

Multiband Techniques for Texture Classification and Segmentation

B.S. Manjunath
University of California

G.M. Haley
Ameritech, Hoffmann Estates

Wei-Ying Ma
Microsoft Research Asia

S.D. Newsam
Lawrence Livermore National Laboratory

1	Introduction.....	455
1.1	Image Texture • 1.2 Gabor Features for Texture Classification and Image Segmentation • 1.3 Chapter Organization	
2	Gabor Functions.....	457
2.1	One-Dimensional Gabor Function • 2.2 An Analytic Gabor Function •	
2.3	2.3 Two-Dimensional Gabor Function — Cartesian Form • 2.4 Two-Dimensional Gabor Function — Polar Form • 2.5 Multiresolution Representation with Gabor Wavelets	
3	Microfeature Representation.....	459
3.1	Transformation into Gabor Space • 3.2 Local Frequency Estimation •	
3.3	3.3 Transformation into Microfeatures	
4	The Texture Model.....	461
4.1	4.1 The Texture Micromodel • 4.2 Macrofeatures • 4.3 The Texture Macromodel	
5	Experimental Results.....	462
5.1	5.1 Classification	
6	Image Segmentation Using Texture.....	463
7	Image Retrieval Using Texture.....	464
8	Texture Motifs.....	465
9	Summary.....	468
	Acknowledgments.....	469
	References.....	469

1 Introduction

1.1 Image Texture

Texture as an image feature is very useful in many image processing and computer vision applications. There is an extensive literature on texture analysis in the image processing literature where the primary focus has been on classification, segmentation and synthesis. Texture features have been used in diverse applications such as satellite and aerial image analysis, medical image analysis for detection of abnormalities, and more recently, in image retrieval using texture as a descriptor. In this chapter, we present an approach to characterizing texture using a multiband decomposition of

the image with application to classification, segmentation, object detection, and image retrieval.

In texture classification and segmentation, the objective is to partition the given image into a set of homogeneous textured regions. Aerial images are excellent examples of textured regions where different areas such as water, sand, vegetation, and so forth have distinct texture signatures. In many other cases, such as in the classification of tissues in the magnetic resonance images of the brain, homogeneity is not that well defined. If an image consists of multiple textured regions, as is the case with most natural imagery, segmentation can be achieved through classification. This, however, is a chicken-and-egg problem as classification requires an estimate of the region boundaries — note that

texture is a region property and individual pixels are labeled on the basis of information in a small neighborhood around the pixels. This may lead to problems near region boundaries as the computed texture descriptors are corrupted from pixels not belonging to the same region.

Early work on texture classification focussed on spatial image statistics. These include image correlation [10], energy features [28], features from co-occurrence matrices [23] and run-length statistics [20]. During the last 15 years, much attention has been given to generative models such as those using the Markov random fields (MRFs) [8, 9, 12, 13, 16, 17, 24–27, 34] (see also Chapter 4.3 on MRF models). MRF-based methods have proven to be quite effective for texture synthesis, classification, and segmentation. Because general MRF models are inherently dependent on rotation, several methods have been introduced to obtain rotation invariance. Kashyap and Khotanzad [25] developed the “circular autoregressive” model with parameters that are invariant to image rotation. Choe and Kashyap [11] introduced an autoregressive fractional difference model that has rotation (as well as tilt and slant) invariant parameters. Cohen, Fan, and Patel [12] extended a likelihood function to incorporate rotation (and scale) parameters. To classify a sample, an estimate of its rotation (and scale) is required.

Much of the work in MRF models uses the image intensity as the primary feature. In contrast, spatial filtering methods derive the texture descriptors using the filtered coefficient values. A compact representation of the filtered outputs is needed for classification and/or segmentation purposes. The first few moments of the filtered images are often used as feature vectors. For segmentation, one may consider abrupt transitions in the filtered image space or transformations of the filtered images. Malik and Perona [32], for example, argue that a nonlinear transformation of the filtered coefficients is necessary to model preattentive segmentation by humans.

Laws [28] was perhaps among the first to propose the use of energy features for texture classification. In recent years, multi-scale decompositions of the images have been extensively used in deriving image texture descriptors and segmentation [4, 5, 7, 18, 21, 22, 29, 35, 36, 41–43, 45]. Orthogonal wavelets (see Chapter 4.2) and Gabor wavelets have been widely used for computing such multiscale decompositions. Gabor functions, which are modulated Gaussians, are described in Section 2.

For feature-based approaches, rotation-invariance is achieved by using anisotropic features. Porat and Zeevi [43] use first- and second-order statistics based on three spatially localized features, two of which (dominant spatial frequency and orientation of dominant spatial frequency) are derived from a Gabor-filtered image. Leung and Peterson [29] present two approaches, one that transforms a Gabor-filtered image into rotation-invariant features and the other that rotates the image before filtering; however, neither uses the spatial resolving capabilities of the Gabor filter. You and Cohen [46] use filters that are tuned over a training set to provide

high discrimination among its constituent textures. Greenspan et al. [21] use rotation-invariant structural features obtained via multiresolution Gabor filtering. Rotation invariance is achieved by using the magnitude of a discrete Fourier transform (DFT) in the rotation dimension.

Many researchers have used the Brodatz album [6] for evaluating the performance of their texture classification and segmentation schemes. However, there is such a large variance in the actual subsets of images used and in the performance evaluation methodology that it is practically impossible to compare the evaluations presented in various papers. For example, Porter and Canagarajah [44] discuss several schemes for rotation-invariant classification using wavelets, Gabor filters, and Gaussian Markov random field (GMRF) models. They conclude, on the basis of experiments on 16 images from the Brodatz set, that the wavelet features provide better classification performance compared with the other two texture features. A similar study by Manian and Vasquez [33] also concluded that orthogonal wavelet features provide better invariant descriptors. A study by Pichler et al. [40], from an image segmentation point of view, concludes that Gabor features provide better segmentation results compared with orthogonal wavelet features. Perhaps the most comprehensive study on evaluating different texture descriptors is provided by Manjunath and Ma [36], in the context of image retrieval. They use the entire Brodatz texture set and compare features derived from wavelet decomposition, tree-structured decomposition, Gabor wavelets, and multiresolution simultaneous autoregressive (MRSAR) models. They conclude that Gabor features and MRSAR model features outperform features from orthogonal or tree structured wavelet decomposition. More recently, the study presented by Haley and Manjunath [22] indicated that the rotation-invariant features from Gabor filtering compare favorably with GMRF-based schemes. They also provide results on the entire Brodatz data set.

1.2 Gabor Features for Texture Classification and Image Segmentation

The following sections describe this rotation-invariant texture feature set. For detailed experimental results, we refer the reader to Haley and Manjunath [22]. The texture feature set is derived by filtering the image through a bank of modified Gabor kernels. The particular set of filters forms a multi-resolution decomposition of the image. Although there are several viable options, including orthogonal wavelet transforms, Gabor wavelets were chosen for their desirable properties:

- Gabor functions achieve the theoretical minimum space-frequency bandwidth product [14, 15, 19] (i.e., spatial resolution is maximized for a given bandwidth)
- A narrow-band Gabor function closely approximates an analytic (frequency causal) function (see also Chapter 4.4 for a discussion on analytic signals). Signals convolved

with an analytic function are also analytic, allowing separate analysis of the magnitude (envelope) and phase characteristics in the spatial domain.

- The magnitude response of a Gabor function in the frequency domain is well behaved, having no sidelobes
- Gabor functions appear to share many properties with the human visual system [38]

While Gabor functions are a good choice, the standard forms can be further improved. Under certain conditions, very-low-frequency effects (e.g., due to illumination and shading variations) can cause a significant response in a Gabor filter, leading to misclassification. An analytic form is introduced (see Section 2.2) to minimize these undesirable effects. When the center frequencies are evenly spaced on concentric circles, the polar form of the two-dimensional (2D) Gabor function allows superior frequency domain coverage, improves rotation-invariance, and simplifies analysis, compared with the standard 2D form.

1.3 Chapter Organization

This chapter is organized as follows: In Section 2 we introduce an analytic Gabor function and a polar representation for the 2D Gabor filters. A multiresolution representation of the image samples using Gabor functions is presented. In Section 3, the Gabor space samples are then transformed into a *microfeature* space where a rotation independent feature set is identified. Section 4 describes a texture model based on *macrofeatures* that are computed from the texture *micro-features*. These macrofeatures provide a global description of the image sample and are useful for classification and segmentation. Section 5 gives experimental results on rotation-invariant texture classification. Section 6 outlines a segmentation scheme, called *EdgeFlow* [30] which uses the texture energy features to partition the image. Section 7 gives an application of using texture descriptors to image retrieval [31, 36]. Some retrieval examples in the context of aerial imagery are shown. Finally, Section 8 describes a technique for characterizing textures common to a class of objects [3, 39] and demonstrates how it can be used to assist in object recognition.

2 Gabor Functions

2.1 One-Dimensional Gabor Function

A Gabor function is the product of a Gaussian function and a complex sinusoid. Its general one-dimensional (1D) form is

$$g_S(x, \omega_C, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \cdot \exp\left(\frac{-x^2}{2\sigma^2}\right) \cdot \exp(j\omega_C x) \quad (1)$$

$$G_S(\omega, \omega_C, \sigma) = \exp\left[\frac{-\sigma^2(\omega - \omega_C)^2}{2}\right] \quad (2)$$

Thus, Gabor functions are bandpass filters. Gabor functions are used as complete, albeit nonorthogonal, basis sets. It has been shown that a function $i(x)$ is represented exactly [19] as

$$i(x) = \sum_{n=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} \beta_{n,k} \cdot h_{n,k}(x) \quad (3)$$

where $h_{n,k}(x) = g_S(x - nX, k\Omega, \sigma)$, and $X\Omega = 2\pi$.

2.2 An Analytic Gabor Function

$G_S(\omega, \omega_C, \sigma)$ exhibits a potentially significant response at $\omega = 0$ and at very low frequencies. The response to a constant-valued input (i.e., $\omega = 0$) relative to the response to an input of equal magnitude at $\omega = \omega_C$ can be computed as a function of octave bandwidth [4]:

$$|G_S(0)| / |G_S(\omega_C)| = 2^{-\gamma} \quad (4)$$

where $\gamma = (2^B + 1)/(2^B - 1)$ and $B = \log_2((\omega_C + \delta)/(\omega_C - \delta))$ and δ is the half bandwidth. It is interesting to note that the response at $\omega = 0$ depends upon the B but not ω_C . This behavior manifests itself as an undesirable response to interimage and intraimage variations in contrast and intensity due to factors unrelated to the texture itself, potentially causing misclassification. Cases include

- Sample images of a texture with differences in average intensity
- Images with texture regions having differences in contrast and/or intensity (Bovik [4] has demonstrated that region boundaries defined in segmentation using unmodified Gabor filters vary according to these differences between the regions)
- Images with uneven illumination

There are two approaches to avoiding these problems: preprocessing the image or modifying the Gabor function. Normalizing each image to have a standard average intensity and contrast corrects for interimage, but not intraimage, variations. Alternative methods of image preprocessing are required to compensate for intraimage variations, such as point logarithmic processing [4] or local normalization.

