Unsupervised Texture Segmentation Using Dominant Image Modulations

T.B. Yap, T. Tangsukson, P.C. Tay, N.D. Mamuya, and J.P. Havlicek
School of Electrical & Computer Engineering
The University of Oklahoma, Norman, OK 73019-1023

Abstract

In this paper we present an unsupervised modulation domain technique for segmenting textured images. A dominant component AM-FM analysis is performed on the image, and estimates of the locally dominant amplitude and frequency modulations are extracted at each pixel. Modulation domain density clustering is then applied to estimate the maximum number of textured regions that might be present in the image. The feature space is augmented with horizontal and vertical spatial information prior to the application of k-means clustering to arrive at an initial image segmentation. Connected components labeling with minor region removal and morphological smoothing are then applied to yield the final segmentation. We demonstrate the technique on several synthetic and natural images.

1. Introduction

AM-FM image modeling is an important emerging area that represents images in terms of spatially localized amplitude and frequency modulations [8]. Computed dominant modulations have been used in a variety of applications including nonstationary analysis [2, 5, 7, 14], edge detection and image enhancement [15], recovery of 3D shape from texture [2, 19, 20], computational stereopsis [3], and texture segmentation and classification [1, 2, 6]. For texture segmentation, the typical approaches have based the segmentation on either Gabor filter amplitude and phase responses or the application edge detection algorithms to the computed dominant modulations.

In two recent papers we described a more general and robust technique that performs segmentation by clustering the dominant modulations in a modulation domain feature space [21, 22]. The primary shortcoming of this technique is that it is only *partially unsupervised*: the number of textured regions present in the image must be known *a priori*. In this paper, we present an enhanced algorithm that uses non-parametric density clustering similar to that described in [17] to determine the number of regions that are present in an image and to perform fully unsupervised texture segmentation. This approach has been developed very recently and we have not yet had an opportunity to compare the results quantitatively with those delivered by the overwhelmingly large number of segmentation techniques that have appeared in the literature. However, the experiments conducted to date indicate that our proposed approach consistently delivers correct pixel classification rates exceeding 93%, making it competitive with the best techniques that have been previously reported.

2. Dominant Component AM-FM Analysis

We model the image as a sum of nonstationary 2D AM-FM functions according to

$$t(\mathbf{x}) = \sum_{k=1}^{K} t_k(\mathbf{x}) = \sum_{k=1}^{K} a_k(\mathbf{x}) \exp[j\varphi_k(\mathbf{x})]. \quad (1)$$

Note that this model is complex-valued. Prior to analyzing a real-valued image $s(\mathbf{x})$, it is necessary to add an imaginary part $q(\mathbf{x})$ so that $t(\mathbf{x}) = s(\mathbf{x}) + jq(\mathbf{x})$. Writing $\mathbf{x} = [x\ y]^T$, we use an imaginary part given by the 2D directional Hilbert transform [8]

$$q(\mathbf{x}) = \text{p.v.}\frac{1}{\pi}\int_{\mathbb{R}} s(x-\xi, y)\,\frac{d\xi}{\xi}. \quad (2)$$

For each image component $t_k(\mathbf{x})$ in (1), the AM function $a_k(\mathbf{x})$ captures the local contrast. Similarly, the FM function $\nabla\varphi_k(\mathbf{x})$ captures local texture orientation and granularity.

Using the *dominant component analysis* technique (DCA) described in [8], we apply the complex-valued image $t(\mathbf{x})$ to a multiband Gabor filterbank. The response $y_i(\mathbf{x})$ of each filterbank channel is demodulated using the spatially localized nonlinear algorithms

$$\nabla\varphi_k(\mathbf{x}) \approx \text{Re}\left[\frac{\nabla y_i(\mathbf{x})}{j y_i(\mathbf{x})}\right], \quad (3)$$

$$a_k(\mathbf{x}) \approx \left|\frac{y_i(\mathbf{x})}{G_i[\nabla\varphi_k(\mathbf{x})]}\right| \quad (4)$$

