

PERCEPTUALLY-TUNED MULTISCALE COLOR-TEXTURE SEGMENTATION

Junqing Chen, Thrasyvoulos N. Pappas

Electrical and Computer Engineering Dept.
Northwestern University, Evanston, IL 60208

Aleksandra Mojsilovic, Bernice E. Rogowitz

IBM T. J. Watson Research Center
Yorktown Heights, NY 10598

ABSTRACT

We present a perceptually-tuned multiscale image segmentation algorithm that is based on spatially adaptive color and texture features. The proposed algorithm extends a previously proposed approach to include multiple texture scales. The determination of the multiscale texture features is based on perceptual considerations. We also examine the perceptual tuning of the algorithm and how it is affected by the presence of different texture scales. The multiscale extension is necessary for segmenting higher resolution images, and is particularly effective in segmenting objects shown in different perspectives. The performance of the proposed algorithm is demonstrated in the domain of photographic images.

1. INTRODUCTION

Many Content-Based Image Retrieval (CBIR) systems rely on scene segmentation for retrieval [1, 2]. The focus of this paper is on segmentation of images of natural scenes based on color and texture. Segmentation of natural images is particularly difficult because, on the one hand, it is impossible to separate the color and spatial frequency components of each texture, and on the other, textures that appear uniform to the human eye exhibit nonuniform statistical characteristics due to effects of lighting, perspective, etc. Thus, the problem of combining spatial texture and color to obtain segmentations that are consistent with human perception is quite challenging. The key to addressing this problem is in combining perceptual models and principles about the processing of texture and color information with an understanding of image characteristics. Although significant effort has been devoted to understanding perceptual issues in image analysis (e.g., [3–5]), relatively little work has been done in applying perceptual principles to complex scene segmentation (e.g., [6]).

In [7, 8], we presented an image segmentation algorithm that is based on spatially adaptive color and spatial texture features. The perceptual aspects of this algorithm were further developed in [9, 10], while in [10] we considered perceptual tuning of the algorithm based on subjective tests. In this paper, we consider multiscale feature extraction and discuss how the different texture scales affect the perceptual tuning of the algorithm. Our earlier work focused on small thumbnail images. Multiple scales become necessary when segmenting higher resolution images, and are especially useful for segmenting objects shown in different perspectives.

The focus of this work is in the domain of photographic images. The subject matter includes nature, buildings, people, pure

This material is based upon work supported by the National Science Foundation (NSF) under Grant No. CCR-0209006. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

textures, objects, indoor scenes, etc. As we explained in [7, 8], the key to the success of the proposed approach is the recognition of the fact that it is not necessary to obtain a complete understanding of a given image: In many cases, the identification of a few key segments (such as “sky,” “mountains,” “people,” etc.) may be enough to classify the image in a given category.

In Section 2, we review the segmentation algorithm. The multiscale feature extraction is presented in Section 3. Section 4 discusses the perceptual tuning of the algorithm, and Section 5 presents segmentation results.

2. SEGMENTATION ALGORITHM OVERVIEW

In this section, we review the basic elements of the adaptive color-texture segmentation algorithm that was presented in [8, 9]. The flow chart of the algorithm is shown in Fig. 1. The algorithm is based on two types of spatially adaptive features. One describes the local color composition, and the other the spatial characteristics of the grayscale component of the texture. These features are first developed independently, and then combined to obtain the overall segmentation.

The color composition features consist of the (spatially adaptive) dominant colors and associated percentages in the vicinity of each pixel. The use of spatially adaptive dominant colors reflects, on the one hand, the fact that the human visual system (HVS) cannot simultaneously perceive a large number of colors, and on the other, the fact that image colors are spatially varying. The spatially adaptive dominant colors are obtained using the adaptive clustering algorithm (ACA) for segmentation [11]. The color feature representation is as follows:

$$f_c(x, y, N_{x,y}) = \{(c_i(x, y, N_{x,y}), p_i(x, y, N_{x,y}))\}, \\ i = 1, \dots, M, p_i(x, y, N_{x,y}) \in [0, 1] \} \quad (1)$$

where each of the dominant colors, $c_i(x, y, N_{x,y})$, is a three dimensional vector in *Lab* space and $p_i(x, y, N_{x,y})$ is the corresponding percentage. $N_{x,y}$ denotes the neighborhood around the pixel at location (x, y) and M is the total number of colors in the neighborhood. A reasonable choice is $M = 4$. Finally, a perceptual metric (OCCD) [12] is used to determine the similarity of two color feature vectors.