An equally effective and more straightforward approach is to modify the Gabor function to be analytic¹ (see also Chapter 4.4 on analytic signals) by forcing the real and imaginary parts to become a Hilbert transform pair. This is accomplished by replacing the real part of $g_S(x)$, $g_{S\text{Re}}(x)$, with the inverse Hilbert transform of the imaginary part, $-\hat{g}_{S\text{Im}}(x)$:

$$g_A(x) = -\hat{g}_{S\text{Im}}(x) + jg_{S\text{Im}}(x). \quad (5)$$

¹Since $G_S(\omega) \neq 0$ for $\omega \neq 0$, a Gabor function only approximates an analytic function.

The Fourier transforms of the real and imaginary parts of $g_S(x)$ are respectively conjugate symmetric and conjugate antisymmetric, resulting in cancellation for $\omega \leq 0$:

$$G_A(\omega) = \begin{cases} G_S(\omega) - G_S^*(-\omega), & \omega > 0 \\ 0, & \omega \leq 0 \end{cases} \quad (6)$$

Because it is analytic, $G_A(\omega)$ possesses several advantages over for $G_S(\omega)$ for many applications including texture analysis:

- Improved low-frequency response because $|G_A(\omega)| < |G_S(\omega)|$ for small ω and $|G_A(0)| = 0$;
- Simplified frequency domain analysis since $G_A(\omega) = 0$ for $\omega \leq 0$;
- Reduced frequency domain computations because $G_A(\omega) = 0$ for $\omega \leq 0$.

These advantages are achieved without requiring additional processing. Thus, it is an attractive alternative for most texture analysis applications.

2.3 Two-Dimensional Gabor Function — Cartesian Form

The Gabor function is extended into two dimensions as follows. In the spatial-frequency domain, the Cartesian form is a 2D Gaussian formed as the product of two 1D Gaussians from (2):

$$\begin{aligned} G_C(\omega_x, \omega_y, \omega_{Cx}, \omega_{Cy}, \theta, \sigma_x, \sigma_y) \\ = G(\omega_x, \omega_{Cx}, \sigma_x) \cdot G(\omega_y, \omega_{Cy}, \sigma_y) \end{aligned} \quad (7)$$

where θ is the orientation angle of G_C , $x' = x \cos \theta + y \sin \theta$ and $y' = -x \sin \theta + y \cos \theta$. In the spatial domain, G_C is separable into two orthogonal 1D Gabor functions from (1) that are respectively aligned to the x' and y' axes:

$$g_C(x, y, \omega_{Cx}, \omega_{Cy}, \theta, \sigma_x, \sigma_y) = g(x', \omega_{Cx}, \sigma_x) \cdot g(y', \omega_{Cy}, \sigma_y) \quad (8)$$

As in (3), an image is represented exactly [1, 2]² as

$$i(x, y) = \sum_{n_x=-\infty}^{\infty} \sum_{n_y=-\infty}^{\infty} \sum_{k_x=-\infty}^{\infty} \sum_{k_y=-\infty}^{\infty} \beta_{k_x, k_y, n_x, n_y} \cdot h_{k_x, k_y, n_x, n_y}(x, y) \quad (9)$$

where $h_{k_x, k_y, n_x, n_y}(x, y) = g(x - n_x X, k_x \Omega_x, \sigma_x) \cdot g(y - n_y Y, k_y \Omega_y, \sigma_y)$; σ_x , σ_y , X , Y , Ω_x and Ω_y are constants; and $X\Omega_x = Y\Omega_y = 2\pi$. Approximations to $\beta_{k_x, k_y, n_x, n_y}$ are obtained by using [38]

$$\hat{\beta}_{k_x, k_y, n_x, n_y} = i(x, y) * h_{k_x, k_y, n_x, n_y}(x, y) \approx \beta_{k_x, k_y, n_x, n_y}, \quad (10)$$

²The proofs in the references are based on the standard, not analytic, form of the Gabor function.

provided that the parameters are chosen appropriately.

2.4 Two-Dimensional Gabor Function — Polar Form

An alternative approach to extending the Gabor function into two dimensions is to form, in the frequency domain, the product of a 1D analytic Gabor function $G(\omega)$ (the subscript is omitted to indicate that the concepts are generally applicable to the standard form as well) of radial frequency ω and a Gaussian function of orientation θ :

$$G_P(\omega, \theta, \omega_C, \theta_C, \sigma_\rho, \sigma_\theta) = G(\omega, \omega_C, \sigma_\rho) \cdot \exp\left[\frac{-\sigma_\theta^2(\theta - \theta_C)^2}{2}\right] \quad (11)$$

$$\begin{aligned} g_P(x, y, \omega_C, \theta_C, \sigma_\rho, \sigma_\theta) \\ = \iint G_P\left(\sqrt{\omega_x^2 + \omega_y^2}, \tan^{-1}(\omega_y/\omega_x), \omega_C, \theta_C, \sigma_\rho, \sigma_\theta\right) \cdot \\ \times \exp[j2\pi(\omega_x x + \omega_y y)] d\omega_x d\omega_y \end{aligned} \quad (12)$$

where $\omega = \sqrt{\omega_x^2 + \omega_y^2}$ and $\tan(\theta) = \omega_y/\omega_x$. Thus, (11) is a 2D Gaussian in the polar, rather than Cartesian, spatial-frequency domain. The frequency domain regions of both polar and Cartesian forms of Gabor functions are compared in Fig. 1.

In the Cartesian spatial-frequency domain, the -3 dB contour of the Cartesian form is an ellipse, whereas the polar form has a narrower response at low ω and a wider response at high

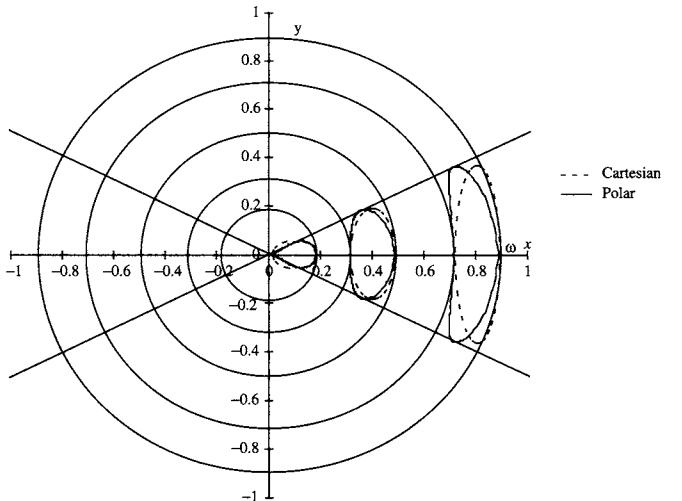


FIGURE 1 Contours (-3 dB) of Cartesian and polar Gabor functions of varying bandwidths. The angular -3 -dB width of the polar Gabor functions is 45 degrees.

ω . When arranged as “flower petals” (equally distributed along a circle centered at the origin), the polar form allows more uniform coverage of the frequency domain, with less overlap at low frequencies and smaller gaps at high frequencies. The polar form is more suited for rotation-invariant analysis because the response always varies as a Gaussian with rotation. The Cartesian form varies with rotation in a more complex manner, introducing an obstacle to rotation invariance and complicating analysis.

2.5 Multiresolution Representation with Gabor Wavelets

The Gabor function is used as the basis for generating a wavelet family for multiresolution analysis (see Chapter 4.2 on wavelets). Wavelets have two salient properties: the octave bandwidth B and the octave spacing $\Delta = \log_2(\omega_{s+1}/\omega_s)$ are both constant, where ω_s is the center frequency. The filter spacing is achieved by defining

$$\omega_s = \omega_0 \cdot 2^{-s\Delta}, \quad s \in \{0, 1, 2, \dots\} \quad (13)$$

where ω_0 is the highest frequency in the wavelet family. Constant bandwidth requires that σ_ρ be inversely proportional to ω_s :

$$\sigma_{\rho_s} = \frac{1}{\kappa \omega_s} \quad (14)$$

where

$$\kappa = \frac{2B - 1}{\sqrt{2 \ln 2} \cdot (2B + 1)}$$

is a constant. The orientations of the wavelets are defined as

$$\theta_r = \theta_0 + \frac{2\pi r}{R}, \quad (15)$$

where θ_0 is the starting angle, the second term is the angular increment, and r and R are both integers such that $0 \leq r < R$. Using (13), (14), and (15) in (11), the 2D Gabor wavelet family is defined as

$$\begin{aligned} H_{s,r}(\omega_x, \omega_y) &= G_P \left(\sqrt{\omega_x^2 + \omega_y^2}, \tan^{-1}(\omega_y/\omega_x), \omega_s, \theta_r, \frac{1}{\kappa \omega_s}, \sigma_\theta \right) \\ &= G \left(\sqrt{\omega_x^2 + \omega_y^2}, \omega_s, \frac{1}{\kappa \omega_s} \right) \cdot \exp \left[\frac{-\sigma_\theta^2 (\tan^{-1}(\omega_y/\omega_x) - \theta_r)^2}{2} \right] \end{aligned}$$

$$h_{s,r,n_x,n_y}(x, y) = g_P \left(x - n_x X_s, y - n_y Y_s, \omega_s, \theta_r, \frac{1}{\kappa \omega_s}, \sigma_\theta \right), \quad (17)$$

where X_s and Y_s , the sampling intervals, are inversely proportional to the bandwidths corresponding to s . As in (9), an image is represented using the polar wavelet form of the Gabor function from (17):

$$i(x, y) = \sum_{n_x=-\infty}^{\infty} \sum_{n_y=-\infty}^{\infty} \sum_{s=0}^{\infty} \sum_{r=0}^{R-1} \beta_{s,r,n_x,n_y} \cdot h_{s,r,n_x,n_y}(x, y). \quad (18)$$

Approximations to β_{s,r,n_x,n_y} are obtained as in (10):

$$\hat{\beta}_{s,r,n_x,n_y} = i(x, y) * h_{s,r,n_x,n_y}(x, y) \approx \beta_{s,r,n_x,n_y}, \quad (19)$$

and parameters X_s , Y_s , ω_0 , κ and σ_θ are chosen appropriately. Instead of a rectangular lattice, a polar Gabor wavelet representation has the shape of a cone.

