A Color Classification Method for Color Images Using a Uniform Color Space

Shoji Tominaga

Faculty of Engineering, Osaka Electro-Communication University
Neyagawa, Osaka 572, Japan

Abstract

The present paper describes a color classification method that partitions color image data into a set of uniform color regions. The ability to classify spatial regions of the measured image into a small number of uniform regions can be useful for several problems including image segmentation and image representation. The input image data are first mapped from device coordinates into an approximately uniform perceptual color space. Colors are classified by means of cluster detection in the uniform color space. The classification process is composed of two stages of basic classification and reclassification. The basic classification is based on histogram analysis to detect color clusters sequentially. The principal components of the color data are extracted for effective discrimination of clusters. At the reclassification stage, the extracted representative colors are reclassified on a color distance. The performance of the method is discussed in an experiment.

1 Introduction

A color classification method is described for partitioning an image into a set of perceptually uniform color regions. Our color classification algorithm accepts high resolution color images as input, and yields a new representation of the data in the form of a set of spatial regions, each described by a single color value. The ability to classify spatial regions of input image into a small number of uniform color regions can be useful for essential problems of color image analysis, including image segmentation and image representation [1-6].

All color information about an image is summarized in a three-dimensional color space (see Figure 1). Uniform color regions in an image plane give rise to clusters in the color space. If the image contains large regions of pixels all having approximately the same color, we have a dense cluster in the color space. Therefore color classification takes place by cluster detection in the color space. Sequential cluster detection causes sequential detection of uniform color regions in the image plane. Selection of the color space is crucial to the color classification problem, because the arrangement and shape of clusters depend on the color space selected. Therefore a perceptually uniform color space should be used.

Colors in pixels are classified on the color specifications in the uniform color space. We propose a color classification process which is composed of two steps of basic classification and reclassification. The basic classification algorithm uses a recursive method to detect clusters of color data. We seek spatial regions in the image that can be classified as a uniform region based on both (1) their similarity in color and (2) their identification as a cluster in the uniform color space. We apply an iterative histogram analysis to the data in the uniform color space. The analysis method is also related to the histogram analysis approach for image segmentation [3]-[5]. Since the algorithm analyzes iteratively one-dimensional histogram from a set of histograms, the approach seems to be simple and easy for automatic detection of clusters in a color space. However it has been pointed out that performance of the detection depends upon the spatial arrangement of clusters in the fixed color coordinates, and so clusters cannot be detected in some cases.

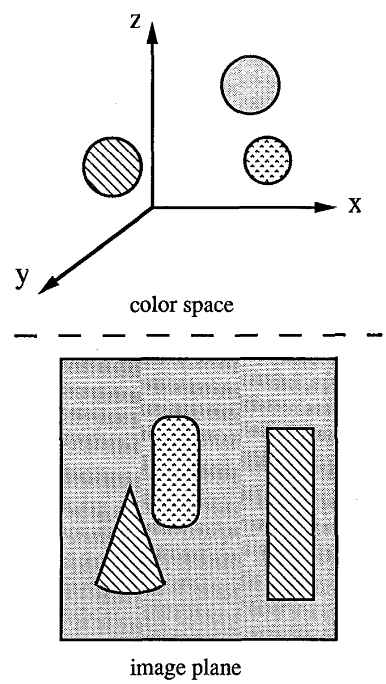

Fig. 1. Color clusters and image partition.

In this paper we do not use any fixed color coordinates, but the coordinates of principal component vectors of color distribution in a uniform color space. The principal components of three-dimensional color data are determined at every iteration of histogram computation, and the analysis is done on the transformed axes. At the stage of reclassification, the extracted representative colors by the basic classification are reclassified on a color distance. We propose reclassification of the extracted representative colors based on such a distance measure as color difference or hue difference to analyze relations of color clusters.

