Designing Multiple Gabor Filters for Multi-Texture Image Segmentation

Thomas P. Weldon and William E. Higgins,†
Department of Electrical and Computer Engineering, University of North Carolina at Charlotte
†Department of Electrical Engineering, The Pennsylvania State University

Abstract

We consider the problem of segmenting multi-textured images using multiple Gabor filters. In particular, we present a mathematical framework for a multichannel texture-segmentation system consisting of a parallel bank of filter channels, a vector classifier stage, and a postprocessing stage. The framework establishes mathematical relationships between the predicted texture-segmentation error, the frequency spectra of constituent textures, and the parameters of the filter channels. The framework also permits the systematic formulation of filter-design procedures and provides predicted vector output statistics that are useful for classifier design. This paper focuses on the mathematical framework and provides experimental results that confirm the utility of the framework in the design of a complete image-segmentation system. The results demonstrate effective segmentation using a straightforward classifier and fewer than half the number of filters needed in previously proposed approaches.

Subject terms: Gabor prefilter, Gabor filter, Gabor function, texture segmentation, statistical image analysis, texture analysis, computer vision, image segmentation.

1 Introduction

Gabor filters have been employed in such diverse applications as texture segmentation,[1–12] image retrieval,[13] document analysis,[14,15] image coding,[16,17] retina identification,[18] character recognition,[19] target detection,[20,21] fractal dimension measurement,[22] edge detection,[23] line characterization,[24] medical image compression,[25] and image representation.[26] Despite such wide ranging applications, a variety of issues still remain when considering the problem of designing multiple Gabor filters for segmenting multi-textured images. Can a small number of well-designed filters effectively segment an image? Can detailed mathematical criteria be developed that lead to the specification of effective filters? Based on the mathematical criteria, can efficient filter-design procedures be devised? To address these issues, we propose a comprehensive mathematical framework for segmenting multi-textured images using multiple Gabor filters. [1]

Prior investigators have explored a variety of approaches to texture segmentation using Gabor filters. We group these efforts into two categories: *filter-bank approaches* where the Gabor filters are selected from some predetermined subband, wavelet, or other decomposition partitioning the frequency plane, and *filter-design approaches* where the Gabor filters are designed for a specific texture-segmentation task.

A number of different filter-bank approaches have been proposed. A wavelet decomposition modeled after biological vision systems was used by Daugman,[16] and a similar decomposition was used by Jain and Farrokhnia.[3] Other decompositions were proposed by Randen and Husøy,[29] Unser,[5] Raghu and Yegnanarayana,[30] Wu and Bhanu,[31] Manjunath and Ma,[13] Manjunath and Chellapa,[32] Tang *et al.*,[33] Turner,[34] Malik and Perona,[35] Bigün and du Buf,[8,36] Chang and Kuo,[37] and Lu *et al.*[12] In these approaches, the frequency domain was subdivided in various ways using a set of predetermined candidate filters that were not necessarily optimal for a given texture-segmentation task. The number of filters in the resulting filter banks can also present an unacceptable computational burden for certain applications. Furthermore, large feature-vector dimensions at the output of a large filter bank may require a complicated classifier and may be subject to "the curse

of dimensionality".[38] Although Tang *et al.*[33] reduced the number of filters to two, their method focuses on bipartite (two-texture) segmentations. Finally, filters in wavelet decompositions imply an initial decomposition of the frequency domain; also, the sensitivity of wavelet coefficients to translation can present problems.[5]

Other investigators have considered filter-design approaches, where the filters were designed for a particular texture-segmentation task. The use of filters tailored to a specific task offers the potential of reducing the segmentation error or reducing the number of filters. However, prior filter-design approaches were typically limited to the design of one filter per texture or one filter per texture pair.[1,2] The center frequency of each Gabor filter was set equal to a peak frequency in the spectrum of the corresponding texture, and the Gabor-filter bandwidth was set in proportion to its center frequency. A similar approach was suggested by Tan.[11] A second filter-design approach developed by Dunn *et al.* employed a detailed procedure for designing a single filter to segment two textures.[6,7] In this approach, measured output statistics and a Rician statistical model were used to predict image-segmentation error and establish the filter design. More recently, Teuner *et al.*[39] presented a method for selecting the best filters from a pyramidal Gabor wavelet decomposition. Recently, Casasent *et al.* described approaches employing composite filters comprised of several Gabor filters with filter parameters tailored to recognizing targets in clutter.[20,21,40] The work, however, focused on target detection rather than texture segmentation, employed initial filter parameters based on nominal target characteristics, combined filters into a macro-Gabor filter, used a fixed number of filters, and employed a neural network approach. Finally, Carevic and Caelli[41] described a multiple-filter design procedure based on clustering in the frequency domain. Although the method provided relatively free selection of filter parameters, the filter design was based on clustering in the frequency spectrum rather than a direct measure of predicted segmentation error.