3 Microfeature Representation

3.1 Transformation into Gabor Space

As described in Section 2, a set of 2D Gabor wavelets can represent an image. Assuming that the image is spatially limited to $0 \leq x < N_x X_s$, $0 \leq y < N_y Y_s$, where N_x and N_y represent the number of samples in their respective dimensions, and is bandlimited to $0 < \omega \leq \omega_H$,³ the number of Gabor wavelets needed to represent the image is finite. Substituting $\hat{\beta}_{s,r,n_x,n_y}$ from (19) for β_{s,r,n_x,n_y} in (18), a texture image is approximately represented using the polar wavelet form of the Gabor function as

$$i(x, y) \approx \sum_{n_x=0}^{N_x-1} \sum_{n_y=0}^{N_y-1} \sum_{s=0}^{S-1} \sum_{r=0}^{R-1} \hat{\beta}_{s,r,n_x,n_y} \cdot h_{s,r,n_x,n_y}(x, y), \quad (20)$$

where parameters S , R , X_s , Y_s , ω_0 , κ and σ_θ are chosen appropriately. Note that the s subscript is added to N_x and N_y to indicate their dependencies on X_s and Y_s . Thus, a texture image is represented with relatively little information loss by the coefficients $\hat{\beta}_{s,r,n_x,n_y}$.

Following Bovik et al. [5], $\hat{\beta}_{s,r,n_x,n_y}$ is interpreted as a channel or band $b_{s,r}(n_x, n_y)$ of the image $i(x, y)$ tuned to the carrier frequency $\omega_s = \omega_0 \cdot 2^{-s\Delta}$ (13), oriented at angle $\theta_r = \theta_0 + 2\pi r/R$ (15) and sampled in the spatial domain at intervals of X_s and Y_s . Since $b_{s,r}(n_x, n_y)$ is formed by convolution with a narrowband, analytic function (19), $b_{s,r}(n_x, n_y)$ is also narrowband and analytic and is therefore

³For sampled texture images, the upper frequency bound is enforced, although aliasing may be present since natural textures are generally not bandlimited. It is both reasonable and convenient to assume that, for textures of interest, a lower frequency bound $\tau \omega_L$ ($\tau > 0$) exists below which there is no useful discriminatory information.

decomposable into amplitude and phase components that can be independently analyzed:

$$b_{s,r}(n_x, n_y) = a_{s,r}(n_x, n_y) \cdot \exp(j\psi_{s,r}(n_x, n_y)), \quad (21)$$

where $a_{s,r}(n_x, n_y) = |b_{s,r}(n_x, n_y)|$ and $\psi_{s,r}(n_x, n_y) = \arg(b_{s,r}(n_x, n_y))$. $a_{s,r}(n_x, n_y)$ contains information about the amplitude and amplitude modulation (AM) characteristics of the texture's periodic features within the band, and $\psi_{s,r}(n_x, n_y)$ contains information about the phase, frequency and frequency modulation (FM) characteristics (see Chapter 4.4 for a discussion on AM/FM signals). For textures with low AM in band (s, r) , $a_{s,r}(n_x, n_y)$ is approximately constant over (n_x, n_y) . For textures with low FM in band (s, r) , the slope of $\psi_{s,r}(n_x, n_y)$ with respect to (n_x, n_y) is nearly constant.

Both $a_{s,r}(n_x, n_y)$ and $\psi_{s,r}(n_x, n_y)$ are rotation-dependent and periodic in r such that

$$a_{s,((r+R/2))_R}(n_x, n_y) = a_{s,r}(n_x, n_y) \quad (22)$$

$$\psi_{s,((r+R))_R}(n_x, n_y) = \psi_{s,r}(n_x, n_y) \quad (23)$$

$$\psi_{s,((r+R/2))_R}(n_x, n_y) = -\psi_{s,r}(n_x, n_y) \quad (24)$$

Rotating $i(x, y)$ by θ degrees produces a circular shift in r of $-R\theta/180$ degrees for $a_{s,r}(n_x, n_y)$ and $-R\theta/360$ degrees for $\psi_{s,r}(n_x, n_y)$.

3.2 Local Frequency Estimation

While $\psi_{s,r}(n_x, n_y)$ contains essential information about a texture, it is not directly usable for classification. However, local frequency information can be extracted from $\psi_{s,r}(n_x, n_y)$ as follows:

$$\phi_{s,r}(n_x, n_y) = \begin{cases} \theta_\nabla, & |\theta_r - \theta_\nabla| \leq 90^\circ \\ \theta_\nabla + 180^\circ, & |\theta_r - \theta_\nabla| > 90^\circ \end{cases}, \text{ and} \quad (25)$$

$$\begin{aligned} u_{s,r}(n_x, n_y) &= \sqrt{\nabla_x^2(\psi_{s,r}(n_x, n_y)) + \nabla_y^2(\psi_{s,r}(n_x, n_y))} \cdot \\ &\quad \times \cos(\theta_r - \phi_{s,r}(n_x, n_y)), \\ &= \sqrt{\nabla_x^2(\psi_{s,r}(n_x, n_y)) + \nabla_y^2(\psi_{s,r}(n_x, n_y))} \cdot \\ &\quad \times |\cos(\theta_r - \theta_\nabla)| \end{aligned} \quad (26)$$

where $\nabla_x()$ and $\nabla_y()$ are gradient estimation functions, θ_r is the orientation of the Gabor function, and $\theta_\nabla = \tan^{-1}(\nabla_y(\psi_{s,r}(n_x, n_y))/\nabla_x(\psi_{s,r}(n_x, n_y)))$ is the direction of the

gradient vector. $u_{s,r}(n_x, n_y)$ is a spatially localized estimate of the frequency along the direction θ_r , and $\phi_{s,r}(n_x, n_y)$ is the direction of maximal phase change rate, i.e., highest local frequency.

3.3 Transformation into Microfeatures

To facilitate discrimination between textures, $b_{s,r}(n_x, n_y)$ is further decomposed into *microfeatures* that contain local amplitude, frequency, phase, direction and directionality characteristics. In the following, for simplicity, R is assumed to be even. The microfeatures are defined to be:

$$f_{As,p}(n_x, n_y) = \sum_{r=0}^{R/2-1} a_{s,r}(n_x, n_y) \cdot a_{s,((r+p))_R}(n_x, n_y), \quad 0 \leq p \leq R/4; \quad (27)$$

$$f_{Ps,q}(n_x, n_y) = \left| \sum_{r=0}^{R/2-1} u_{s,r}(n_x, n_y) \cdot \exp\left(-\frac{2\pi jrq}{R/2}\right) \right|, \quad 0 \leq q \leq R/4; \quad (28)$$

$$\begin{aligned} f_{Ys,q}(n_x, n_y) &= \left| \sum_{r=0}^{R-1} \exp(j\phi_{s,r}(n_x, n_y)) \cdot \exp\left(-\frac{2\pi jrq}{R}\right) \right|, \\ &\quad q = 1, 3, \dots, R-1; \end{aligned} \quad (29)$$

$$\begin{aligned} f_{DAs,q}(n_x, n_y) &= \arg \left[\sum_{r=0}^{R/2-1} a_{s,r}(n_x, n_y) \cdot \exp\left(-\frac{2\pi jrq}{R/2}\right) \right], \\ &\quad 1 \leq q \leq R/4; \end{aligned} \quad (30)$$

$$\begin{aligned} f_{DFs,q}(n_x, n_y) &= \arg \left[\sum_{r=0}^{R/2-1} u_{s,r}(n_x, n_y) \cdot \exp\left(-\frac{2\pi jrq}{R/2}\right) \right], \\ &\quad 1 \leq q \leq R/4; \end{aligned} \quad (31)$$

$$\begin{aligned} f_{DYs,q}(n_x, n_y) &= \arg \left[\sum_{r=0}^{R-1} \exp(j\phi_{s,r}(n_x, n_y)) \cdot \exp\left(-\frac{2\pi jrq}{R}\right) \right], \\ &\quad q = 1, 3, \dots, R-1. \end{aligned} \quad (32)$$

$f_{As,p}(n_x, n_y)$ contains the amplitude envelope information from $b_{s,r}(n_x, n_y)$. Because of the $R/2$ periodicity of $a_{s,r}$ (22), only $R/2$ components are needed in the sum in (27). Eliminating the redundant components from the circular autocorrelation allows complete representation by the $0 \leq p \leq R/4$ components of $f_{As,p}(n_x, n_y)$. It is rotation-invariant because the

autocorrelation operation eliminates the dependence on r , and thus, on θ .

$f_{Fs,q}(n_x, n_y)$ contains the frequency envelope information from $b_{s,r}(n_x, n_y)$. Similar to $a_{s,r}(n_x, n_y)$, $u_{s,r}(n_x, n_y)$ has $R/2$ periodicity. Since $u_{s,r}(n_x, n_y)$ is real, $f_{Fs,q}(n_x, n_y)$ is conjugate symmetric in q , and consequently, its $0 \leq q \leq R/4$ components are sufficient for complete representation. It is rotation-invariant because the DFT operation maps rotationally induced shifts into the complex numbers' phase components, which are removed when the magnitude operation is performed.

$f_{Ys,q}(n_x, n_y)$ contains the directionality information from $b_{s,r}(n_x, n_y)$. Since $\phi_{s,(r+R/2)_R}(n_x, n_y) = \phi_{s,r}(n_x, n_y) + 180^\circ$, only the components with odd q are nonzero. For the same reason as $f_{Fs,q}(n_x, n_y)$, $f_{Ys,q}(n_x, n_y)$ is rotation-invariant.

$f_{DAs,q}(n_x, n_y)$, $f_{DFs,q}(n_x, n_y)$ and $f_{DYs,q}(n_x, n_y)$ contain the direction information from $b_{s,r}(n_x, n_y)$. Because $f_{DAs,q}(n_x, n_y)$ and $f_{DFs,q}(n_x, n_y)$ are conjugate symmetric in q , they are represented completely by their $0 \leq q \leq R/4$ components. However, the $q=0$ component is always zero since the DFTs are on real sequences in both cases. $f_{DYs,q}(n_x, n_y)$ has the same nonzero indexes as $f_{Ys,q}(n_x, n_y)$. $f_{DAs,q}(n_x, n_y)$, $f_{DFs,q}(n_x, n_y)$ and $f_{DYs,q}(n_x, n_y)$ are inherently rotation-variant since the phases of the DFT contain all of the direction information.

