
Name Grade Title

J. OVERHOLT 15 Nov, 2002
Signature Date

Description of Information Reviewed:

Title: Color Image Segmentation in the Color and Spatial Domains

Author/Originator(s): Dr. Robert Karlsen & Dr. Grant Gerhart

Publication/Presentation/Release Date: June 2003

Purpose of Release: Jun 23-26 2003 Conference titled: The 16th International conference on Industrial & Engineering Applications of Artificial Intelligence and Expert Systems, in Loughborough, UK

An abstract, summary, or copy of the information reviewed is available for review.

Reviewer's Determination (check one)

1. Unclassified Unlimited.

2. Unclassified Limited, Dissemination Restrictions IAW __________

3. Classified. Cannot be released, and requires classification and control at the level of __________

Security Office (AMSTA-CM-XS):

Concur/Nonconcur [signature] 18 Nov 02
Signature Date

Public Affairs Office (AMSTA-CM-PI):

Concur/Nonconcur Margaret Compton Nov 20, 2002
Signature Date

STA 7114-F