These filter-design approaches have a variety of limitations. Some approaches focused on the filter response to one or two textures rather than the filter response to all textures. Bovik *et al.* selected filters that responded principally to a single texture.[1,2] Also, prior methods of the present authors were limited to the design of a single filter for segmenting a bipartite image[6,7,10] or, more recently, to the design of a single filter to segment multiple textures.[9] We have previously considered Gabor prefilter design in concert with a Gaussian postfilter,[9] but the design of a set of prefilters in concert with a set of postfilters remains to be addressed. (Since our approach incorporates *Gaussian postfilters*, as later described in Fig. 1, we refer to the Gabor filters used in our approach as *Gabor prefilters*.) These earlier approaches also focused on the scalar output of individual filters rather than the vector output of multiple filters. In addition, wavelet-decompositions, as in Teuner *et al.*,[39] are often limited to octave-bandwidth filters that may not necessarily be the best bandwidth for a particular segmentation task. The multi-filter design approach of Carevic and Caelli[41] used initial clustering that relied on distinct spectral features, limited radial bandwidths to one octave, merged frequency regions under 100 pixels (effectively limiting bandwidth), and incorporated other parameters restricting the number of frequency rings or affecting the filter-design clustering results. Finally, the previous methods do not provide a comprehensive mathematical framework that leads to a filter-design method based directly on predicted segmentation error.

To address the design of a multichannel system for texture segmentation, we present a new mathematical framework for the problem. This framework forms the basis for the design of multiple Gabor prefilters and multiple Gaussian postfilters for texture segmentation. The framework provides mathematical relationships between the power spectrum of the textures, the parameters of the Gabor prefilters, the parameters of subsequent Gaussian postfilters, the vector output statistics of multiple channels, and the predicted image-segmentation error. It also provides vector output statistics that can be used to design a Bayesian classifier.

In Section 2, we define the problem under consideration and introduce the multichannel system for segmenting multi-textured images. The mathematical framework for the problem is then developed in Section 3. This framework leads to efficient procedures for designing the filters and other components of the multichannel system. Section 4 presents experimental results supporting the efficacy of our approach. These results demonstrate, for example, effective segmentation of an eight-texture image using only two filters. Thus, our methodology can lead to significantly simpler systems than prior approaches.

2 Problem Statement

Fig. 1 illustrates the multichannel system that we use for segmenting multi-textured images. In Fig. 1, the input image is filtered through parallel channels.

Each channel, $j = 1, 2, \ldots, k$, consists of a Gabor prefilter $h_j(x, y)$, magnitude operator $|\cdot|$, and Gaussian postfilter $g_{p_j}(x, y)$. These combined channel outputs are classified by the vector classifier. The output of the classifier is then passed through a postprocessing stage to form the final segmented image. Although the development focuses on the case of multiple channels ($k > 1$), our methods can be applied to the design of a single channel.

The $N \times N$ pixel input image is denoted $I(x, y)$ in the system of Fig. 1. The image is assumed to be composed of $\mathcal{N} \geq 2$ different textures denoted $t_1, t_2, \ldots, t_{\mathcal{N}}$. Representative samples of each of the $\mathcal{N}$ textures are used to design the k channels, the classifier, and the postprocessing stage. In principle, the system of Fig. 1 is is similar to other recently proposed multichannel segmentation approaches.[2,3,5,33,42] But, in our work, the filter parameters, classifier, and postprocessing components remain open for explicit design; i.e., the number of channels, the Gabor prefilter parameters, the Gaussian postfilter parameters, the classifier parameters, and the postprocessing all remain to be specified. Further, our approach considers the vector behavior of all channels in addition to the behavior of individual channels.