The spatial texture features describe the spatial characteristics of the grayscale component of the texture, and are based on a multiscale frequency decomposition such as the steerable pyramid [13] or the Gabor transform [14]. Such decompositions have been widely used as descriptions of early visual processing in mammals. We use the local median energy of the subband coefficients as a simple but effective characterization of spatial texture. Median operators tend to respond to texture within uniform regions and suppress responses associated with transitions between regions. In [9]

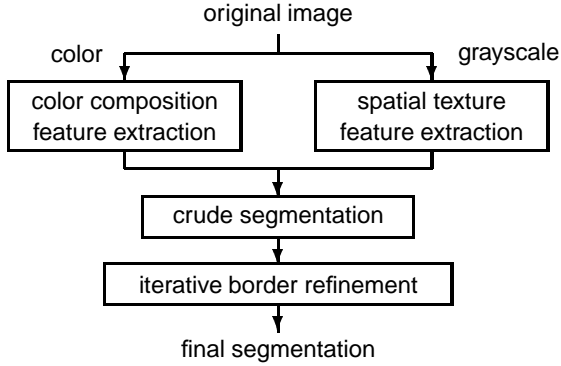


Fig. 1. Schematic of segmentation algorithm

we used a one-level steerable filter decomposition with four orientations, as shown in Fig. 2 (left). The texture features then consist of a classification of each pixel into one of the following categories: *smooth, horizontal, vertical, +45°, -45°, and complex*.

The spatial texture feature extraction consists of two steps. First we classify pixels into smooth and nonsmooth categories. Then we further classify nonsmooth pixels into the remaining categories. Let $s_0(x, y)$, $s_1(x, y)$, $s_2(x, y)$, and $s_3(x, y)$ represent the subband coefficient at location (x, y) that corresponds to the horizontal, $+45^\circ$, vertical, and -45° slope directions, respectively, as shown in Fig. 2 (left). We use $s_{\max}(x, y)$ to denote the maximum absolute value of the four coefficients, and $s_i(x, y)$ to denote the subband index that corresponds to that maximum. A pixel (x, y) is classified as smooth if the median of $s_{\max}(x', y')$ over a neighborhood of (x, y) is below a threshold T_0 . In [8] this threshold was determined using a two-level K -means over the image. As we showed in [10], this threshold can be determined by subjective tests. If the pixel is nonsmooth, then it is further classified as follows. We compute the percentage for each value (orientation) of the index $s_i(x', y')$ in the neighborhood of (x, y) . If the maximum of the percentages is higher than a threshold T_1 (e.g., 42%) and the difference between the first and second maxima is greater than a threshold T_2 , (e.g., 12%), then there is a dominant orientation in the window and the pixel is classified accordingly. Otherwise, the pixel is classified as complex. The first threshold ensures the existence of a dominant orientation and the second ensures its uniqueness. Again, these thresholds can be determined by subjective tests. The use of maximum is due to the fact that neighboring subband filters typically have significant overlap (e.g., in the steerable filter decomposition) and the maximum carries significant information about the texture orientation.

The segmentation algorithm combines the color composition and spatial texture features to obtain segments of uniform color texture. This is done in two steps. The first relies on a multi-grid region growing algorithm to obtain a crude segmentation. The segmentation is crude due to the fact that the estimation of the spatial and color texture features requires a finite window. The second uses an elaborate border refinement procedure, which progressively relies on the color composition features to obtain accurate and precise border localization.