Since all transformations in this decomposition are invertible (assuming boundary conditions are available), it is possible to exactly reconstruct $b_{s,r}(n_x, n_y)$ from their microfeatures. Thus, $f_{As,p}(n_x, n_y)$, $f_{Fs,q}(n_x, n_y)$, $f_{Ys,q}(n_x, n_y)$, $f_{DAs,q}(n_x, n_y)$, $f_{DFs,q}(n_x, n_y)$ and $f_{DYs,q}(n_x, n_y)$ provide a nearly exact representation of $i(x, y)$.

4 The Texture Model

4.1 The Texture Micromodel

A texture may be modeled as a vector-valued random field $\mathbf{f} = [\mathbf{f}_A \mathbf{f}_F \mathbf{f}_Y \mathbf{f}_{DA} \mathbf{f}_{DF} \mathbf{f}_{DY}]^T$, where \mathbf{f}_A , \mathbf{f}_F , \mathbf{f}_Y , \mathbf{f}_{DA} , \mathbf{f}_{DF} and \mathbf{f}_{DY} are vectors containing the microfeature components for all s and p or q indexes. It is assumed that \mathbf{f} is stationary. Accurate modeling of \mathbf{f} is not practical from a computational point of view. Such modeling is also not needed if the objective is only texture classification (and not synthesis). Further, we assume a Gaussian distribution of \mathbf{f} strictly for mathematic tractability and simplicity, although many sample distributions were observed to be very non-Gaussian.

Given these assumptions, the *micromodel* for texture t is stated as the multivariate Gaussian probability distribution function:

$$p(\mathbf{f}|t) = \frac{1}{\sqrt{(2\pi)^{N_f} |C_{ft}|}} \exp\left(\frac{-(\mathbf{f} - \boldsymbol{\mu}_{ft}) C_{ft}^{-1} (\mathbf{f} - \boldsymbol{\mu}_{ft})^T}{2}\right) \quad (33)$$

where $\boldsymbol{\mu}_{ft} = E\{\mathbf{f} | t\}$ and $C_{ft} = E\{\mathbf{f}(\mathbf{f}^T | t\}$ – $E\{\mathbf{f} | t\}(E\{\mathbf{f}^T | t\}$, are the mean and covariance of \mathbf{f} , respectively, and N_f is the number of microfeatures.

4.2 Macrofeatures

While microfeatures can be used to represent a texture sample, microfeatures are spatially localized and do not characterize global attributes of textures. For instance, consider the textures in Fig. 2. Most of the spatial samples in the upper-right and lower-left quadrants of texture A would be classified as texture B based on microfeatures alone. Furthermore, \mathbf{f}_{DA} , \mathbf{f}_{DF} and \mathbf{f}_{DY} are rotation dependent, making them unsuitable for rotation-invariant classification.

For classification, a better texture model is derived from the micromodel parameters, $\boldsymbol{\mu}_{ft}$ and C_{ft} . For instance, for the two textures shown in Fig. 2, the standard deviations of \mathbf{f}_{DA} , \mathbf{f}_{DF} , and \mathbf{f}_{DY} provide excellent discrimination information not available in the microfeatures themselves. A texture t 's *macrofeatures* are defined to be $\mathbf{F} = [F_{CA} F_{CF} F_{CY} F_{AM} F_{FM} F_{YM} F_{DMA} F_{DMF} F_{DMY}]^T$, where

$$\begin{bmatrix} F_{CA} \\ F_{CF} \\ F_{CY} \\ F_{AM} \\ F_{FM} \\ F_{YM} \\ F_{DMA} \\ F_{DMF} \\ F_{DMY} \end{bmatrix} = \begin{bmatrix} E\{\mathbf{f}_A | t\} \\ E\{\mathbf{f}_F | t\} \\ E\{\mathbf{f}_Y | t\} \\ \sqrt{E\{\mathbf{f}_A^2\} - E\{\mathbf{f}_A\}^2} \\ \sqrt{E\{\mathbf{f}_F^2\} - E\{\mathbf{f}_F\}^2} \\ \sqrt{E\{\mathbf{f}_Y^2\} - E\{\mathbf{f}_Y\}^2} \\ \sqrt{E\{\mathbf{f}_{DA}^2\} - E\{\mathbf{f}_{DA}\} \times E\{\mathbf{f}_{DA}\}^*} \\ \sqrt{E\{\mathbf{f}_{DF}^2\} - E\{\mathbf{f}_{DF}\} \times E\{\mathbf{f}_{DF}\}^*} \\ \sqrt{E\{\mathbf{f}_{DY}^2\} - E\{\mathbf{f}_{DY}\} \times E\{\mathbf{f}_{DY}\}^*} \end{bmatrix} \quad (34)$$

where $\mathbf{f}^2 = \langle \mathbf{f}, \mathbf{f} \rangle = [(F_{CA} \ 0, 0(F_{CA} \ 0, 0) \ (F_{CA} \ 0, 1(F_{CA} \ 0, 1) \ \dots \ (f_{DY} \ S-1, R-1 \cdot f_{DY} \ S-1, R-1)]^T$. For a texture t , F_{CA} , F_{CF} and F_{CY} describe amplitude, frequency and directionality characteristics, respectively, of the "carrier." F_{AM} , F_{FM} and F_{YM} describe a texture's amplitude modulation, frequency

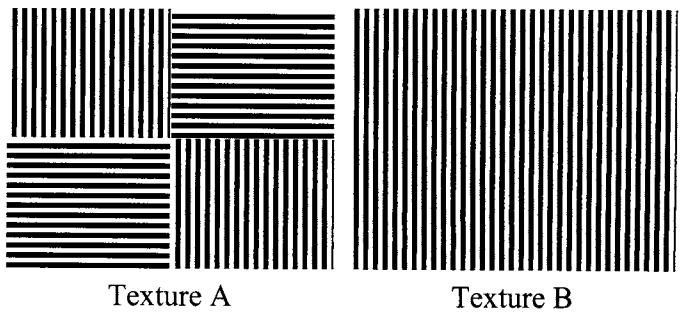


FIGURE 2 Textures with similar microfeatures.

modulation and directionality modulation characteristics, respectively. F_{CA} , F_{CF} , F_{CY} , F_{AM} , F_{FM} and F_{YM} are all rotation-invariant because the microfeatures upon which they are based are rotation-invariant. F_{DMA} , F_{DMF} and F_{DMY} capture the directional modulation characteristics. Although f_{DA} , f_{DF} and f_{DY} are rotation dependent, their variances are not. Means of f_{DA} , f_{DF} and f_{DY} are directional in nature and are not used as classification features. For simplicity, off-diagonal covariances are not used, although they may contain useful information.

The expected values of f are estimated using the mean and variance of a texture sample's microfeatures.

4.3 The Texture Macromodel

For purposes of classification, a texture t is modeled as a vector-valued Gaussian random vector \mathbf{F} with the conditional probability density function

$$p(\hat{\mathbf{F}}|t) = \frac{1}{\sqrt{(2\pi)^{N_F} |\mathbf{C}_{\mathbf{F}_t}|}} \exp\left(\frac{-\left(\hat{\mathbf{F}} - \boldsymbol{\mu}_{\mathbf{F}_t}\right) \mathbf{C}_{\mathbf{F}_t}^{-1} \left(\hat{\mathbf{F}} - \boldsymbol{\mu}_{\mathbf{F}_t}\right)^T}{2}\right) \quad (35)$$

where $\boldsymbol{\mu}_{\mathbf{F}_t} = E\{\mathbf{F} | t\}$ and $\mathbf{C}_{\mathbf{F}_t} = E\{\mathbf{F} \cdot \mathbf{F}^T | t\} - E\{\mathbf{F} | t\} \cdot E\{\mathbf{F}^T | t\}$ are the mean and covariance of \mathbf{F} , respectively, N_F is the number of macrofeatures, and $\hat{\mathbf{F}}$ is an estimate of \mathbf{F} based on a sample of texture t . This is the texture *macromodel*.

The parameters $\boldsymbol{\mu}_{\mathbf{F}_t}$ and $\mathbf{C}_{\mathbf{F}_t}$ are estimated from statistics over M samples for each texture t :

$$\begin{aligned} \hat{\boldsymbol{\mu}}_{\mathbf{F}_t} &= \frac{1}{M} \sum_{m=1}^M \hat{\mathbf{F}}_m \quad \text{and} \\ \hat{\mathbf{C}}_{\mathbf{F}_t} &= \frac{1}{M-1} \sum_{m=1}^M \left(\hat{\mathbf{F}}_m - \hat{\boldsymbol{\mu}}_{\mathbf{F}_t} \right) \cdot \left(\hat{\mathbf{F}}_m - \hat{\boldsymbol{\mu}}_{\mathbf{F}_t} \right)^T \end{aligned} \quad (36)$$

where $\hat{\mathbf{F}}_m$ is the estimate of \mathbf{F} based on sample m of texture t .

5 Experimental Results

Experiments were performed on two groups of textures. The first group comprises 13 texture images [47] digitized from the Brodatz album [6] and other sources. Each texture was digitized at rotations of 0, 30, 60, 90, 120, and 150 degrees as 512×512 pixels, each of which was then subdivided into $16 \times 128 \times 128$ subimages. Figure 3 presents the 120-degree rotations of these images. The second group comprises 109 texture images from the Brodatz album digitized at 0 degrees with 512×512 pixels at 300-DPI resolution, each of which was then subdivided into $16 \times 128 \times 128$ subimages. A polar, analytic Gabor transform was used with parameter values

$\omega_0 = 0.8\pi$, $\theta_0 = 0$ degrees, $S = 4$, $R = 16$, $\kappa = 0.283$ ($B = 1$ octave), and $\sigma_\theta = 0.0523/\text{degree}$ (-3-dB width of 90 degrees).

Classification performance was demonstrated with both groups of textures. Half of the subimages (separated in a checkerboard pattern) were used to estimate the model parameters (mean and covariance of the macrofeatures) for each type of texture, whereas the other half were used as test samples. Features were extracted from all of the subimages in an identical manner. To reduce filter-sampling effects at high frequencies due to rotation, the estimation of model parameters was based on the features from subimages at all rotations in the first group of images.

5.1 Classification

A model of each type of texture was established using half of its samples to estimate mean and covariance, the parameters required by (34). For the other half of the samples, each was classified as the texture t that maximized $p(\mathbf{F}|t)$. Because of rank deficiency problems in the covariance matrix due to high interfeature correlation, off-diagonal terms in the covariance matrix were set to zero.

