Unsupervised Classification of Complex Color Texture Images

Raimondo Schettini and Andrea Pessina
Istituto Tecnologie Informatiche Multimediali, Consiglio Nazionale delle Ricerche
Milano, Italy

Abstract

Much work has been done on texture feature definition for grey-level image analysis but, surprisingly, color texture has received very little attention. In this work we present a novel and effective way of analyzing complex color texture. Experimental results on a database of images of colored granites confirm the feasibility of the method.

Introduction

In the context of computer analysis, texture analysis has very often been limited to monochromatic images,[2,10] while many applications require that texture description include both color and structural aspects. Recently Tan and Kittler have developed two procedures for color texture description in supervised image classification.[6,7] In the first, which is very computational intensive, single color bands are processed independently by a set of orthogonal masks derived from the Discrete Cosine Transform, for a total of 24 features. In the second procedure information regarding color and information regarding texture are processed separately: the former is derived from the analysis of color histograms, while the latter is computed on a grey-level image obtained by a linear combination of the color channels (for a total of 14 features). This approach can not fully exploit color information since histograms only convey coarse global information.

This paper focuses on the definition of a new, rather small, effective color texture feature set for the unsupervised classification and segmentation of complex color texture images. The key idea is to use the orientation difference between two vector colors in a orthonormal color space as their color difference measure. We take, for each pixel, the angular color difference between its own vector color and the average vector color computed in the surrounding neighborhood, producing a grey-level "color contrast image." A set of texture features is then computed from the loworder spatial moments of the area around each pixel of the color contrast image. A clustering algorithm is then introduced to process the texture features together with the pixel color (for a total of nine features), to produce an image pixel classification.

Definition of the Texture Features

Color Difference Description

In order to effectively describe the color aspect of texture images we need a concise representation of the color related to change in the material composition of the scene. Since color texture is usually composed of a spatial arrangement of few classes of color pigment, we use a color representation scheme and a color difference measure derived from a pigmentation model of dielectric srfaces.[5] If we represent colors as vectors in the camera RGB color space, we can use the orientation difference between two vector colors as their color difference measure (Figure 1).

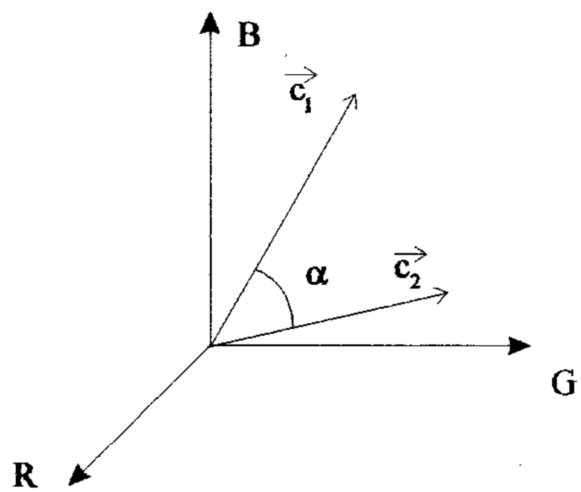

Figure 1. Included angle between two vector colors

We use the included angle between vectors $\mathbf{c}_1$ and $\mathbf{c}_2$ to quantify their difference in orientation:

$$\alpha(c_1, c_2) = \frac{2}{\pi}\arccos\left(\frac{c_1 \cdot c_2}{|c_1||c_2|}\right)$$

This description scheme can be generalized to use any set of orthonormal color basis vectors, such as those defined by principal component analysis on a representative sample of pigment colors.