2 Color specifications in a uniform color space

Mapping of a measured image into a perceptually uniform color space is normally based on a nonlinear transformation of observed RGB values. The CIE recommended the CIE-L*a*b* space for a uniform color space which approximates close to the Munsell space in terms of the tristimulus values X, Y, and Z. This space is produced by plotting in rectangular coordinates the quantities L*, a*, and b* defined by

$$L^* = 116(Y/Y_0)^{1/3} - 16 \tag{1}$$

$$a^* = 500[(X/X_0)^{1/3} - (Y/Y_0)^{1/3}] \tag{2}$$

$$b^* = 200[(Y/Y_0)^{1/3} - (Z/Z_0)^{1/3}], \tag{3}$$

where the constant X_0, Y_0, and Z_0 are the tristimulus values of the standard white. L* is termed the metric lightness, which corresponds to about ten times Munsell Value. This space defines a uniform metric-space representation of color so that a perceptual color difference is represented by the Euclidian distance. In real processing, to get the color specifications in the L*a*b* coordinates, we have to transform observed RGB values into the tristimulus values. We can assume that a linear relationship is between the effective reflectances and the tristimulus values. Therefore this mapping is described in the form $\mathbf{p} = \mathbf{T}\mathbf{s}$, where $\mathbf{p} = [X, Y, Z]^t$ and $\mathbf{s} = [\rho_R, \rho_G, \rho_B]^t$. The 3x3 transformation matrix $\mathbf{T}$ can be determined based on real measurement of color samples.

3 Overview of color classification process

Figure 2 shows the outline of our color classification process by a flow chart. The process is composed of two stages of the basic classification by sequential cluster detection in a color space and the reclassification of the extracted representative colors.

At the stage of the basic classification, color clusters in a uniform color space are detected sequentially by a histogram analysis. Image regions with the corresponding uniform colors are extracted, and then labeled in the image plane. Once a color class is determined, the data identified with the class are removed from the data set. The cluster detection is repeated on the remaining image data. This sequential process is closed when no cluster remains in the color space. The closing condition is classified into two cases. In one case, all three histograms are unimodal (i.e., have only one peak) on principal components for the remaining color data. We decide that the data belong to one color cluster.

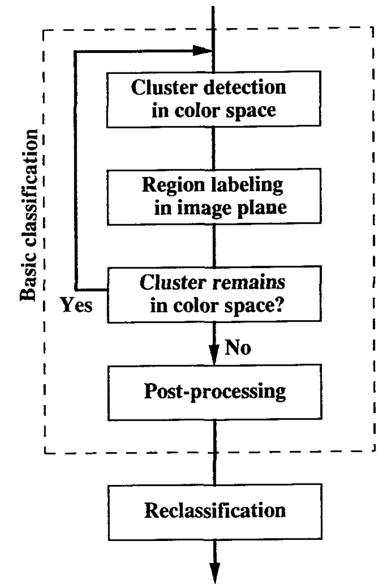

Fig. 2. Outline of color classification process.

After a color label is assigned to the segmented regions with uniform color in the image plane, the sequential process ends. In a second case, if histograms have no well-defined peaks, we can conclude that the remaining color data are decreased and sparse in the color space. By regarding the remaining image regions as noise, a finishing process is executed. These noisy regions are merged into the neighboring regions with color labels.

In the post-processing part of the basic classification, we first treat the remaining pixels without color labels, and apply smoothing operation to all the segmented regions. These processings are a type of region merging. Next representative colors are determined for each label of the classified colors. Thus we obtain a fundamental set of different colors, which compose an image, and the segmented image with uniform color regions of the basic colors.

The basic classification does not use explicitly a distance measure of color. Color difference and hue difference are useful perceptual measures for color image analysis. For instance the hue difference is considered to be useful for identifying objects in a scene with color variation due to shade and shadow. In this stage we reclassify the previously extracted fundamental set of colors on the basis of a color distance. We adopt a pattern-clustering algorithm.

4 Basic classification

4.1 Coordinate transformation of color clusters

Performance of the cluster detection depends heavily upon the spatial arrangement of clusters in the color space. If the clusters are globular and do not overlap each other, a fixed coordinate system can be used for extracting them. However in the orthogonal coordinate system, parallel diagonal clusters will overlap when projected on either fixed axis. Therefore the clusters cannot be distinguished by one-dimensional thresholding with respect to each coordinate alone. Here we propose use of the principal component coordinates of color distribution.

The principal component coordinates are obtained from the eigenvalues and eigenvectors of the covariance matrix of color data vectors. We note that the mean vector and the eigenvectors describe the position and the orientation of whole color clusters. Now define a three-dimensional column vector $\mathbf{m}$ and a 3x3 matrix $\mathbf{R}$, respectively, to be the mean vector and the covariance matrix of a color vector $\mathbf{c}$ in the original orthogonal coordinate system (x, y, z) as

$$\mathbf{m} = E[\mathbf{c}], \quad \mathbf{R} = E[(\mathbf{c}-\mathbf{m})(\mathbf{c}-\mathbf{m})^t], \tag{4}$$