We now give further details on the configuration of the multichannel system in Fig. 1. In the jth channel, the input image $I(x, y)$ is first filtered with a bandpass *Gabor prefilter* having a spatial impulse response $h_j(x, y)$ given by

$$\begin{aligned} h_j(x,y) &= g_j(x,y)\ e^{-j2\pi(u_j x + v_j y)} \\ &= \frac{1}{2\pi\sigma_{g_j}^2}\, e^{-\frac{(x^2+y^2)}{2\sigma_{g_j}^2}}\ e^{-j2\pi(u_j x + v_j y)}\ , \end{aligned} \tag{1}$$

where $g_j(x, y)$ is a two-dimensional Gaussian and (x, y) are spatial coordinates. The impulse response $h_j(x, y)$ is a complex sinusoid with *center frequency* (u_j, v_j) that is modulated by a Gaussian envelope.[6] The scale, or size, of the envelope of $h_j(x, y)$ is determined by σ_{g_j}. For simplicity, a symmetric Gaussian envelope $g_j(x, y)$ is used. The effect of an asymmetric Gaussian envelope is given elsewhere,[7] and the present methods can accommodate asymmetric $g_j(x, y)$ by replacing the symmetric form given in (1).

Taking the Fourier transform of $h_j(x, y)$, the frequency response of the jth Gabor prefilter is

$$H_j(u, v) = \mathcal{F}\{h_j(x, y)\} = G_j(u - u_j, v - v_j)\ , \tag{2}$$

where $\mathcal{F}\{\cdot\}$ is the Fourier transform operator, and $G_j(u, v)$ is the Fourier transform of the Gaussian $g_j(x, y)$:

$$G_j(u, v) = \mathcal{F}\{g_j(x, y)\} = \ e^{-2\pi^2\sigma_{g_j}^2(u^2+v^2)}\ . \tag{3}$$

The output of the prefilter $i_{h_j}(x, y)$ is then the convolution of the input image with the filter response,

$$i_{h_j}(x, y) = h_j(x, y) * \ I(x, y)\ , \tag{4}$$

where $*$ denotes convolution in two dimensions. The subscript "h_j" in $i_{h_j}(x, y)$ indicates the output of Gabor prefilter $h_j(x, y)$ in the jth channel. The next processing step in the jth channel is to compute the magnitude of the output of the Gabor prefilter

$$m_j(x, y) = |i_{h_j}(x, y)|. \tag{5}$$

The statistics of $m_j(x, y)$ have been shown to be approximately Rician for bandpass-filtered textures and are later used in estimating the statistics of the channel outputs.[6,9,10] The final step in each channel is to apply a lowpass Gaussian postfilter $g_{p_j}(x, y)$ to the prefilter output $m_j(x, y)$ yielding the postfiltered image

$$m_{p_j}(x, y) \ = \ m_j(x, y) * \ g_{p_j}(x, y)\ , \tag{6}$$

with

$$g_{p_j}(x, y) \ = \ \frac{1}{2\pi\sigma_{p_j}^2}\, e^{-\frac{(x^2+y^2)}{2\sigma_{p_j}^2}}\ , \tag{7}$$

where the parameter σ_{p_j} determines the Gaussian postfilter in the jth channel.

The four parameters $(u_j, v_j, \sigma_{g_j}, \sigma_{p_j})$ completely determine the Gabor prefilter and Gaussian postfilter in the jth channel. The values of these four filter