Classification performance for the first group of textures is summarized in Table 1. Of 624 sample images, 604 (96.8%) were correctly classified. The misclassification rate per competing texture type is $(100\% - 96.8\%)/12 = 0.27\%$. Bark was misclassified as brick, bubbles, pigskin, sand, and straw; sand as bark; pigskin as bark and wool; grass as leather; leather as grass and straw; wool as bark and pigskin; water as straw; and wood as straw.

Classification performance for the second group of textures (the complete Brodatz album): Of 872 sample images, 701 (80.4%) were classified correctly. The misclassification rate per competing texture type is $(100\% - 80.4\%)/108 = 0.18\%$. Perhaps some comments are in order regarding the classification rate. Many of the textures in the Brodatz album are not

TABLE 1 Classification performance for first group of textures.

Sample Type	Classified Correctly, %
Bark	87.5
Sand	97.9
Pigskin	95.8
Bubbles	100
Grass	95.8
Leather	93.8
Wool	91.7
Raffia	100
Weave	100
Water	97.9
Wood	97.9
Straw	100
Brick	100

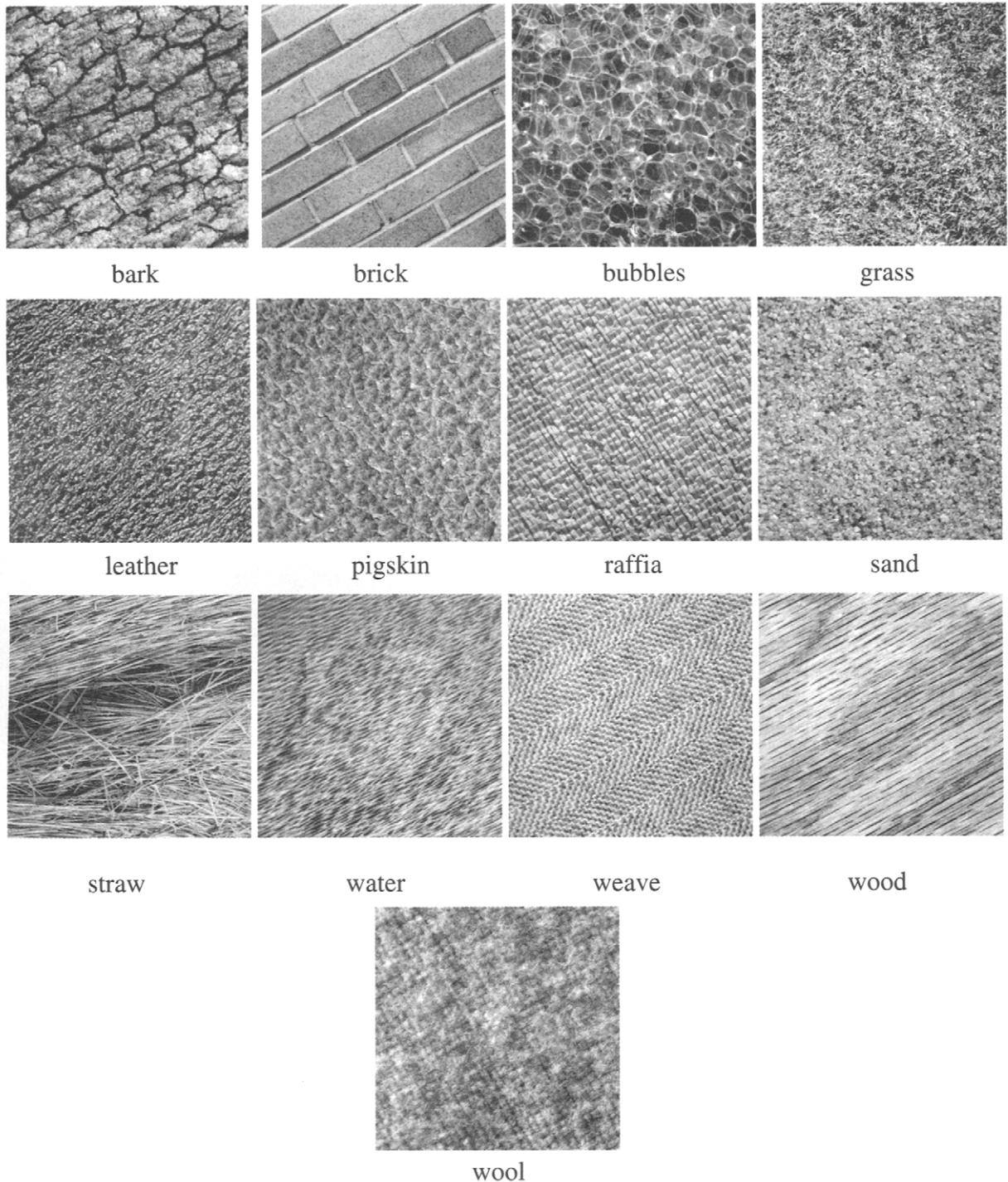


FIGURE 3 Textures from first group. Each texture was digitized at rotations of 0, 30, 60, 90, 120, and 150 degree. Table 1 summarizes the results for rotation invariant classification for these textures.

homogeneous. Although one can use a selected subset of textures, it will make comparisons between different algorithms more difficult. Finally, for comparison purposes, when using the same subset of the Brodatz album used by Chang and Kuo [7], 100% of the samples were correctly classified.

6 Image Segmentation Using Texture

Image segmentation can be achieved by classification or by considering the gradient in the texture feature space. Here we outline a novel technique, called *EdgeFlow*, that uses

the texture feature as input to partition the image. A detailed description of this technique can be found in Ma and Manjunath [30].

The *EdgeFlow* method uses a predictive coding model to identify and integrate the direction of change in a given set of image attributes, such as color and texture, at each image pixel location. Toward this objective, the following values are computed: $E(\mathbf{x}, \theta)$, which measures the edge energy at pixel \mathbf{x} along the orientation θ ; $P(\mathbf{x}, \theta)$, which is the probability of finding an edge in the direction θ from \mathbf{x} ; and $P(\mathbf{x}, \theta + \pi)$ which is the probability of finding an edge along $\theta + \pi$ from \mathbf{x} . These edge energies and the associated probabilities are computed using the features of interest.

Consider the Gabor filtered outputs represented by equation (21):

$$b_{s,r}(\mathbf{x}) = a_{s,r}(\mathbf{x}) \cdot \exp(j\psi_{s,r}(\mathbf{x}))$$

By taking the amplitude of the filtered output across different filters at the location represented by \mathbf{x} , a texture feature vector characterizing the local spectral energies in different spatial-frequency bands is formed:

$$\mathbf{A}(\mathbf{x}) = [a_1(\mathbf{x}), a_2(\mathbf{x}), a_3(\mathbf{x}), \dots, a_N(\mathbf{x})] \quad (37)$$

where, for simplicity the combination of s and r indices is numbered from 1 through N . The texture edge energy, which is used to measure the change in local texture, is computed as

$$E(\mathbf{x}, \theta) = \sum_{1 \leq i \leq N} |a_i(\mathbf{x}) * GD_{\sigma, \theta}(\mathbf{x})| \cdot w_i, w_i = 1 / \sum_{\mathbf{x}} a_i(\mathbf{x}) \quad (38)$$

where GD is the first derivative of the Gaussian along the orientation θ . The weights w_i normalize the contribution of edge energy from the various frequency bands. The error in predicting the texture energies in the neighboring pixel locations is used to compute the probabilities $\{P(s, \theta)\}$. For example, a large prediction error in a certain direction implies a higher probability of finding the region boundary in that direction. Thus, at each location \mathbf{x} we have $\{[E(\mathbf{x}, \theta), P(\mathbf{x}, \theta), P(\mathbf{x}, \theta + \pi)]|_{0 \leq \theta \leq \pi}\}$. From these measurements, an *edge flow* vector is constructed whose direction represents the *flow* direction along which a boundary is likely to be found, and whose magnitude is an estimate of the total edge energy along that direction.

The distribution of the edge flow vectors in the image forms a flow field that is allowed to propagate. At each pixel location, the flow is in the estimated direction of the boundary pixel. A boundary location is characterized by flows in opposing directions toward it. On a discrete image grid, the flow typically takes a few iterations to converge.

Figure 4 shows two images, one with different textures, and another with an illusory boundary. For the textured image, the edge flow vectors are constructed at each location as outlined

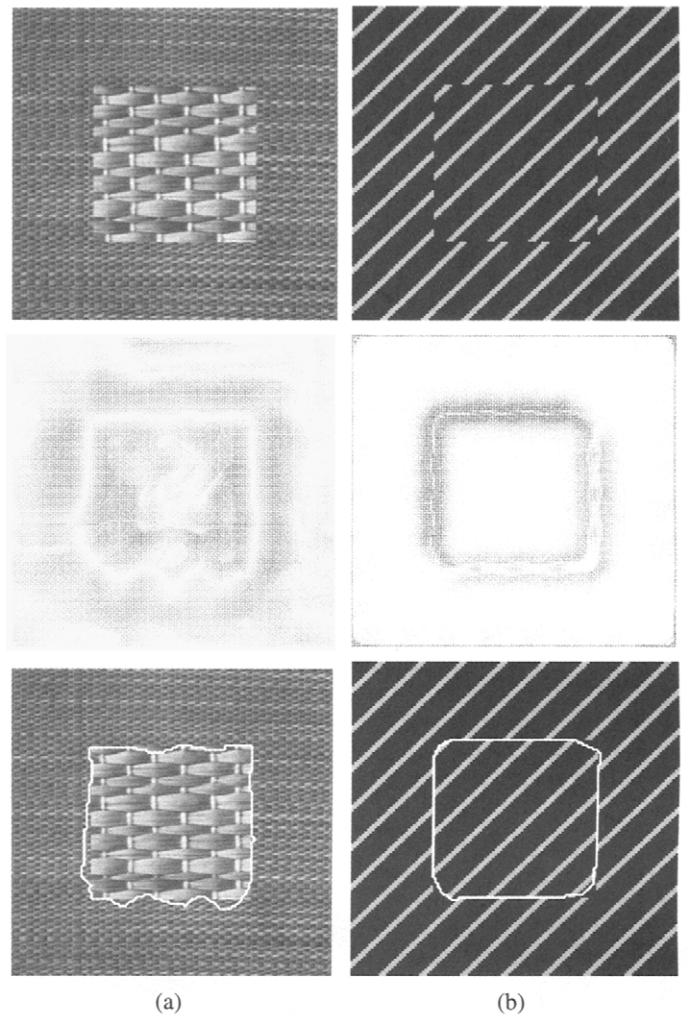


FIGURE 4 Segmentation using *EdgeFlow*. From top to bottom are the original image, edge flow vectors, and detected boundaries. (a) Texture image example. (b) An illusory boundary detected using the texture phase component from the Gabor filtered images.

above and the final segmentation result is shown in the figure. It turns out the phase information in the filtered outputs is quite useful in detecting illusory contours, as illustrated in the figure. The details of computing the phase discontinuities can be found in [30].