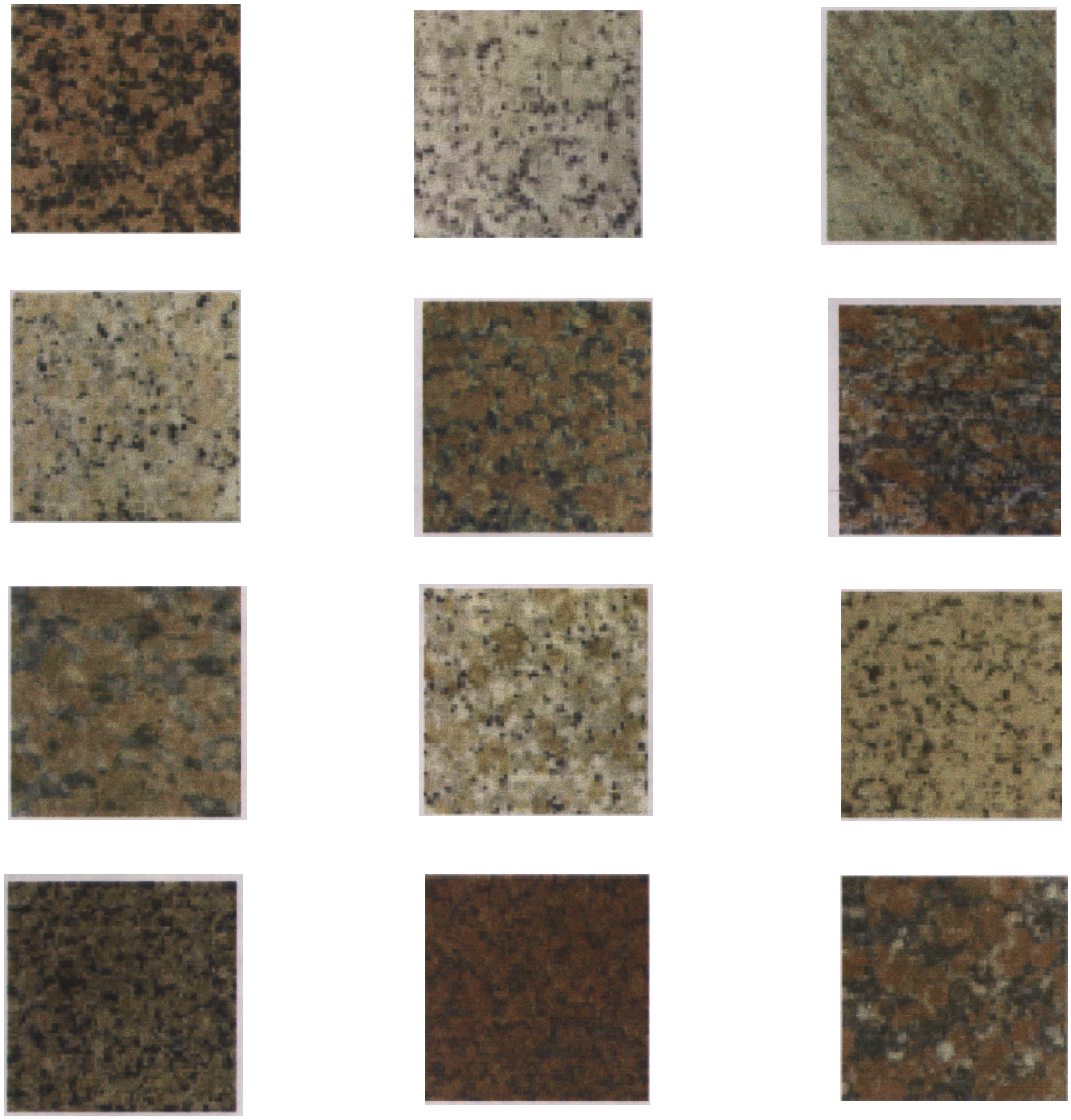

Figure 2. The database of colored granites.

It can be shown that, consistent with the above definition of color difference, the definition of the mean of a color sample, $\mathbf{c}_1$, ..$\mathbf{c}_n$, can be expressed as:

$$\bar{c}_i = \frac{1}{n}\sum_{i=1}^{n}\frac{c_i}{|c_i|}$$

Definition of the Color Contrast Image

We use the color difference definition described above to define a "color contrast image" in which color contrast information is concisely and effectively coded in grey-levels.

Our algorithm first computes an image in which each pixel is labeled with the angular color difference between its own vector color and the average vector color computed in a predefined pixel neighborhood. In Formula, the color contrast image C(x,y) is defined as:

$$C(x,y) = \frac{2}{\pi}\arccos\left(\frac{c(x,y)\bullet\bar{c}(x,y)}{|c(x,y)||\bar{c}(x,y)|}\right)$$

Computing Color Texture Features

Over the past few years many methods for extracting texture features directly from image statistics or from the spatial frequency domain have been proposed.[10] In particular, extracting texture features by spatial filtering appears a well suited method for unsupervised texture analysis. We have therefore adopted a moment-based approach for obtaining texture features directly from the color contrast image.

Moment values have already been used to characterize grey-level textures, e.g. ref. 4 and 8. In 1994 Tuceryan showed that discrete low-order moments computed in a small window around each pixel define a feature vector that allows the correct segmentation of a number of binary iso-order-statistics and grey-level texture images.[9] The moments of a given pixel (i,j) of the color contrast image are given by the following equation:

$$M_{pq}(i,j) = \frac{1}{W^2}\sum_{i=k-W/2}^{i=k+W/2}\sum_{j=1-W/2}^{j=1+W/2} c(i,j)\left[\left(\frac{i-k}{W/2}\right)^p\left(\frac{i-1}{W/2}\right)^q\right]$$

where integrals are approximated by discrete sums, and the terms within brackets are the normalized coordinates of the pixels in the window of width W centered in the pixel (i,j).

The computation for a given pixel of the discrete moments over a finite square window can be interpreted as a convolution of the image with a mask. The size of the mask can be interpreted as a scale parameter to fit the coarseness of the texture to be detected.