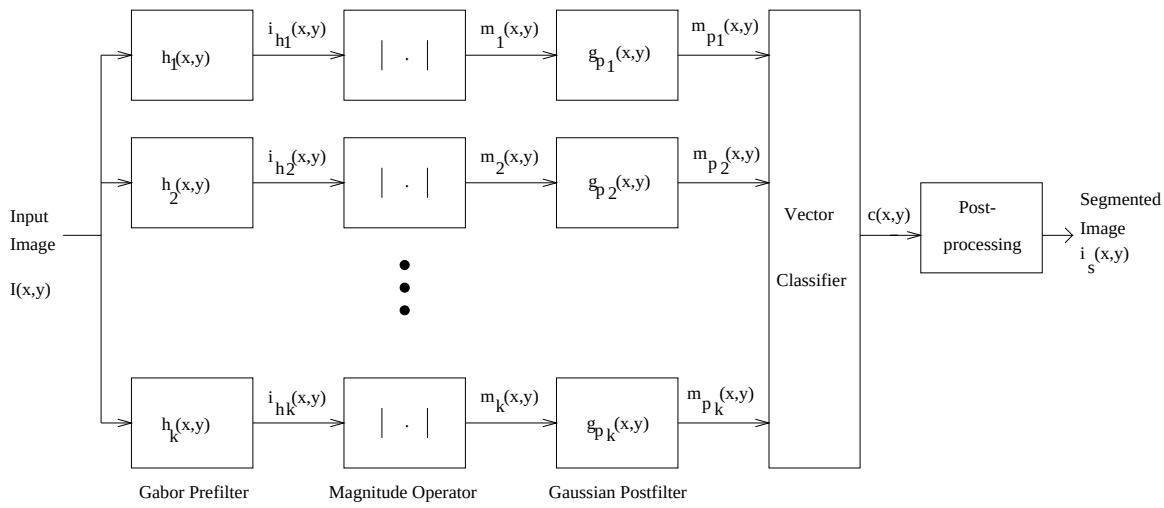

Figure 1: Multichannel system for texture segmentation. The number of channels is not predetermined, and can be varied as part of the design process.

parameters are not only free to vary within a given channel, but are free to vary from channel to channel. The number of channels k in the system is also free to vary and is not set *a priori*. Within each channel, we refer to $i_{h_j}(x, y)$ as the *prefiltered image*, $m_j(x, y)$ as the *prefilter output*, and $m_{p_j}(x, y)$ as the *postfilter output*.

The outputs of the k channels form a k-dimensional feature vector for each of the N^2 pixels in the input image $I(x, y)$. The vector classifier then generates an $N \times N$ classified image $c(x, y)$ from the k postfilter outputs. A Bayesian classifier, based on predicted multivariate output statistics, is used. Finally, the classifier output is postprocessed to improve performance at boundaries between different textures. The output of postprocessing is the final segmented $N \times N$ image $i_s(x, y)$. $\mathcal{N}$ different gray-scale values are used in the segmented image $i_s(x, y)$ to represent the $\mathcal{N}$ textures.

For the multichannel system of Fig. 1, the problem is to design a set of k channels and associated classifier that minimizes the predicted texture-segmentation error, given representative samples of $\mathcal{N}$ textures.

3 Mathematical Framework

Our approach to designing the system of Fig. 1 is to develop a mathematical framework that is useful both in describing system behavior and in designing system components. The proposed framework provides relationships between the parameters of the k Gabor prefilters, the parameters of the k Gaussian postfilters, the multivariate probability density of the k-dimensional output vector, and the predicted image-segmentation error. Section 3.1 first presents relationships between the sample textures, filter parameters, and multivariate probability density of the k filter outputs. The predicted multivariate probability density is then used to estimate segmentation error in Section 3.2. Finally, the effects of errors in the vicinity of texture boundaries are considered in Section 3.3.

3.1 Statistical Model

We begin by presenting a statistical model for the vector output of the k channels. A prefiltered texture is first modeled as a dominant sinusoid plus bandpass noise. This bandpass model then leads to a Rician distribution for the magnitude of the prefiltered texture. Next, a Gaussian distribution is used for the postfilter outputs of individual channels. The Gaussian statistics are derived from the foregoing Rician statistics. Finally, a multivariate Gaussian distribution is proposed for the vector output of the k channels.

Previously, we had shown that the output of a Gabor prefilter can be modeled as a dominant complex exponential plus bandlimited noise.[9,10] In this work, the bandpass frequency spectrum of the Gabor-filtered texture is decomposed into a dominant complex sinusoidal component plus a remainder component. Let A_{ij} represent the amplitude of the dominant spectral component at some frequency (u_{ij}, v_{ij}) within the passband for channel j and texture t_i, and

let $n_{ij}(x,y)$ represent the remaining portion of the passband signal for channel j and texture t_i. Then, the output of the Gabor prefilter $i_{h_j}(x,y)$ for texture t_i is denoted $i_{h_{ij}}(x,y)$ and can be approximated as

$$i_{h_{ij}}(x,y) \approx A_{ij}\, e^{j2\pi(u_{ij}\,x+v_{ij}\,y+\theta_{ij})} + n_{ij}(x,y) \quad (8)$$

where θ_{ij} accounts for variable phase shift in the complex sinusoid and where (u_{ij}, v_{ij}) is the dominant frequency within the passband of the Gabor prefilter $h_j(x,y)$.