3. MULTISCALE FEATURE EXTRACTION

In our previous work [7,9,10], we used a one-level decomposition. However, humans perceive texture at different scales. In fact, the HVS can perceive multiple scales at the same time. This is done

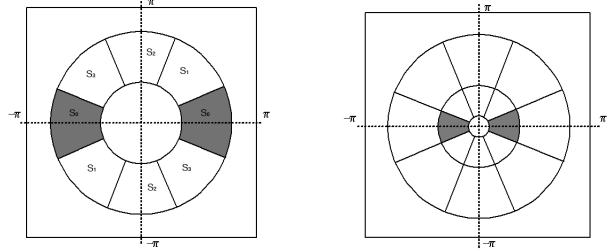


Fig. 2. Steerable Filter Frequency Response. Top: One-level Decomposition. Bottom: Two-level Decomposition

through the use of multiple narrowly tuned spatial frequency channels [15]. Thus, it is important that a computer-based segmentation algorithm be able to detect textures at different scales. While it is difficult to capture texture at multiple scales in thumbnail images, multiscale feature extraction becomes necessary for higher resolution images. Multiple scale analysis can also help capture an object in different perspectives as one uniform object. In this section, we will show how to combine features from different scales into an overall texture classification.

One of the key ideas in the proposed adaptive color-texture segmentation approach is to keep the number of texture parameters small, so that they can be robustly estimated from a limited number of pixels that may be available in each region. Even if the regions are large, the changing texture characteristics that are typical in natural images also dictate that we be able to estimate the texture parameters using a small window. Thus, we would like to keep the texture categories the same as we consider multiple scales.

Our multiscale feature extraction scheme is motivated by the following observations. A texture may be smooth at a finer scale, horizontal at a coarser scale, and smooth again at an even coarser scale. In such a case, the texture will be perceived as horizontal. If on the other hand, a texture is horizontal at one scale and vertical at another, then a human could detect both orientations. In such a case, it would make more sense to classify the texture as complex (given the above texture categories). Finally, if a texture is complex at one scale and horizontal at another, the horizontal orientation is more likely to dominate the human perception.

Based on the above observations, we propose the following rules for extending the one-level texture feature extraction method to multiple scales:

1. For each scale, use the texture extraction method described in the previous section.
2. If downsampling is performed in the multiscale decomposition, upsample the texture class images obtained at each scale to the original image size, so that the texture class images from all scales have the same size.
3. Combine the texture classes of different scales using the following rules:
 - A pixel is classified as smooth only if it is classified as smooth at all of the scales.
 - A pixel is classified as horizontal, vertical, $+45^\circ$, or -45° , if all the scales are consistent, where classification in any given direction at one scale is consistent with a complex or smooth classification at another scale, but is not consistent with a classification in any other direction at another scale. Due to the crudeness of the texture classification, we also consider neighboring directions as consistent with each other.



Fig. 3. Two-level texture map based on steerable filter decomposition. (a) Original grayscale image. (b) Texture classes at scale 0. (c) Texture classes at scale 1 (upsampled to original image size). (d) Combined texture classification.

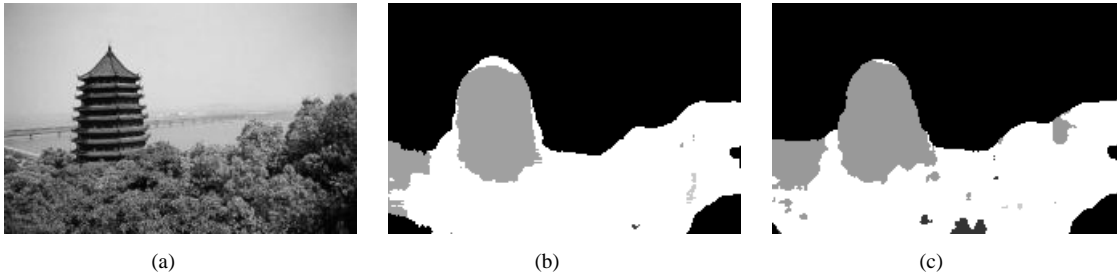


Fig. 4. Spatial texture feature extraction. (a) Original grayscale image. (b) Texture classes from steerable decomposition. (c) Texture classes from Gabor decomposition.