Figure 5 shows another example of texture-based segmentation, illustrating the results at two different choices for the scale parameter that controls the *EdgeFlow* segmentation. Other examples of using color, texture and phase in detecting image boundaries are shown in Fig. 6.

7 Image Retrieval Using Texture

In recent years, texture has emerged as an important visual feature for content-based image retrieval. We presented an

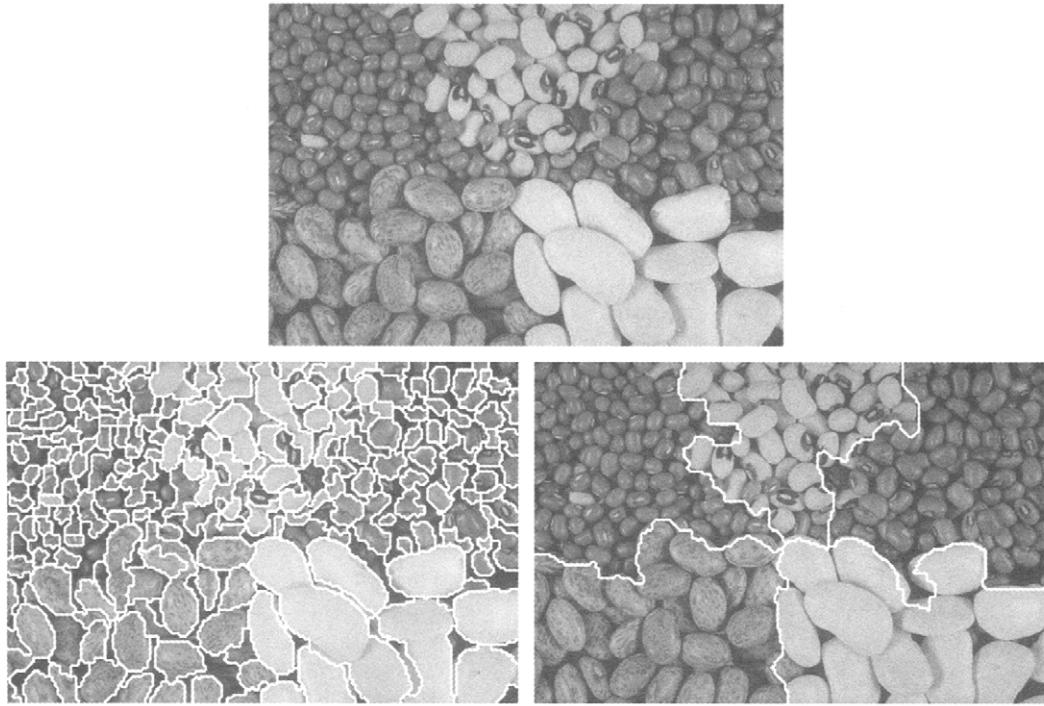


FIGURE 5 The choice of *scale* plays a critical role in the *EdgeFlow* segmentation. Two different segmentation results shown above are the result of two different choices for the scale parameter in the algorithm.

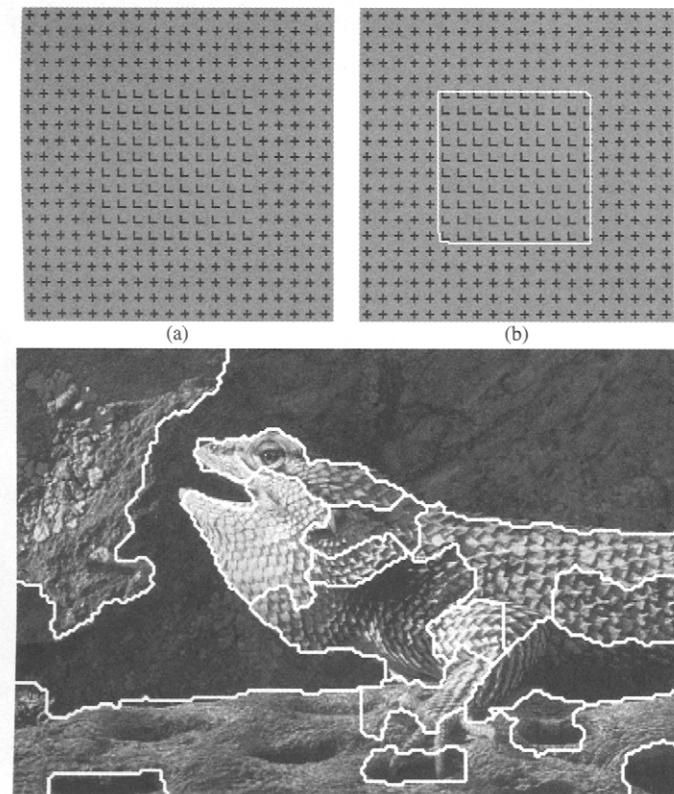


FIGURE 6 Two other examples of segmentation. (a) An illusory boundary, (b) segmentation using texture phase in the *EdgeFlow* algorithm, and (c) segmentation using color and texture energy.

image-retrieval system for browsing a collection of large aerial imagery using texture [31]. Texture turns out to be a surprisingly powerful descriptor for aerial imagery and many of the geographically salient features, such as vegetation, water, urban development, parking lots, airports, and so forth, are well characterized by their texture signature. The particular texture descriptor used our report [31] was based on the mean and standard deviation of $A(x)$ computed in (37). Measuring the similarity between two patterns in the texture feature space is an important issue in image retrieval. A hybrid neural network algorithm was used to learn this similarity and thus construct a *texture thesaurus* that would facilitate fast search and retrieval. Figures 7 and 8 show two *query by example* results wherein the input to the search engine was an image region, and the system was asked to retrieve similar looking patterns in the image database.

8 Texture Motifs

The previous section demonstrated how texture descriptors can be used to identify homogeneous regions in aerial images in a similarity-retrieval framework. However, a major challenge in using texture features to support more complex interaction with remote-sensed imagery is that the objects of interest often consist of multiple textures. For example, golf courses typically consist of grass-covered fairways lined by trees. Grass and trees each result in distinctive textures but neither by itself characterizes a golf course.

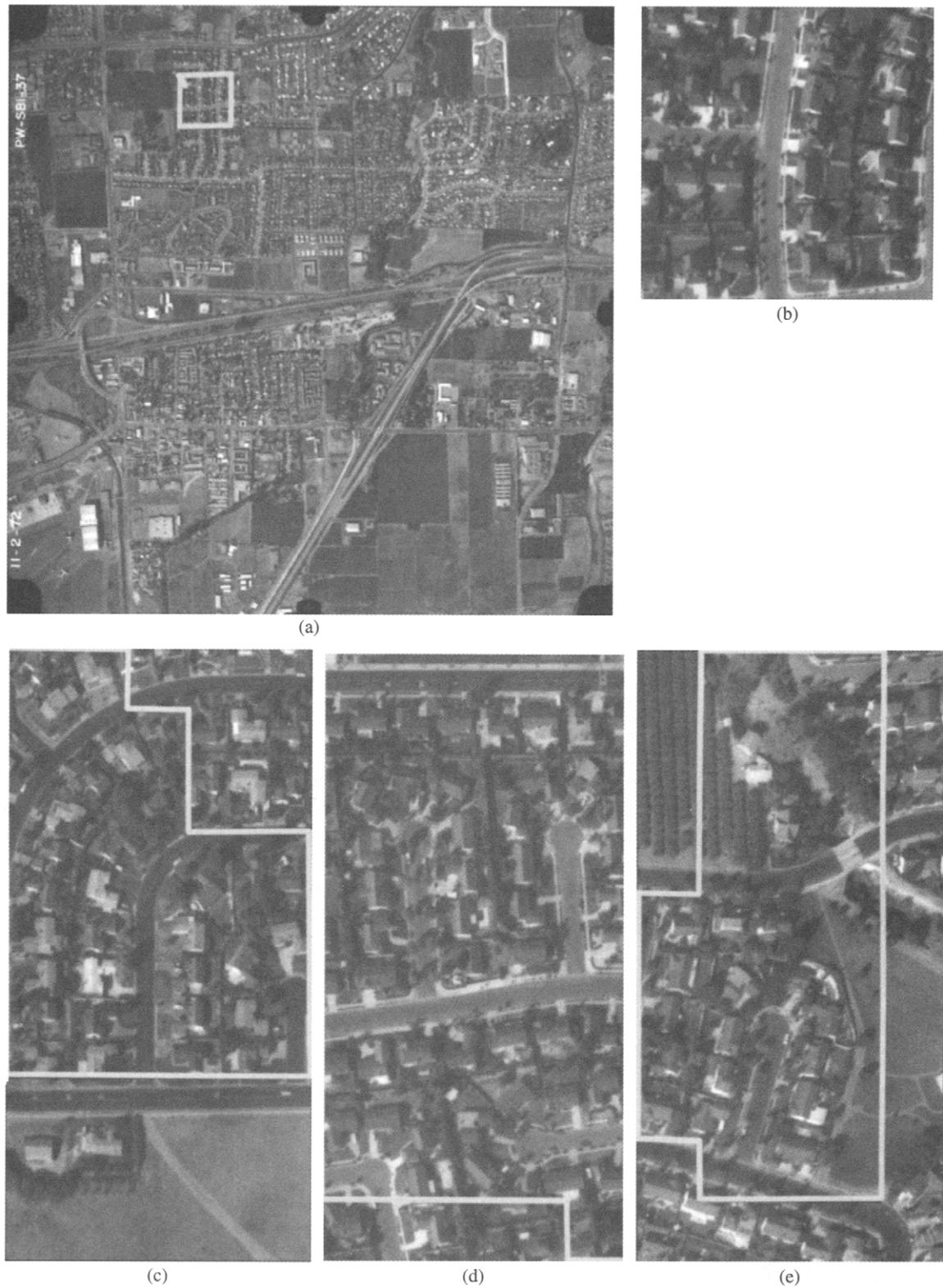


FIGURE 7 An example of a texture-based search using a data set of aerial photographs of the Santa Barbara area taken over a 30-year period. Each photograph is approximately 5000×5000 pixels in size. (a) shows the down-sampled version of the aerial photograph from which the query is derived. (b) shows a full-resolution detail of the region used for the query. The region contains a housing development. (c)–(e) show the ordered three best results of the query. The results come from three different aerial photographs that were taken the same year as the photograph used for the query.