0-7803-6514-3/00/$10.00©2000 IEEE

to estimate the unknown component AM and FM functions in (1), where $G_i(\cdot)$ is the frequency response of the i'th filterbank channel. The dominant image modulations $a_D(\mathbf{x})$ and $\nabla\varphi_D(\mathbf{x})$ are extracted on a pointwise basis from the channel that locally maximizes the selection criterion [8]

$$\Psi_i(\mathbf{x}) = \frac{|y_i(\mathbf{x})|}{\max_{\Omega}|G_i(\Omega)|}. \quad (5)$$

Use of this criterion is motivated by the fact that (5) tends to select as dominant those components that have a high-amplitude and also have frequency vectors lying near the maximum transmission frequency of the channel, thereby reducing demodulation errors due to noise and cross-component interference. As is well known, the dominant modulations provide a rich description of the local texture structure that is present in an image [2, 8, 14].

Although we expressed the demodulation algorithms (3) and (4) in terms of a continuous spatial variable $\mathbf{x}$ in the interest of economy of notation, we will refer to discrete image coordinates (m, n) for the remainder of the paper. To construct segmentation features from the dominant modulations obtained using (3)-(5), we first convert the cartesian frequency vector to polar coordinates: the magnitude frequency is given by $R(m,n) = |\nabla\varphi_D(m,n)|$ and the frequency orientation is given by $\theta(m,n) = \arg\nabla\varphi_D(m,n)$. We henceforth denote the discrete-domain dominant AM function by $A(m,n)$. The three features A, R, and θ are each divided by their respective sample standard deviations to obtain normalized features $\widetilde{A}$, $\widetilde{R}$, and $\widetilde{\theta}$.

3. Determining the Number of Regions

Undoubtedly, the most difficult aspect of performing unsupervised texture segmentation is determining the number of regions that are present in an image without *a prior* information [4, 9, 11–13, 16–18, 23] Indeed, this was the primary shortcoming of the algorithms described in [21, 22]. Similar to the technique described in [17], our approach is to estimate the local density of points in the feature space by applying a 3D Gaussian low-pass filter with kernel

$$K(m,n,p) = \left(2\pi\sigma^2\right)^{-\frac{3}{2}} e^{-\sqrt{m^2+n^2+p^2}/2\sigma^2}. \quad (6)$$

Application of this filter requires discretizing, or *binning*, the floating point feature values $\widetilde{A}$, $\widetilde{R}$, and $\widetilde{\theta}$. For an image of size 256×256, we typically divide each feature axis into 16 equal sized bins and use a value of $\sigma = 2$ for the space constant in (6). Using a larger number of bins can produce improved results, but only at the expense of considerably increased computation.

The 3D image that results from applying the filter (6) to the $\widetilde{A}$-$\widetilde{R}$-$\widetilde{\theta}$ feature space assumes large values where the density of points in feature space is high and small values where the points are sparse. We apply a gradient ascent algorithm to this filtered image to identify all local maxima and group the feature vectors into clusters about the detected maxima. The number of points in each resulting cluster is compared to a global threshold, where we use a threshold value of 500 points for images of size 256×256. Minor clusters are removed by merging them with the closest cluster that passes threshold, where closeness is defined with respect to the nearest neighbor rule. We denote the number of remaining clusters by M and calculate the corresponding M cluster centroids. Our algorithm then assumes that the number of regions present in the image is between one and M.

4. Segmentation by k-Means Clustering

The density-based clustering procedure described in the preceding section delivers an estimate M of the maximum number of regions present in the image and a set of M candidate cluster centers. Our approach for segmenting the image is to apply k-means clustering repeatedly for $1 \leq k \leq M$, where the centroids of the k largest clusters found by the density-based clustering procedure are used as initial cluster centers. The final cluster configuration is selected from among these by validating each result with respect to the usual squared-error criterion [10]. Details of the k-means algorithm will be given below.

For the cluster configuration selected by validation, the cluster labels are mapped back to the image domain and two post processing steps are applied to arrive at a final segmentation. First, connected components labeling with minor region removal is applied to enforce a spatial correspondence constraint and remove minor regions of misclassified pixels. Only the k largest connected components are retained, where k corresponds to the number of clusters in the k-means result selected by the validation procedure. Second, an isotropic morphological majority filter is applied to smooth the boundaries of the segmented regions. For images of size 256×256, we use a 9×9 circularly symmetric structuring element.