- Pixels that do not satisfy the above conditions are classified as complex.

Thus, the complex category includes pixels that are classified as complex at some scales and smooth at the remaining scales, or pixels that have inconsistent classification at different scales.

As we mentioned above, the use of multiple scales is particularly useful in capturing textures shown in different perspectives. Figure 3 shows the texture classifications that are obtained in each of two different scales, as well as the combined classification. Note that by combining texture information from two scales, the building is consolidated into one vertical texture region. Note that the horizontal lines are too far apart to be captured by either scale.

Finally, we should point out that the window size that is necessary for detecting a texture should increase as the scale becomes coarser, provided that no downsampling is performed. On the other hand, if the subband decomposition is critically sampled, as in the case of the discrete wavelet transform (DWT), the window should be fixed at all scales. The window used for computing the color composition features should be chosen accordingly.

4. PERCEPTUAL TUNING

We now discuss the selection of key parameters of the algorithm based on subjective experiments. But first we discuss the selection of the multiscale filter bank.

4.1. Filter Bank Selection

We have considered several multiscale frequency decompositions. The simplest is the DWT, which we used to obtain four texture classes, namely smooth, horizontal, vertical, and complex. Note that the DWT cannot distinguish between the two diagonal directions. We then considered more elaborate decompositions such as a steerable filter and a Gabor decomposition, which provide two more texture classes, *i.e.*, $+45^\circ$ and -45° .

We found that the proposed approach works effectively with any complete/overcomplete directional decomposition. We found that while the performance of the algorithm depends on the structure of the frequency decomposition (*e.g.*, the number of levels and orientations and their spacing), it is relatively independent of the detailed filter characteristics. As we saw in Section 2, neighboring orientation filters typically have significant overlap. Using sharper orientation filters does not make much of a difference, because it is the maximum of the filter responses that determines the texture orientation. Figure 4 shows the texture maps obtained by a one-level steerable pyramid decomposition in (b) and a one-level Gabor decomposition in (c), both with four orientations. The same procedure was used to obtain the two texture maps. The Gabor filters used were of size 9×9 pixels; the filter design and parameters were the same as those in [16]. In Fig. 4(b) and (c), black denotes smooth, white denotes complex, and light gray denotes horizontal textures. As expected, there are no major differences in the two results.

4.2. Subjective Experiments

Several key parameters of the segmentation algorithm can be determined by subjective tests. These include the threshold T_0 for the smooth/nonsmooth classification and the thresholds necessary for determining if there is a dominant orientation (T_1 and T_2). Another important parameter is the threshold for the color composition feature similarity.

The subjective experiments isolate small patches of images corresponding to homogeneous texture and color distributions. The texture patches are considered out of context, just as the algorithm does not make use of any context information. The parameter selections are based on a combination of texture statistics and how humans perceive textures. For more details on the subjective experiments,¹ we refer the reader to [10, 17]. Here, we focus on how the determination of these parameters is affected by scale.

¹Available online at <http://peacock.ece.utk.edu/FeatureTest/>.



Fig. 5. Image segmentation based on one-level texture classes.



Fig. 6. Image segmentation based on multiscale texture classes.

The experimental stimuli consisted of 37 uniform color texture segments of images from a photo CD. The textures were available at four or five scales. By using textures of different scales, there was no need to compute texture statistics in several scales. Thus, the thresholds obtained this way can be used across scales. We used a fixed window size that was reasonably big (e.g., 23×23 pixels), so that several texture scales can be perceived.

Note that, by displaying several texture scales, we can also find the minimum scale that can be perceived at that window size. Conversely, since the minimum window size at which a texture can be perceived is inversely proportional to the scale, this experiment can be used to determine the minimum window size. The determination of such a minimum window size is important for the performance of the segmentation algorithm. This is because, in order to obtain accurate border localization and adaptation to local texture characteristics, it is important to keep this parameter as small as possible. On the other hand, the window size should be big in order to obtain accurate estimates of the texture characteristics. Thus, it is necessary to select the smallest window size that captures the texture characteristics at a given scale.