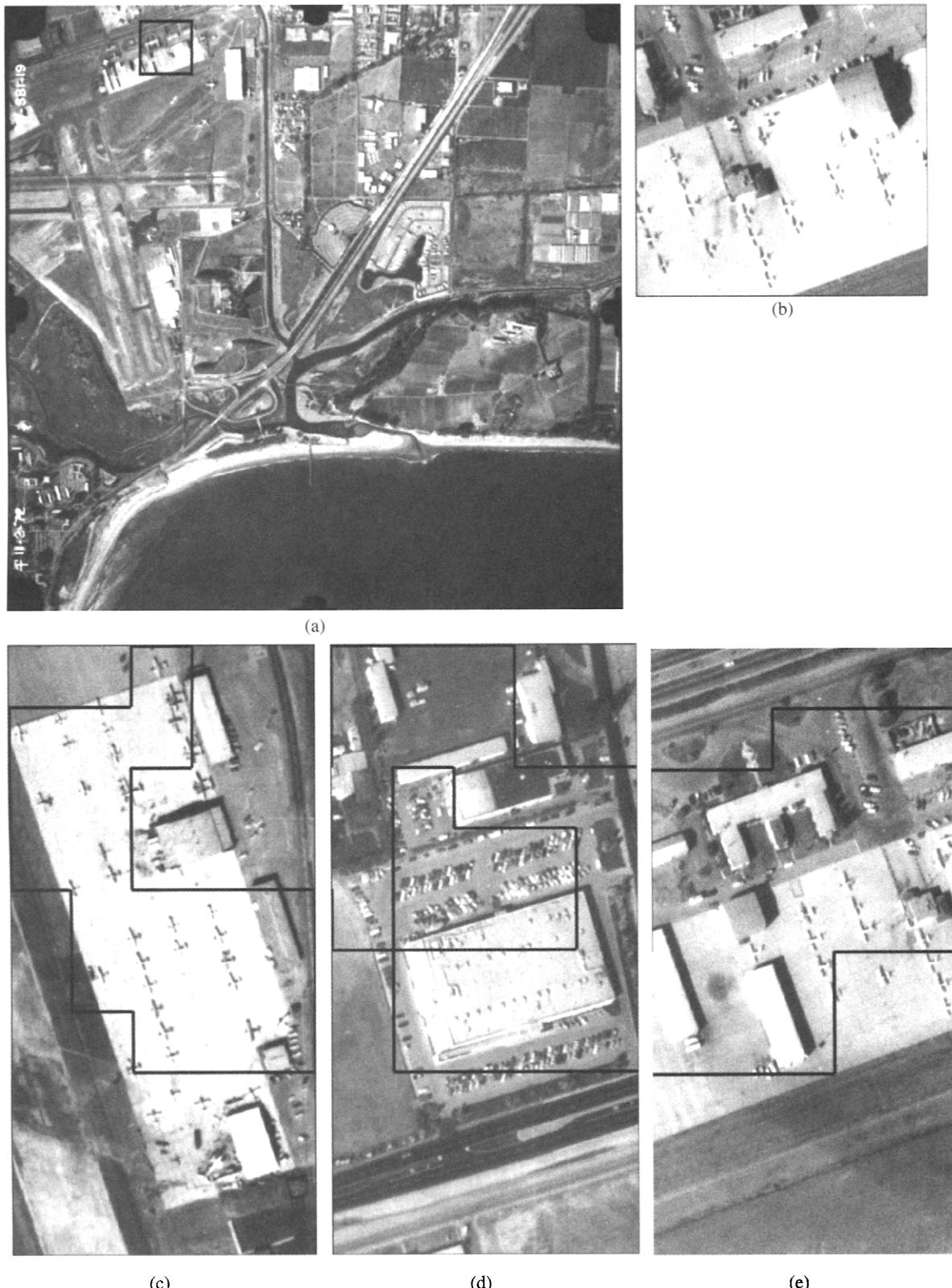


FIGURE 8 Another example of a texture based search. (a) shows the down-sampled version of the aerial photograph from which the query is derived. (b) shows a full-resolution detail of the region used for the query. The region contains aircraft, cars and buildings. (c)–(e) show the ordered three top matching retrievals. Once again, the results come from three different aerial photographs. This time, the second and third results are from the same year (1972) as the query photograph but the first match is from a different year (1996).

A texture-based representation of golf courses must account for the multiple shared textures. One way to do this is to use statistical models to characterize the *texture motifs* for a class of objects.

A texture motif is defined as *a characteristic spatial pattern common to a class of objects* [39]. Accurate characterizations of texture motifs can facilitate object recognition. However, developing effective characterizations is a challenge largely due to the high dimensionality of the texture feature spaces. A statistical framework can be adopted in which the texture motifs are modeled using Gaussian mixture models (GMMs) [3]. The model parameters can be learned in an unsupervised fashion from unlabeled training samples using the expectation maximization (EM) algorithm. The advantages of this approach are (a) preprocessing steps, such as segmentation, are not required; (b) the learning phase is completely automated, only requiring unlabeled training samples; and (c) the models are compact because the model form is fixed and only the parameter values differ from one class of objects to another.

Another benefit of the GMM/EM framework is that the texture motifs can be learned in an orientation-invariant manner using a broad category of texture descriptors *that need not be orientation-invariant themselves*. The texture descriptor based on the Gabor filter outputs as computed in (37) is an example of such a descriptor. This invariance is accomplished by letting the orientation of a motif be one of the missing parameters in the EM construction [39]. The significance of this result is that (a) only the “normalized” orientation of a texture motif needs to be characterized, and (b) the training set need not contain all possible orientations of a motif.

The statistical models learned using the GMM/EM framework can be used to classify novel image regions in order to facilitate object recognition. Pixels can be assigned motif labels using a maximum a posteriori (MAP) classifier. Figure 9

shows an example of such a labeling. The image in Fig. 9a contains two texture motifs at different orientations. A model learned from similar training images is then used to make the motif assignments in Fig. 9b and the orientation assignments in Fig. 9c. Note that even though the motifs are labeled in an orientation-invariant manner, the GMM/EM framework preserves knowledge of the motif orientations.

Figures 10b and 10c show the motif and orientation assignments for the aerial image of a harbor region in Fig. 10A, again using a model learned from other harbor regions. Such harbor regions contain several texture motifs, such as the water regions and the different densities of rows of boats. Note, again, that the motifs are labeled in an orientation-invariant manner, yet knowledge of the motif orientations is preserved.

9 Summary

We have presented schemes for texture classification and segmentation using features computed from Gabor-filtered images. Image texture research has seen much progress during the last two decades, and both random field model-based approaches and multiband filtering methods will have applications to texture analysis. Model-based methods are particularly useful for synthesis and rendering. Filtering methods compare favorably with the random field methods for classification and segmentation, and they can be efficiently implemented on dedicated hardware. Finally, texture features appear quite promising for image database applications such as search and retrieval. A variation of the Gabor filtering-based texture descriptor—referred to as the homogeneous texture descriptor—is included in the current ISO/IEC MPEG-7 multimedia content description interface standard [37].

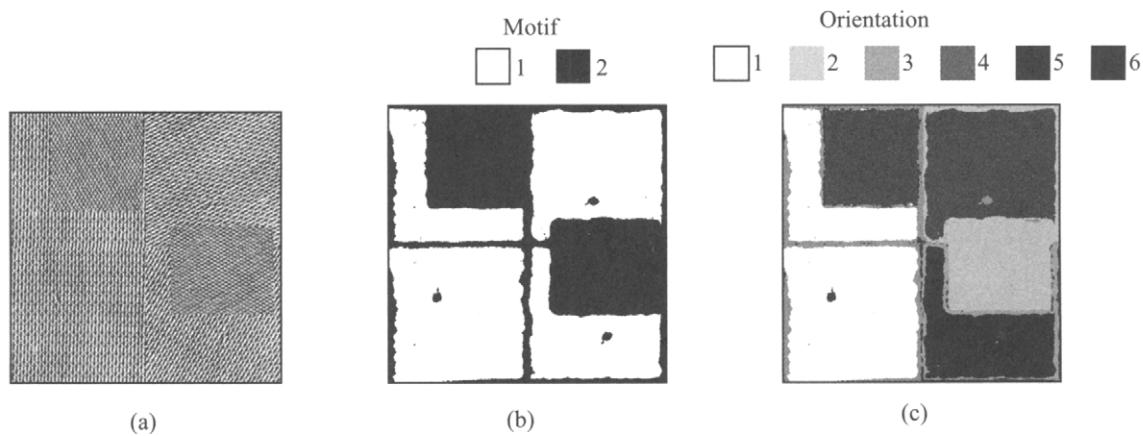


FIGURE 9 (a) An image with two texture motifs. (b) Motif assignments. Note that different orientations of a motif are labeled consistently. (c) Orientation assignments. Note that knowledge of motif orientation is preserved.

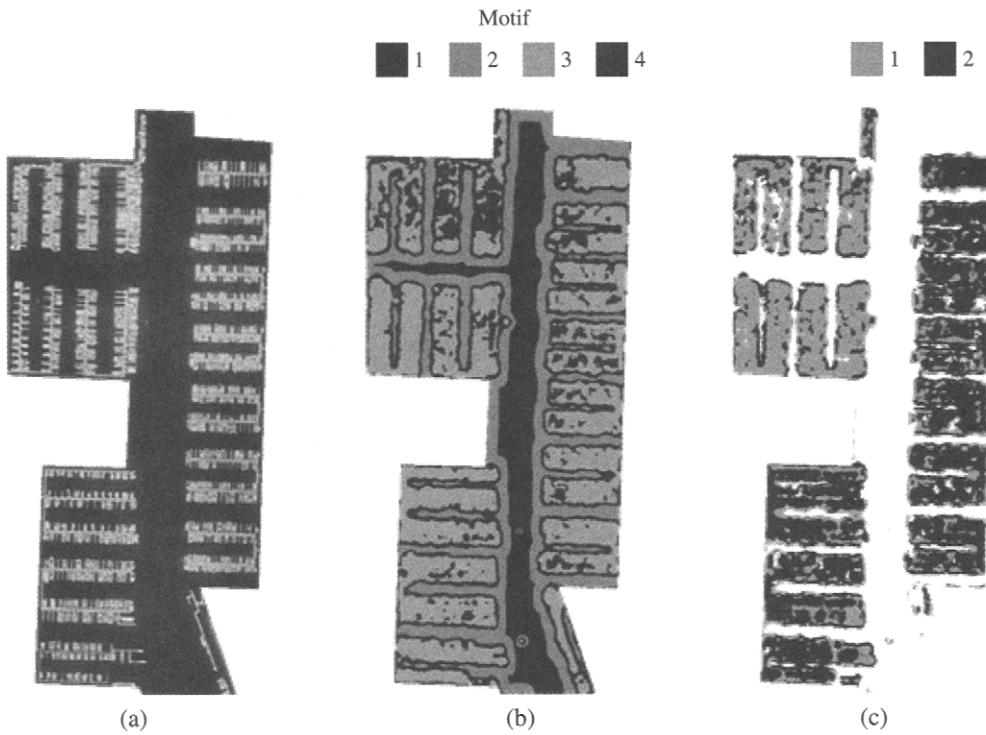


FIGURE 10 (a) An aerial image of a harbor region. (b) Motif assignments. Note that different orientations of a motif are labeled consistently. (c) Orientation assignments. Note that knowledge of motif orientation is preserved.