The envelope of a sinusoid plus bandpass white Gaussian noise follows a Rician probability density function.[43–45] Similarly, the bandpass model in (8) suggests a Rician distribution for $m_j(x,y)$, and we have previously shown this to be the case.[9,10] Although it is apparent that the residual passband energy represented by $n_{ij}(x,y)$ is not necessarily "noise-like," this model produces effective results in practice while avoiding increased complexity.

Let $m_{ij}(x,y)$ denote the prefilter output $m_j(x,y)$ in channel j when the input texture is t_i. Then, the Rician probability density function $p_i(m_j;\ A_{ij}, N_{ij})$ describing the statistics of $m_{ij}(x,y)$ is:[9,10]

$$p_i(m_j;\ A_{ij}, N_{ij}) \approx \frac{2m_j}{N_{ij}}\, e^{-(\frac{m_j^2+A_{ij}^2}{N_{ij}})}\, I_0(\frac{2m_jA_{ij}}{N_{ij}}) \quad (9)$$

where $m_j \in m_{ij}(x,y)$, and $I_0(\cdot)$ is the modified Bessel function of the first kind with zero order.[6,43–45] A_{ij}^2 represents the power of the dominant sinusoid, and N_{ij} represents the remaining power in the passband modeled as noise. The overall approach is general in that the bandpass image is modeled as a dominant complex sinusoidal component with the difference between the filtered image and the sinusoid being attributed to the noise term $n_{ij}(x,y)$.

Given the Rician density function, we have also shown previously that the parameters A_{ij} and N_{ij} can be estimated from the power spectrum of the sample textures t_i.[9] A smoothed spectral estimate $P_i(u,v,\sigma_g)$ is used to find A_{ij} and N_{ij}. Let $S_i(u,v)$ be the power spectrum of texture t_i, and define $P_i(u,v,\sigma_g)$ as the frequency-domain convolution

$$P_i(u,v,\sigma_{g_j}) = |G_j(u,v)|^2 * S_i(u,v)\,, \quad (10)$$

where $|G_j(u,v)|$ is the Gaussian kernel from (2) and (3) describing the envelope of a Gabor prefilter for some σ_{g_j}. In practice, $P_i(u,v,\sigma_{g_j})$ can be calculated efficiently for all Gabor prefilter center frequencies (u,v) simultaneously using the form

$$P_i(u,v,\sigma_{g_j}) = \mathcal{F}\{\ (g_j(x,y) * g_j(x,y))\ R_i(x,y)\ \} \quad (11)$$

where $\mathcal{F}\{R_i(x,y)\} = S_i(u,v)$. A fast Fourier transform (FFT) is used to implement a discrete form of (11), giving $P_i(u,v,\sigma_{g_j})$ at a discrete set of center frequencies (u,v) for a particular σ_{g_j}.[9] The Rician parameters A_{ij}^2 and N_{ij} for a particular filter center frequency (u_j, v_j) are then given by[9]

$$A_{ij}^2 = A_i^2(u_j, v_j, \sigma_{g_j}) \quad (12)$$

$$N_{ij} = N_i(u_j, v_j, \sigma_{g_j}) \quad (13)$$

where

$$A_i^2(u,v,\sigma_{g_j}) \approx P_i(u,v,\sigma_{g_j}) - N_i(u,v,\sigma_{g_j}) \quad (14)$$

and

$$N_i(u,v,\sigma_{g_j}) \approx \frac{P_i(u,v,\sigma_{g_j}) - P_i(u,v,\sigma_{g_\beta})}{[1-(\frac{\sigma_{g_j}}{\sigma_{g_\beta}})^2]}\,. \quad (15)$$

Further details on Eqs. (14) and (15) can be found in Weldon and Higgins.[9] In practice, $A_i^2(u,v,\sigma_{g_j})$ and $N_i(u,v,\sigma_{g_j})$ are computed for each σ_{g_j} and each texture t_i under consideration using (14) and (15). The parameter σ_{g_β} is empirically chosen, with a typical value of $\sigma_{g_\beta} = 2\sigma_{g_j}$. In (12), A_{ij} tends to be larger in the vicinity of strong spectral peaks. In (13), the term N_{ij} tends to be larger in flat, or "noise-like," regions of the frequency spectrum. At all frequencies (u_j, v_j), the sum of N_{ij} and A_{ij}^2 is the total power of the sinusoid plus noise.