where the expectation E is taken over color data on the permitted image regions. An orthogonal decomposition of $\mathbf{R}$ gives us the eigenvalues $\lambda_1, \lambda_2, \lambda_3$, arranged in descending order ($\lambda_i \geq \lambda_{i+1}$), and the corresponding eigenvectors $\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3$. These eigenvectors determines the unit vectors of the principal component axes for the given data set. Then we have a linear transformation for transforming the color vector $\mathbf{c}$ from the original coordinate system into a vector $\mathbf{c}'$ into the new coordinate system of the principal component axes. This transformation is expressed in the equation

$$\mathbf{c}' = \mathbf{U}^t(\mathbf{c} - \mathbf{m}), \tag{5}$$

where $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3]$. The new color features c'_1, c'_2, and c'_3, defined by the entries of $\mathbf{c}'$, have zero means and variances λ_1, λ_2, and λ_3.

4.2 Algorithm of cluster detection

Figure 3 shows the flow chart of an iterative process for automatically extracting a color cluster. The procedure at each step is described below.

1. First we initialize a color label array *img(i, j)* and a mask array *imgm(i, j)*. The numbers i and j denote the indices of rows and columns in an image. The color label array stores the color numbers for pixels already classified. Whenever a color cluster is newly extracted, a color number is generated, and the corresponding color regions are labeled on *img(i, j)*. The mask array determines whether an image pixel is still considered to be part of the data set for cluster detection, or whether it has already been classified.

2. At every iteration for extracting a cluster, we compute the principal components of color distribution in a uniform color space. This computation is done for the regions of image data whose mask is set.

3. We project the color data onto the principal components axes by the transformation (5), and compute the histograms of the color features c'_1, c'_2, and c'_3, one by one.

4. The histogram is analyzed to find significant mountains from large peaks and valleys. Candidates for the significant mountains are selected, and a function is computed for evaluating each candidate. The criterion function for significant mountains is described in the next subsection. We evaluate the significant mountains one by one from the first histogram ($i = 1$). The mode of the color distribution is then classified into the following three cases.

5. In one case, if a histogram is multimodal, the most significant mountain in the histogram is selected. A pair of thresholds is determined as the color features corresponding to two valleys at the sides of the mountain. The image is split using the thresholds. A mask for describing the extracted subregions is created on *imgm(i, j)*. We further continue with cluster detection with respect to these subregions.

6. In a second case, if the first histogram is noisy and has no well-defined peaks, then it meets the closing condition of the sequential color classification. The remaining pixels without labels become too sparse to create a cluster in the color space. The finishing process is executed.

7. In the third case, one of the histograms is unimodal. If the first histogram is unimodal, the succeeding second and third histograms are analyzed. If all histograms are unimodal, then we extract the color data which are decided to belong to one color cluster. The extracted pixels are labeled on *img(i, j)*. Detection of a color cluster ends. If no region without color labels remains, then the finishing process is executed.

4.3 Analysis of histogram

The exact extraction of a color cluster relies on the exact detection of significant mountains on one-dimensional histogram, computed along the principal component vectors in a uniform color space. To find significant mountains we specify several parameters for describing the shape of a mountain. These parameters include a peak height, a valley bottom height, and their positions

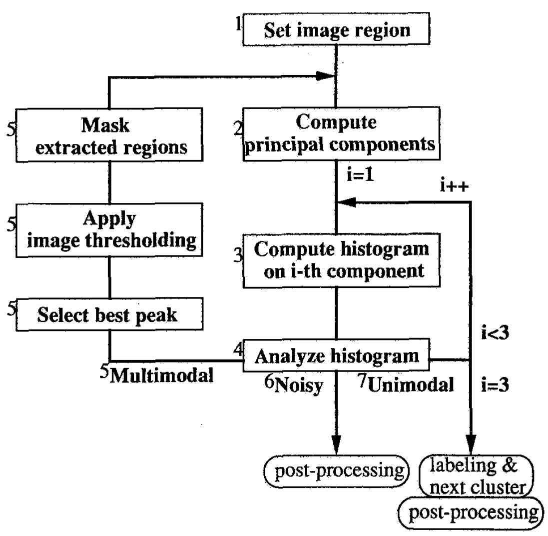

Fig. 3. Iterative color cluster detection.

on a color feature axis (principal component vectors). The domain of a mountain to be detected ranges between two significant valleys at the sides of the peak. Hence thresholds are the feature values corresponding to the positions of the valley bottoms.