Acknowledgments

This work was supported in part by grants from National Science Foundation (NSF) (94-1130, 97-04795, EIA-0080134), the Office of Naval Research (ONR/AASERT) (N0014-98-0515, N0014-01-0391). This work was performed in part under the auspices of the U.S. Department of Energy by University of California, Lawrence Livermore National Laboratory under Contract W-7405-Eng-48. The authors would also like to thank Sitaram Bhagavathy for carefully proofreading the manuscript.

References

- [1] M. J. Bastiaans, "A sampling theorem for the complex spectrogram, and Gabor's expansion of a signal in Gaussian elementary signals," *Opt. Eng.* 20, 594–598 (1981).
- [2] M. J. Bastiaans, "Gabor's signal expansion and degrees of freedom of a signal," *Opt. Acta*, 29, 1223–1229 (1982).
- [3] S. Bhagavathy, S. D. Newsam, and B. S. Manjunath, "Modeling object classes in aerial images using texture motifs," *Proceedings of the IAPR International Conference on Pattern Recognition*, 2, 981–984 (2002).
- [4] A. C. Bovik, "Analysis of multichannel narrow-band filters for image texture segmentation," *IEEE Trans. Signal Process.* 39, 2025–2042 (1991).
- [5] A. C. Bovik, M. Clark, and W. S. Geisler, "Multichannel texture analysis using localized spatial filters," *IEEE Trans. Pattern Anal. Mach. Intell.* 12, 55–73 (1990).
- [6] P. Brodatz, *Textures: A Photographic Album for Artists and Designers* (Dover Press, New York, 1966).
- [7] T. Chang and C. C. J. Kuo, "Texture analysis and classification with tree-structured wavelet transform," *IEEE Trans. Image Process.* 2, 429–441 (1993).
- [8] R. Chellappa and S. Chatterjee, "Classification of textures using Gaussian Markov random field models," *IEEE Trans. Acoust. Speech Signal Process.* 33, 959–963 (1985).
- [9] R. Chellappa, R. L. Kashyap, and B.S. Manjunath, "Model-based texture segmentation and classification," *Handbook of Pattern Recognition and Computer Vision*, C. H. Chen, L. F. Pau, and P. F. P. Wang, eds. (World Scientific Publishing, Hackensack, NJ, 1992).
- [10] P. C. Chen and T. Pavlidis, "Segmentation by texture using correlation," *IEEE Trans. Pattern Anal. Mach. Intell.* 5, 64–69 (1983).
- [11] Y. Choe and R. L. Kashyap, "3-D shape from a shaded and textural surface image," *IEEE Trans. Pattern Anal. Mach. Intell.* 13, 907–918 (1991).
- [12] F. S. Cohen, Z. Fan, and M. A. Patel, "Classification of rotated and scaled textured image using Gaussian Markov random field models," *IEEE Trans. Pattern Anal. Mach. Intell.* 13, 192–202 (1991).
- [13] G. R. Cross and A. K. Jain, "Markov random field texture models," *IEEE Trans. Pattern Anal. Mach. Intell.* 5, 25–39 (1983).
- [14] J. G. Daugman, "Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression," *IEEE Trans. Acoust. Speech Signal Process.* 36, 1169–1179 (1988).
- [15] J. G. Daugman, "Uncertainty relation for resolution in space, spatial frequency and orientation optimized by

- two-dimensional visual cortical filters," *J. Opt. Soc. Amer.* 2, 1160–1169 (1985).
- [16] H. Derin and H. Elliott, "Modeling and segmentation of noisy and textured images using Gibbs random fields," *IEEE Trans. Pattern Anal. Mach. Intell.* 9, 39–55 (1987).
- [17] H. Derin, H. Elliott, R. Cristi, and D. Geman, "Bayes smoothing algorithms for segmentation of binary images modeled by Markov random fields," *IEEE Trans. Pattern Anal. Mach. Intell.* 6, 707–720 (1984).
- [18] D. Dunn, W. E. Higgins, and J. Wakeley, "Texture segmentation using 2-D Gabor elementary functions," *IEEE Trans. Pattern Anal. Mach. Intell.* 16, 130–149 (1994).
- [19] D. Gabor, "Theory of communication," *J. Inst. Elect. Eng.* 93, 429–457 (1946).
- [20] M. M. Galloway, "Texture analysis using gray level run lengths," *Comput. Graph. Image Process.* 4, 172–179 (1975).
- [21] H. Greenspan, S. Belongie, R. Goodman, and P. Perona, "Rotation invariant texture recognition using a steerable pyramid," *Proc. IEEE International Conference on Image Processing* (Jerusalem, Israel) (October 1994).
- [22] G. M. Haley and B. S. Manjunath, "Rotation invariant texture classification using a complete space-frequency model," *IEEE Trans. Image Process.* 8, 256–269 (1999).
- [23] R. Haralick and R. Bosley, "Texture features for image classification," *Third ERTS Symposium, NASA SP-351* (1973), pp. 1219–1228.
- [24] R. L. Kashyap, R. Chellappa, and A. Khotanzad, "Texture classification using features derived from random field models," *Patt. Recog. Lett.* 43–50 (1982).
- [25] R. L. Kashyap and A. Khotanzad, "A model-based method for rotation invariant texture classification," *IEEE Trans. Pattern Anal. Mach. Intell.* 8, 472–481 (1986).
- [26] S. Krishnamachari and R. Chellappa, "Multiresolution Gauss-Markov random fields for texture segmentation," *IEEE Trans. Image Process.* 6, 251–267 (1997).
- [27] S. Lakshmanan and H. Derin, "Simultaneous parameter estimation and segmentation of Gibbs random fields using simulated annealing," *IEEE Trans. Pattern Anal. Mach. Intell.* 11, 799–813 (1989).
- [28] K. Laws, Textured Image Segmentation, Ph.D. thesis, University of Southern California, 1978.
- [29] M. M. Leung and A. M. Peterson, "Scale and rotation invariant texture classification," *Proc. 26th Asilomar Conference on Signals, Systems and Computers* (Pacific Grove, CA, October 1992).
- [30] W. Y. Ma and B. S. Manjunath, "EdgeFlow: A framework of boundary detection and image segmentation," *Proceedings of the IEEE International Conference on Computerized Vision Pattern Recognition* (San Juan, Puerto Rico, June 1997), pp. 744–749.
- [31] W. Y. Ma and B. S. Manjunath, "A texture thesaurus for browsing large aerial photographs," *J. Amer. Soc. Inf. Science*, 49, 633–648 (1998).
- [32] J. Malik and P. Perona, "Preattentive texture discrimination with early vision mechanisms," *J. Opt. Soc. Amer. A*, 7, 923–932 (1990).
- [33] V. Maniyan and R. Vasquez, "Scaled and rotated texture classification using a class of basis function," *Patt. Recog.* 31, 1937–1948 (1998).
- [34] B. S. Manjunath, T. Simchony, and R. Chellappa, "Stochastic and deterministic networks for texture segmentation," *IEEE Trans. Acoust. Speech Signal Process.* 38, 1039–1049 (1990).
- [35] B. S. Manjunath and R. Chellappa, "A unified approach to boundary perception: edges, textures and illusory contours," *IEEE Trans. Neural Networks*, 4, 96–108 (1993).
- [36] B. S. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Trans. Pattern Anal. Mach. Intell.* 18, 837–842 (1996).
- [37] B. S. Manjunath, P. Salembier, and T. Sikora, *Introduction to MPEG-7: Multimedia Content Description Interface* (Wiley International, London, 2002).
- [38] S. Marcelja, "Mathematical description of the responses of simple cortical cells," *J. Opt. Soc. Amer.* 70, 1297–1300 (1980).
- [39] S. Newsam and B. S. Manjunath, "Normalized texture motifs and their application to statistical object modeling," *Proc. IEEE Intl. Conf. Computer Vision and Pattern Recognition: Workshop on Perceptual Organization in Computer Vision*, (Washington, DC, June 2004).
- [40] O. Pichler, A. Teuner, and B. J. Hosticka, "A comparison of texture feature extraction using adaptive Gabor filtering, pyramidal and tree structured wavelet transforms," *Pattern Recog.* 29, 733–742 (1996).
- [41] O. Pichler, A. Teuner, and B. J. Hosticka, "An unsupervised texture segmentation algorithm for feature space reduction and knowledge feedback," *IEEE Trans. Image Process.* 7, 53–61 (1998).
- [42] M. Porat and Y. Y. Zeevi, "The generalized scheme of image representation in biological and machine vision," *IEEE Trans. Pattern Anal. Mach. Intell.* 10, 452–468 (1988).
- [43] M. Porat and Y. Zeevi, "Localized texture processing in vision: analysis and synthesis in the Gaborian space," *IEEE Trans. Biomed. Eng.* 36, 115–129 (1989).
- [44] R. Porter and N. Canagarajah, "Robust rotation invariant texture classification: wavelets, Gabor filter, and GMRF based schemes," *IEEE Proc. Vis. Image Signal Process.* 144, 180–188 (1997).
- [45] M. R. Turner, "Texture discrimination by Gabor functions," *Biol. Cybern.* 55, 71–82 (1986).
- [46] J. You and H. A. Cohen, "Classification and segmentation of rotated and scaled textured images using 'tuned' masks," *Patt. Recog.* 26, 245–258 (1993).
- [47] Signal and Image Processing Institute, University of Southern California (<http://sipi.usc.edu>).