In performing k-means clustering, we augment the feature space with pixel position information from the image. Horizontal and vertical position features $\widetilde{X}$ and $\widetilde{Y}$ (normalized by sample standard deviation) are added to the $\widetilde{A}$-$\widetilde{R}$-$\widetilde{\theta}$ feature vector of every pixel to indicate the pixel's position within the image. The goal of this increase in feature space dimension from three to five is to encourage the formation of clusters that correspond to spatially connected regions in the image. Note that this requires adding two additional image position coordinates to each of the candidate cluster centers that were delivered by the density-based clustering procedure described in Section 3. These are calculated in

a straightforward manner by averaging the position features of all pixels belonging to each of the M clusters that were computed by the density-based approach.

We define the similarity between pixel (p,q) and pixel (m,n) by

$$s(p,q,m,n) = \Big\{\alpha\Big[\widetilde{A}(p,q) - \widetilde{A}(m,n)\Big]^2 + \beta\Big[\widetilde{R}(p,q) - \widetilde{R}(m,n)\Big]^2 + \gamma\Big[\widetilde{\theta}(p,q) - \widetilde{\theta}(m,n)\Big]^2 + \Big[\widetilde{X}(p,q) - \widetilde{X}(m,n)\Big]^2 + \Big[\widetilde{Y}(p,q) - \widetilde{Y}(m,n)\Big]^2\Big\}^{\frac{1}{2}}. \quad (7)$$

The weights α, β, and γ applied to the features $\widetilde{A}$, $\widetilde{R}$, and $\widetilde{\theta}$ in (7) are based on modulation domain entropy; as discussed in some detail in [21], they are designed to accentuate the features that provide the best class separability information. The basic line of reasoning is that features with histograms that exhibit several well defined modes that are clearly separated from one another tend to be powerful for discriminating between classes, whereas features with flat or unimodal histograms provide relatively little class separability information.

Here, we describe only calculation of the weight α corresponding to the feature $\widetilde{A}$. Similar calculations are used for the weights β and γ in (7). We denote the normalized histogram of $\widetilde{A}$ by $p_{\widetilde{A}}(q)$. The entropy of $\widetilde{A}$ is then given by

$$E_{\widetilde{A}} = -\sum_q p_{\widetilde{A}}(q)\log_2 p_{\widetilde{A}}(q). \quad (8)$$

We define the normalized entropy for $\widetilde{A}$ according to

$$\mathcal{E}_{\widetilde{A}} = \frac{E_{\widetilde{A}}}{\max_q p_{\widetilde{A}}(q) - \min_q p_{\widetilde{A}}(q)} \quad (9)$$

and the total entropy in the $\widetilde{A}$-$\widetilde{R}$-$\widetilde{\theta}$ feature subspace by

$$\mathcal{E}_T = \mathcal{E}_{\widetilde{A}} + \mathcal{E}_{\widetilde{R}} + \mathcal{E}_{\widetilde{\theta}}. \quad (10)$$

The weight α, which reflects the portion of the subspace entropy contributed by features $\widetilde{R}$ and $\widetilde{\theta}$, *i.e.*, the portion not contributed by $\widetilde{A}$, is given by

$$\alpha = (\mathcal{E}_T - \mathcal{E}_{\widetilde{A}})^2/\mathcal{E}_{\widetilde{A}}. \quad (11)$$

As a final note, we emphasize that, for each $k \in [1, M]$, *only one* run of the k-means algorithm is required. This is because we initialize the k-means algorithm with the centroids of the k largest clusters computed by the density clustering approach as opposed to initializing it with random initial cluster centers as is done frequently.

5. Examples

The images shown in Fig. 1(a), (e), (g), (i), and (k) are juxtapositions of several Brodatz or Brodatz-like textures. For the four-texture image of Fig. 1(a), the density-based clustering procedure estimated the maximum number of regions in the image as $M = 10$, whereas the total number of local density maxima prior to thresholding was 17. After k-means clustering with $1 \leq k \leq 10$, the squared-error validation criterion correctly selected the clustering result corresponding to $k = 4$. The raw segmentation obtained from the k-means algorithm for $k = 4$ is shown in Fig. 1(b). The result obtained by applying connected components labeling with minor region removal to the region map of Fig. 1(b) is shown in Fig. 1(c) and the final segmentation result after morphological majority filtering appears in Fig. 1(d). In this case, the correct pixel classification rate is 95.74%.