A set of significant mountains are determined by taking account of the heights of peaks and valley bottoms, and then a criterion function is computed to select the most significant mountain. We use the function

$$f = \frac{A_p}{A_t}\frac{100}{fwhm}, \tag{6}$$

where A_p indicates the area of a histogram between the two valley bottoms, and A_t is the overall area of a histogram (the total number of masked pixels). Moreover $fwhm$ denotes the full-width at half-maximum of the peak. We normalize the range of feature value to 100. Therefore this function describes the mountain shape by combining the size (A_p/A_t) and the sharpness $(100/fwhm)$ of a mountain.

4.4 Post-processing

(1) Processing of the remaining pixels after cluster detection:
First we determine the regions of eight(or four)-connected image pixels for each color label on the color label array *img(i, j)*, and compute color specifications for all the segmented regions. If there are still pixels without a color label, those are merged into the neighboring regions with color labels on *img(i, j)*.

(2) Region smoothing:
Smoothing is applied to all the segmented image regions with color labels. If the area of a region is less than a certain threshold, and also the minimum color difference to one of the neighbors is less than a certain threshold, then the color label of the objective region is replaced with the neighbor's one on *img(i, j)*.

(3) Determination of representative colors:
The average color specifications are computed over all pixels with the same label.

(4) Computation of accuracy:
We define a criterion function to evaluate the performance of the basic color classification. An average color difference is computed between the original image and the estimated image with the representative colors.

5 Reclassification

The problem of color classification based on a distance may be treated as a mathematical pattern-clustering problem [7]. However we believe that a pure clustering approach using a distance function to the original image data is too cumbersome computationally to implement. We apply reclassification by a distance function to the representative colors extracted by the basic classification. The basic procedure is shown in [8].

Suppose that we have a set of K representative colors $\{\mathbf{m}_1, \mathbf{m}_2, ..., \mathbf{m}_K\}$. Let select a threshold T for a color difference. We choose the first cluster center $\mathbf{a}_1$ in the color space as $\mathbf{a}_1 = \mathbf{m}_1$. Next, we compute the color difference from $\mathbf{m}_2$ to $\mathbf{a}_1$. If this difference exceeds T, a new cluster center $\mathbf{a}_2$ is created as $\mathbf{a}_2 = \mathbf{m}_2$. Otherwise $\mathbf{m}_2$ is assigned to the domain of the class $\mathbf{a}_1$. In a similar fashion, the color difference from each representative color $(\mathbf{m}_3, \mathbf{m}_4, ...)$ to every established cluster center is computed and thresholded. A new cluster is created if all of these distances exceed T, otherwise the color is assigned to the class to which it is closest.

Moreover the same procedure can be applied to color classification based on a hue difference. The hue difference, which is originally defined by an angle, can be approximated with a distance by the following transformation. Let us first define a two-dimensional chromaticity plane of hue and saturation as the orthogonal coordinates $(x,\ y)$ in a uniform color space $(x,\ y,\ z)$. If we normalize the saturation to unity, the chromaticity coordinates are projected on a unit circle, and the hue angle is approximated as the distance. This normalized coordinates are given by the transformation

$$(x',\ y') = (\frac{x}{\sqrt{x^2+y^2}}, \frac{y}{\sqrt{x^2+y^2}}). \tag{7}$$

The Euclidian distance in the coordinate system $(x',\ y')$ approximates the hue difference between two colors.

6 Experimental results

Figure 4 shows a picture of three blocks used in an experiment. These blocks were made of paper, and the picture was taken under illumination with a flood lamp for daylight photograph. The scene contains a yellow cone, a red column, and a blue rectangular prism, but they have various shades by lighting. The background is a emerald green paper on which shadows of the objects are clearly seen. The features of this picture are strong shades and shadows but little highlights. A 175x185 digital image has been measured with a drum scanner, and represented in the CIE-L*a*b* color system. Figure 5 shows the color distribution in the color space. Color clusters have been extracted sequentially, and after sixteen iteration, the basic classification process has been terminated. Figure 6 demonstrates the image segmentation results into uniform color regions.

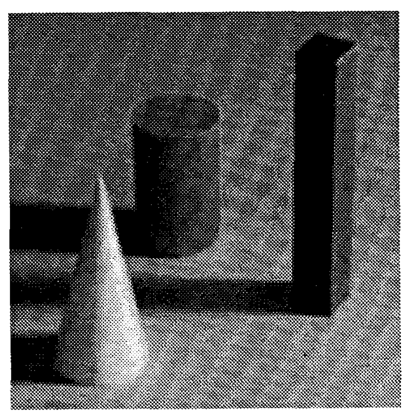

Fig. 4. Picture of three paper blocks.