5. SEGMENTATION RESULTS

Using the multiscale texture classification we developed in this paper, together with the perceptually tuned parameters [10], we can now obtain a perceptually tuned multiscale color-texture segmentation. A comparison of the segmentation results based on one-level texture classes and multiscale texture classification is shown in Figs. 5 and 6, respectively. In both cases, the texture window size was 23×23 . Note that in both cases, the multiscale approach results in improved segmentations.

6. REFERENCES

[1] Y. Rui, T.S. Huang, and S.-F. Chang, "Image retrieval: Current techniques, promising directions and open issues," *J. Visual Communication and Image Representation*, vol. 10, no. 1, pp. 39–62, Mar. 1999.

[2] W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Tr. Patt. Anal. Mach. Int.*, vol. 22, no. 12, pp. 1349–1379, Dec. 2000.

[3] T.P. Minka and R.W. Picard, "Interactive learning using a society of models," *Patt. Recognition*, vol. 30, no. 4, pp. 565–581, Apr. 1997.

[4] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Tr. Patt. Anal. Mach. Int.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[5] E.P. Simoncelli and J. Portilla, "Texture characterization via joint statistics of wavelet coefficient magnitudes," *Proc. ICIP-98, vol. I*, Chicago, IL, Oct. 1998, pp. 62–66.

[6] M. Mirmehdi and M. Petrou, "Segmentation of color textures," *IEEE Tr. Patt. Anal. Mach. Int.*, vol. 22, no. 2, pp. 142–159, Feb. 2000.

[7] J. Chen, T.N. Pappas, A. Mojsilovic, and B.E. Rogowitz, "Adaptive image segmentation based on color and texture," *Proc. ICIP-02*, Rochester, NY, Sept. 2002, vol. 2, pp. 789–792.

[8] J. Chen, T.N. Pappas, A. Mojsilovic, and B.E. Rogowitz, "Image segmentation by spatially adaptive color and texture features," in *Proc. ICIP-03*, Barcelona, Spain, Sept. 2003, vol. 1, pp. 1005–1008.

[9] J. Chen, T.N. Pappas, A. Mojsilovic, and B.E. Rogowitz, "Perceptual color and texture features for segmentation," in *Human Vision and Electronic Imaging VIII*, B. E. Rogowitz and T. N. Pappas, Eds., Santa Clara, CA, Jan. 2003, vol. Proc. SPIE Vol. 5007, pp. 340–351.

[10] J. Chen, T.N. Pappas, A. Mojsilovic, and B.E. Rogowitz, "Perceptual tuning of low-level color and texture features for image segmentation," *Asilomar Conf. Signals, Sys., and Comp.*, Pacific Grove, CA, Nov. 2003.

[11] T.N. Pappas, "An adaptive clustering algorithm for image segmentation," *IEEE Tr. Signal Proc.*, vol. SP-40, no. 4, pp. 901–914, Apr. 1992.

[12] A. Mojsilović, J. Hu, and E. Soljanin, "Extraction of perceptually important colors and similarity measurement for image matching, retrieval, and analysis," *IEEE Tr. Image Proc.*, vol. 11, no. 11, pp. 1238–1248, Nov. 2002.

[13] E.P. Simoncelli and W.T. Freeman, "The steerable pyramid: A flexible architecture for multi-scale derivative computation," in *Proc. ICIP-95, vol. III*, Washington, DC, Oct. 1995, pp. 444–447.

[14] J.G. Daugman and D.M. Kammen, "Pure orientation filtering: A scale invariant image-processing tool for perception research and data compression," *Behavior Research Methods, Instruments, and Computers*, vol. 18, no. 6, pp. 559–564, 1986.

[15] R.L. De Valois and K.K. De Valois, *Spatial Vision*, Oxford University Press, New York, 1990.

[16] B.S. Manjunath and W.Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Tr. Patt. Anal. Mach. Int.*, vol. 18, no. 8, pp. 837–842, Aug. 1996.

[17] J. Chen, *Perceptually-Based Color and Texture Features for Image Segmentation and Retrieval*, Ph.D. thesis, Northwestern Univ., Evanston, IL, Dec. 2003.