The moments alone do not provide good texture features for some images, consequently macro-statistical features are obtained over a L × L larger window Q using a hyperbolic transfer function followed by an averaging step:

$$F_{pq}(i,j) = \frac{1}{L^2}\sum_{i=k-L/2}^{k+L/2}\sum_{j=k-L/2}^{k+L/2}\left|\tanh\left(0.01(m_{pq}(i,j)-\overline{m}_{pq})\right)\right|$$

where $\overline{\mathbf{m}}_{pq}$ is the mean of the computed feature. The result of this processing step is a set of "feature images" in which each pixel is a measure of the texture present in the corresponding location of the original image.

Clustering

In this work a K-means clustering algorithm is used to classify the image pixels in the multi-dimensional feature domain.[1] The number of classes in the image is estimated *a priori*. In order to reduce computing time, we subsample the image, considering about ten percent of the pixels, randomly chosen. These samples are processed by the K-means algorithm in order to partition the feature space. We then classify each pixel of the image, using a minimum distance classifier.

Experimental Results

Varying image mosaics were constructed, using the database of colored granites shown in Figure 2, to test the performance of our classification algorithm. The mosaics were composed of four 128 × 128 resolution—24-bit precision—pictures, and have been designed to test the capability of the algorithm to discriminate tiles that appear similar in color, or texture coarseness, or both.

According to the algorithm devised for each image pixel the average color, the difference color values and the moments <m_{00}, m_{10},m_{01},m_{11},m_{02},m_{20},m_{22}> were computed on a 21 × 21 window, while the macro-statistical textural features are computed on a 49 × 49 window. Nine features in all are computed for each pixel. We have determined the window sizes heuristically; however their values were kept constant for all the images used in this experiment.

Figure 3 shows an example of processing of a color texture mosaic. Ignoring the complexity of the images analyzed, it may be said that the classification errors are mainly due to the fact that the features processed for the boundary pixels are computed in windows crossing two or more types of different textures, and no spatial constraints are considered in the processing. If pixels near the boundaries are not taken into account, we have a very good classification rate.

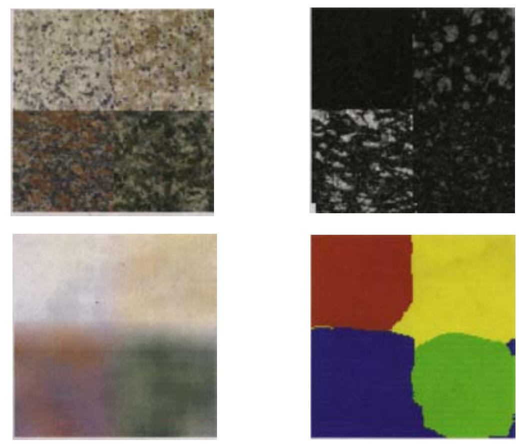

Figure 3. a) the original mosaic b) the "color difference image," e) the "average color image," and d) the classification result.

Conclusions

We have focused here on the definition of an effective color texture feature set for the unsupervised classification and segmentation of complex color texture images. The results are very encouraging. The test images used in this work were kindly provided by Dr. M. Petrou and Dr. M. Mirmehdi of the Department of Electronic and Electrical Engineering of the University of Surrey (UK).

References

1. B. Coleman, H.C. Andrews "Image segmentation by clustering" *Proc. IEEE*, Vol. **67**, pp. 773-785, 1979.
2. J. M. H. Du Buf, M. Kardan, M. Spann, "Texture feature performance for image segmentation," *Pattern Recognition*, Vol. **22** pp. 291-309, 1990.
3. R. Gershon, "Aspects of perception and computation in color vision," *Computer Vision, Graphics, and Image Processing*, Vol. **32**, pp. 224-277, 1985.
4. K. I. Laws, "Textured image segmentation," Ph.D. Thesis, University of Southern California, 1980.
5. K-K.- Sung "A vector signal processing approach to color," *MIT Technical Report AIM* 1349, 1992.
6. T. S. C. Tan, J. Kittler, "On colour texture representation and classification," *Proc. 22th Int. Conference on Image Processing, Singapore*, pp. 390-395, 1992.

7. T. S. C. Tan, J. Kittler, “Colour texture classification using features from colour histogram,” *Proc. 8th Scandinavian Conf. on Image Analysis*, SCIA '93, pp., 1993.
8. M. Tuceryan A. K. Jain, “Texture segmentation using Voronoi polygons,” *IEEE Trans. Pattern Analysis and Machine Intelligence*,Vol. **12**,pp.211-216, 1990.
9. M. Tuceryan, “Moment-based texture segmentation,” *Pattern Recognition Letters*, Vol. **15**, pp. 659-668, 1994.
10. M. Tuceryan A. K. Jain “Texture Analysis,” Handbook of Pattern Recognition and Computer Vision (Eds. C.H. Chen, L.F. Pau, P.S.P. Wang), pp. 236-276, 1994.