An image with five textured regions is shown in Fig. 1(e). In this case, total number of local maxima after density clustering was 26 and $M = 12$ of these clusters passed the size threshold. The k-means clustering algorithm was applied for $1 \leq k \leq 12$, and the validation criterion correctly selected the $k = 5$ result. The final segmentation after application of connected components labeling with minor region removal and morphological filtering appears in Fig. 1(f), where 95.16% of the pixels are correctly classified.

The image of Fig. 1(g) was obtained by rotating the central portion of a woodgrain image counterclockwise by 45°. There were 13 local maxima after density-based clustering, only $M = 6$ of which corresponded to clusters containing enough pixels to pass threshold. The final two-region segmentation after k-means clustering and post processing is given in Fig. 1(h). In this case, 98.23% of the pixels were correctly classified by the unsupervised algorithm described in this paper.

A three-texture image similar to those of Fig. 1(a) and (e) is shown in Fig. 1(i). There were 18 local maxima after density-based clustering. Half of these were eliminated by thresholding, resulting in an estimate of $M = 9$ for the maximum number of regions in the image. After k-means clustering, the validation criterion selected the $k = 3$ result. The corresponding final segmentation is shown in Fig. 1(j) where 95.04% of the pixels are correctly classified. Fig. 1(k) and (l) depict another two-texture example where 95.62% of the pixels were correctly classified by the proposed unsupervised approach.

Finally, two natural images from the MIT Media Lab *VisTex* database are shown in Fig. 1(m) and (o). The maximum number of regions estimated by the density-based clustering procedure was $M = 9$ in both cases, and the squared error validation criterion selected the $k = 2$ result delivered by the k-means algorithm in each case. The final segmentations obtained after connected components labeling with

minor region removal and morphological majority filtering are shown overlayed on the original images in Fig. 1(n) and (p), where pixel values in one region are divided by two with respect to their values in the original image.

6. Conclusion

We presented a fully unsupervised modulation domain technique for segmenting textured images. The key feature of the approach involves application of a Gaussian filter to estimate the density of points in a 3D modulation feature space. Candidate clusters are formed about local maxima in the filtered result and used to estimate the number of regions in the image without *a priori* information. The number and centroids of these clusters are subsequently used to initialize a k-means clustering algorithm in a 5D feature space that also incorporates pixel position information. Finally, the squared-error cluster validation criterion is used to select one of the k-means clustering results for use in computing the final segmentation. Connected components labeling with minor region removal and morphological majority filtering are applied as post processing steps to enforce spatial correspondence, remove small regions of misclassified pixels, and smooth region boundaries in the final segmentation.

The technique is promising in that it consistently obtains the correct number of image regions and delivers correct pixel classification rates exceeding 93%. Future work that remains to be done in assessing the algorithm's effectiveness includes exhaustive testing against a large number of synthetic and natural images and direct quantitative and qualitative comparison of the results with the best competing techniques that have been published.

References

[1] A. C. Bovik, M. Clark, and W. S. Geisler. Multichannel texture analysis using localized spatial filters. *IEEE Trans. Pattern Anal. Machine Intell.*, 12(1):55–73, January 1990.

[2] A. C. Bovik, N. Gopal, T. Emmoth, and A. Restrepo. Localized measurement of emergent image frequencies by Gabor wavelets. *IEEE Trans. Info. Theory*, 38(2):691–712, March 1992.

[3] T. Y. Chen, A. C. Bovik, and L. K. Cormack. Stereoscopic ranging by matching image modulations. *IEEE Trans. Image Proc.*, 8(6):785–797, June 1999.

[4] R. C. Dubes. How many clusters are best? an experiment. *Pattern Recognit.*, 20:645–663, 1987.

[5] B. Friedlander and J. M. Francos. An estimation algorithm for 2-D polynomial phase signals. *IEEE Trans. Image Proc.*, 5(6):1084–1087, June 1996.

[6] J. P. Havlicek. The evolution of modern texture processing. *Elektrik, Turkish Journal of Electrical Engineering and Computer Sciences*, 5(1):1–28, 1997.