The means and variances of the prefilter outputs are found next, since these are used to find the means and variances of the postfilter outputs. The means $\mu_{g_{ij}}$ and variances $s^2_{g_{ij}}$ of the prefilter output $m_j(x,y)$ for each channel j and each texture t_i follow from the Rician density function:[28,46]

$$\mu_{g_{ij}} = \int_0^\infty m_j \; p_i(m_j;\; A_{ij}, N_{ij}) \; dm_j \tag{16}$$

$$s^2_{g_{ij}} = \int_0^\infty (m_j - \mu_{g_{ij}})^2 \; p_i(m_j;\; A_{ij}, N_{ij}) \; dm_j \; . \tag{17}$$

where, in practice, $\mu_{g_{ij}}$ and $s^2_{g_{ij}}$ are computed using numerical approximations of (16) and (17) at each discrete frequency (u,v), for each $\sigma_{g_{ij}}$ and texture t_i under consideration. (For notational convenience in (16) and (17), the argument (u,v,σ_{g_j}) is omitted in $\mu_{g_{ij}}$ and $s^2_{g_{ij}}$.)

The postfiltering generates a spatial average of the Rician-distributed prefilter output. Previously, we have shown that the statistics of the postfilter outputs $m_{p_{ij}}(x,y)$ are approximately Gaussian.[9,10] The means $\mu_{p_{ij}}$ and variances $s^2_{p_{ij}}$ of the postfilter outputs $m_{p_{ij}}(x,y)$ are derived from the prefilter means and variances and the parameters σ_{g_j} and σ_{p_j}:

$$\mu_{p_{ij}} = \mu_{g_{ij}} \tag{18}$$

$$s^2_{p_{ij}} = \frac{\sigma^2_{g_j}}{\sigma^2_{p_j}} s^2_{g_{ij}} \; . \tag{19}$$

where $\sigma^2_{g_j} < \sigma^2_{p_j}$.

The foregoing single-channel results must be extended to vector statistics for multiple channels. Since the output statistics of a single channel are Gaussian, we propose a multivariate Gaussian to describe the vector output statistics for k channels. First, define a matrix $\boldsymbol{\Theta_k}$ that determines the parameters for k channels:

$$\boldsymbol{\Theta_k} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_k \end{bmatrix} = \begin{bmatrix} u_1 & v_1 & \sigma_{g_1} & \sigma_{p_1} \\ u_2 & v_2 & \sigma_{g_2} & \sigma_{p_2} \\ \vdots & \vdots & \vdots & \vdots \\ u_k & v_k & \sigma_{g_k} & \sigma_{p_k} \end{bmatrix} , \tag{20}$$

where each row in $\boldsymbol{\Theta_k}$ defines the parameters $(u_j, v_j, \sigma_{g_j}, \sigma_{p_j})$ for a single channel. Then, the multivariate Gaussian density function describing the vector output $\boldsymbol{m_p}$ when texture t_i is filtered by a given set of channels $\boldsymbol{\Theta_k}$ is given by

$$p_i(\boldsymbol{m_p}, \boldsymbol{\Theta_k}) = \frac{1}{(2\pi)^{k/2} |\boldsymbol{C_i}|^{1/2}} \; e^{-\left(\frac{(\boldsymbol{m_p}-\boldsymbol{\mu_i})^T \boldsymbol{C_i}^{-1} (\boldsymbol{m_p}-\boldsymbol{\mu_i})}{2}\right)} \tag{21}$$

where

$$\boldsymbol{m_p} = \begin{bmatrix} m_{p_1} \\ m_{p_2} \\ \vdots \\ m_{p_k} \end{bmatrix} , \qquad \boldsymbol{\mu_i} = \begin{bmatrix} \mu_{p_{i1}} \\ \mu_{p_{i2}} \\ \vdots \\ \mu_{p_{ik}} \end{bmatrix} , \tag{22}$$

and

$$\boldsymbol{C_i} = \mathrm{E}\left[(\boldsymbol{m_p} - \boldsymbol{\mu_i})(\boldsymbol{m_p} - \boldsymbol{\mu_i})^T\right] . \tag{23}$$

The vector $\boldsymbol{m_p}$ is a sample of the k-dimensional postfilter-output vector, $m_{p_j} \in m_{p_j}(x,y)$ is a sample of the postfilter output in channel j for texture t_i, and $\boldsymbol{\mu_i}$ is the mean of the output vector for texture t_i. $\boldsymbol{C_i}$ is the $k \times k$ covariance matrix of the postfilter outputs for texture t_i, $\mathrm{E}[\,\cdot\,]$ indicates expected value, and superscript T indicates transpose. The argument $\boldsymbol{\Theta_k}$ is included in (21) to indicate that the output statistics depend on the filter parameters. This reflects the dependence of the mean vector $\boldsymbol{\mu_i}$ and covariance matrix $\boldsymbol{C_i}$ on the filter parameters.