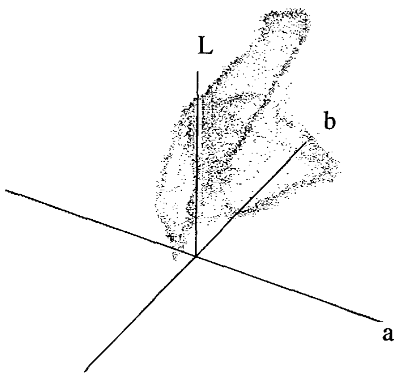

Fig. 5. Color distribution in the L*a*b* color space.

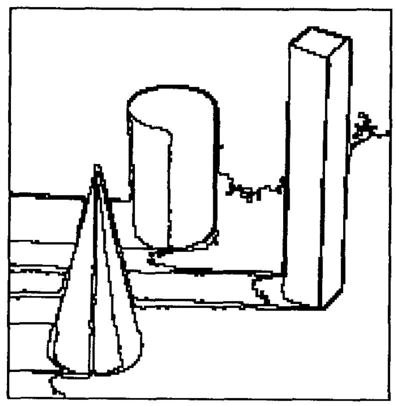

Fig. 6. Segmented image by basic color classification.

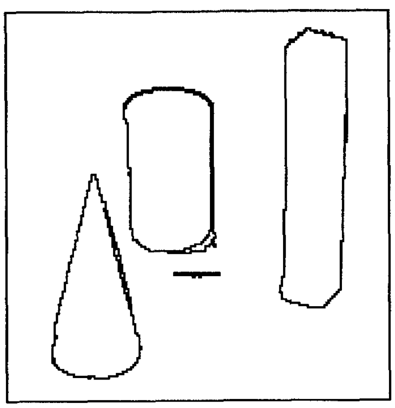

Fig. 7. Reclassification results using color difference and hue difference.

Next the representative colors have been reclassified based on the color distances. The clustering algorithm is executed using the color difference of a threshold $T = 17$. Further these results have been classified based on the hue difference. Figure 7 shows the finally segmented image by using the hue difference $T = 0.3$. All the colors are summarized into only 5 representative colors. Almost all the shades and shadows are removed, and only the silhouette of the objects are detected.

7 Conclusion

The present paper has described a color classification method for partitioning an image into a set of perceptually uniform color regions. The image data are first mapped into an approximately uniform perceptual space. In this space color clusters are detected sequentially. The process of color classification is composed of two stages of the basic classification and the reclassification. In the basic classification, color clusters are extracted by an iterative analysis of one-dimensional histograms. In the reclassification, the representative colors for the extracted clusters are classified on a color distance. The former has an advantage of stable detection of dense clusters, though it takes much computation time. The latter has an advantage of simplicity of the algorithm. The experimental results have shown that a fundamental set of colors composing an image with shades and shadows is extracted at the basic classification stage, and the objects in the original image are extracted at the reclassification stage.

References

[1] R. Nevatia, Machine Perception, Prentice-Hall, Englewood Cliffs, NJ, 1982.

[2] M. D. Levine, Vision in Man and Machine, McGraw-Hill, New York, 1985.

[3] R. Ohlander, K. Price, and D. R. Reddy , Picture segmentation using a recursive region splitting method, Computer Graphics and Image Processing, vol. 8, pp. 313-333, 1978.

[4] Y. Ohta, T. Kanade, and T. Sakai, Color information for region segmentation, Computer Graphics and Image Processing, vol. 13, pp. 224-241, 1980.

[5] S. Tominaga, A color classification algorithm for color images, Proc. 4th British Pattern Recognition Soc. Int. Conf. , J. Kittler (Ed.), Springer-Verlag, New York, 1988.

[6] G. J. Klinker, S. A. Shafer, and T. Kanade, Image segmentation and reflectance analysis through color, SPIE vol. 937 Applications of Artificial Intelligence VI, pp. 229-244, 1988.

[7] R. M. Haralick and G. L. Kelly, Pattern recognition with measurement space and spatial clustering for multiple image, Proc. IEEE, vol. 57, pp. 654-665, 1969.

[8] Y. H. Pao, Adaptive Pattern Recognition and Neural Networks, Addison-Wesley, Reading, MA, 1989.