[7] J. P. Havlicek, D. S. Harding, and A. C. Bovik. The multicomponent AM-FM image representation. *IEEE Trans. Image Proc.*, 5(6):1094–1100, June 1996.

[8] J. P. Havlicek, D. S. Harding, and A. C. Bovik. Multidimensional quasi-eigenfunction approximations and multicomponent AM-FM models. *IEEE Trans. Image Proc.*, 9(2):227–242, February 2000.

[9] T. Hoffmann, J. Puzicha, and J. M. Buhmann. Unsupervised texture segmentation in a deterministic annealing framework. *IEEE Trans. Pattern Anal. Machine Intell.*, 20(8):803–818, August 1998.

[10] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data.* Prentice Hall, Englewood Cliffs, NJ, 1988.

[11] C. Kervrann and F. Heitz. A Markov random field model-based approach to unsupervised texture segmentation using local and global spatial statistics. *IEEE Trans. Image Proc.*, 4(6):856–862, June 1995.

[12] D. A. Langan, J. W. Modestino, and J. Zhang. Cluster validation for unsupervised stochastic model-based image segmentation. *IEEE Trans. Image Proc.*, 7(2):180–195, February 1998.

[13] B. S. Manjunath and R. Chellappa. Unsupervised texture segmentation using Markov radom field models. *IEEE Trans. Pattern Anal. Machine Intell.*, 13(5):478–482, May 1991.

[14] P. Maragos and A. C. Bovik. Image demodulation using multidimensional energy separation. *J. Opt. Soc. Amer. A*, 12(9):1867–1876, September 1995.

[15] S. K. Mitra, S. Thurnhofer, M. Lightstone, and N. Strobel. Two-dimensional Teager operators and their image processing applications. In *Proc. 1995 IEEE Workshop Nonlin. Signal and Image Proc.*, pages 959–962, Neos Marmaras, Halkidiki, Greece, June 20-22, 1995.

[16] H. H. Nguyen and P. Cohen. Gibbs random fields, fuzzy clustering, and the unsupervised segmentation of textured images. *CVGIP: Graph. Models and Image Proc.*, 55(1):1–19, January 1993.

[17] E. J. Pauwels and G. Frederix. Finding salient regions in images. *Comput. Vision, Image Understand.*, 75(1/2):73–85, July/August 1999.

[18] O. Pichler, A. Teuner, and B. J. Hosticka. An unsupervised texture segmentation algorithm with feature space reduction and knowledge feedback. *IEEE Trans. Image Proc.*, 7(1):53–61, January 1998.

[19] B. J. Super and A. C. Bovik. Planar surface orientation from texture spatial frequencies. *Pattern Recog.*, 28(5):728–743, 1995.

[20] B. J. Super and A. C. Bovik. Shape from texture using local spectral moments. *IEEE Trans. Pattern Anal. Machine Intell.*, 17(4):333–343, 1995.

[21] T. Tangsukson and J. P. Havlicek. AM-FM image segmentation. In *Proc. IEEE Int'l. Conf. Image Proc.*, Vancouver, Canada, September 10-13 2000.

[22] T. Tangsukson and J. P. Havlicek. Modulation domain image segmentation. In *Proc. IEEE Southwest Symp. Image Anal., Interp.*, pages 46–50, Austin, TX, April 2-4 2000.

[23] C. S. Won and H. Derin. Unsupervised segmentation of noisy and textured images using Markov random fields. *CVGIP: Graph. Models and Image Proc.*, 54(4):308–328, July 1992.