The components of the mean vector $\boldsymbol{\mu_i}$ in (22) are determined for each texture t_i, $i = 1, 2, \ldots, \mathcal{N}$, and each set of candidate filter parameters $(u_j, v_j, \sigma_{g_j}, \sigma_{p_j})$ using (18). The covariance matrix $\boldsymbol{C_i}$ in (23) presents greater difficulty, and can not be practically computed for all filter combinations because of the inordinate number of combinations. However, the diagonal elements of the covariance matrix are already available as $s^2_{p_{ij}}$ from (19). Thus, we propose using the values of $s^2_{p_{ij}}$ along the diagonal of $\boldsymbol{C_i}$ with all off-diagonal elements equal to zero. This diagonal form is chosen for three reasons:

(1) because the computation of the off-diagonal elements is not practically feasible, (2) the diagonal form affords computational advantages in later stages of the development, and (3) the propensity for strongly correlated features (that would result in nonzero off-diagonal elements in $\boldsymbol{C_i}$) can be reduced by measures described shortly. The diagonal form is then

$$\boldsymbol{C_i} \approx \begin{bmatrix} s_{p_{i1}}^2 & 0 & \cdots & 0 \\ 0 & s_{p_{i2}}^2 & \ddots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & s_{p_{ik}}^2 \end{bmatrix} . \tag{24}$$

The form of (24) implies that the features are uncorrelated, since off-diagonal elements are all zero. To reduce the likelihood of strongly correlated features, we require any pair of Gabor prefilters in $\boldsymbol{\Theta_k}$ to be separated in frequency by more than the sum of their two bandwidths, as defined by $|G_j(u-u_j, v-v_j)| = e^{-1/2}$ in (2). The choice of $e^{-1/2}$ in defining bandwidth is somewhat empirical but has been found to produce effective results. The mirror image of filters reflected across the u and v frequency axes are also disqualified, since these filters have identical responses for real-valued $I(x,y)$. In the following section, the diagonal form for $\boldsymbol{C_i}$ also simplifies the calculation of the determinant and inverse as later needed in (25). Finally, our experimental results support the use of the diagonal form in (24), since both the design of the filters in the system and the design of the classifier are based on (24).

3.2 Segmentation Error Measure

The texture-segmentation error can now be estimated from the multivariate Gaussian distribution (21). In the following, an upper bound on classification error is used to obtain a measure of segmentation error, since a classifier is used in Fig. 1 to perform the segmentation. A bound is obtained using the Bhattacharyya distance. Then, a slight empirical modification, based on our experiments, is proposed.

The Bhattacharyya distance $B(t_\alpha, t_\beta, \boldsymbol{\Theta_k})$, or *B-distance*, between two texture classes, t_α and t_β, for a given set of filters $\boldsymbol{\Theta_k}$ is

$$\begin{aligned} B(t_\alpha, t_\beta, \boldsymbol{\Theta_k}) &= \frac{1}{8}(\boldsymbol{\mu_\alpha} - \boldsymbol{\mu_\beta})^T \left[\frac{\boldsymbol{C_\alpha} + \boldsymbol{C_\beta}}{2}\right]^{-1} (\boldsymbol{\mu_\alpha} - \boldsymbol{\mu_\beta}) \\ &+ \frac{1}{2}\, ln \left(\frac{\left|\frac{1}{2}\left(\boldsymbol{C_\alpha} + \boldsymbol{C_\beta}\right)\right|}{|\boldsymbol{C_\alpha}|^{1/2}|\boldsymbol{C_\beta}|^{1/2}} \right) \end{aligned} \tag{25}$$