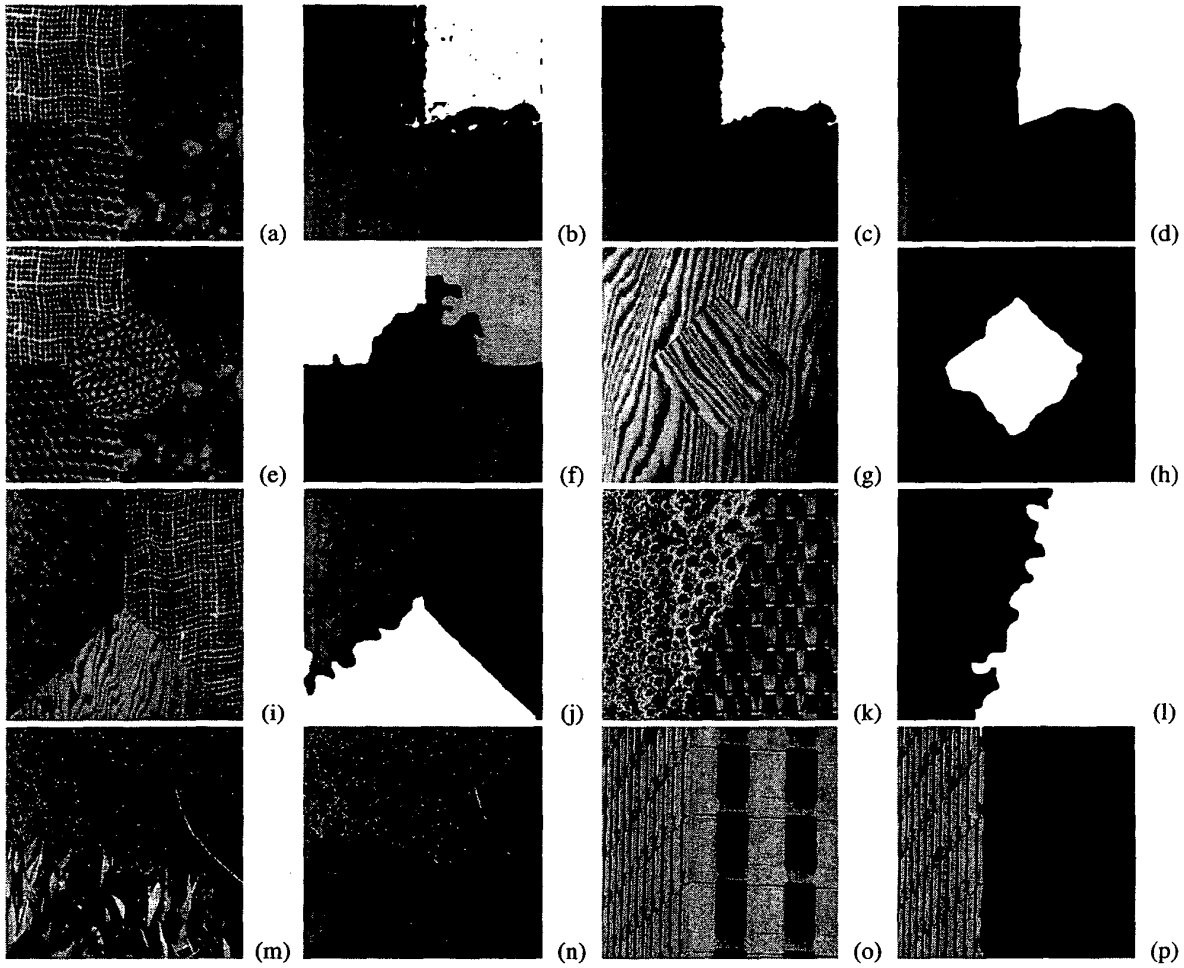

Figure 1. (a) Four-texture original image. (b) Result of k-means clustering for $k = 4$. (c) Result after connected components labeling. (d) Final segmentation obtained after application of morphological majority filtering. The correct pixel classification rate is 95.74%. (e) Five-texture original image. (f) Final segmentation result delivered by the proposed unsupervised algorithm. Correct pixel classification rate is 95.16%. (g) Original *WoodWood* image. (h) Final segmentation result. Correct pixel classification rate is 98.23%. (i) Three-texture original image. (j) Final segmentation result, 95.04% of the pixels are correctly classified. (k) Two-texture original image. (l) Final segmentation result with correct pixel classification rate of 95.62%. (m) Original *GrassPlantsSky.0005* image. (n) Overlay of final segmentation result on the original image. Pixel values in the lower region are divided by two with respect to the original image, whereas pixel values in the upper region are unaltered relative to the original image. (o) Original *Building0008* image. (p) Overlay of final segmentation on original image. Pixel values in the right-hand region are divided by two relative to the original image.