where $\boldsymbol{\mu_\alpha}$ and $\boldsymbol{\mu_\beta}$ are the mean vectors, and $\boldsymbol{C_\alpha}$ and $\boldsymbol{C_\beta}$ are the covariance matrices associated with the two textures.[47] The B-distance can be used to find an upper bound for the classification error with $\mathcal{N}$ multivariate Gaussian classes representing the $\mathcal{N}$ textures. Let $\mathcal{E}_c(\boldsymbol{\Theta_k})$ be the total classification error for all $\mathcal{N}$ textures given filter parameters $\boldsymbol{\Theta_k}$. The error bound for $\mathcal{N}$ textures is then[48,49]

$$\mathcal{E}_c(\boldsymbol{\Theta_k}) < \sum_{\alpha=1}^{\mathcal{N}-1} \sum_{\beta=\alpha+1}^{\mathcal{N}} (\mathcal{P}_\alpha \mathcal{P}_\beta)^{1/2} \rho_{\alpha\beta} \tag{26}$$

where $\mathcal{P}_i$ is the *a priori* probability of texture t_i occurring in the image, and the two-class Bhattacharyya coefficients $\rho_{\alpha\beta}$ are

$$\rho_{\alpha\beta} = e^{-B(t_\alpha, t_\beta, \boldsymbol{\Theta_k})} .$$

We indicate the error's dependence on the filter parameters by including the argument $\boldsymbol{\Theta_k}$ in (26). Equations (25) and (26) provide the relationship between the image-segmentation error and the multivariate Gaussian statistics of the vector output of the k channels. The predicted error is taken to be the bound given in (26). In the absence of additional information, the *a priori* probabilities $\mathcal{P}_i$, $i = 1, 2, \ldots, \mathcal{N}$, are taken to be equal.

In practice, the error bound (26) is effective for multichannel filter design when the number of textures is small ($\mathcal{N} < 4$). We have observed that predicted segmentation error using (26) can greatly exceed 1 as the number of textures increases, and these large predicted errors cause difficulty when designing filters. The worst-case error in (26) occurs when all textures are identically distributed; i.e., when $\boldsymbol{\mu_\alpha} = \boldsymbol{\mu_\beta}$ and $\boldsymbol{C_\alpha} = \boldsymbol{C_\beta}$, $\forall\ \alpha, \beta$. In this case, $\rho_{\alpha\beta} = 1$, $\forall\ \alpha, \beta$ in (26); with equal *a priori* probabilities (26) then becomes:

$$\mathcal{E}_c(\boldsymbol{\Theta_k}) < \frac{1}{\mathcal{N}} \binom{\mathcal{N}}{2} = \frac{\mathcal{N}-1}{2}. \tag{27}$$

Thus, the upper bound of the classification error $\mathcal{E}_c(\mathbf{\Theta_k})$ can greatly exceed one for large numbers of textures. To prevent the error estimate from exceeding 1 and to improve the error estimate performance over a wider range of $\mathcal{N}$, we have found the following modification to (26) useful for estimating the classification error:

$$\mathcal{E}_c(\mathbf{\Theta_k}) \approx \frac{1}{\mathcal{N}-1}\sum_{\alpha=1}^{\mathcal{N}-1}\sum_{\beta=\alpha+1}^{\mathcal{N}}(\mathcal{P}_\alpha\mathcal{P}_\beta)^{1/2}\rho_{\alpha\beta} \quad (28)$$

where the worst-case upper bound on $\mathcal{E}_c(\mathbf{\Theta_k})$ in (28) becomes $\frac{1}{2}$. Alternatively, (28) could be scaled for a worst-case upper bound of $1-\frac{1}{\mathcal{N}}$.

3.3 Total Error Measure

Segmentation of textured images requires not only the accurate classification of textures within regions, but also the accurate localization of boundaries between regions. We refer to the former as *classification error*, and the latter as *localization error*. Thus, to form a total measure of image-segmentation error, we augment the classification error $\mathcal{E}_c(\mathbf{\Theta_k})$ with a localization error term.

In considering localization error, it is useful to consider two types of error that arise at texture boundaries: *edge error* and *corner error*. Edge error is loosely defined as the error in determining the boundary between two textures when the boundary is a straight line and no corners or other discontinuities occur in the boundary's vicinity. Corner error is loosely defined as the error in determining the boundary between two textures in the vicinity of a right-angle texture boundary. For the present discussion, this "vicinity" roughly corresponds to the spatial extent of the filter-channel impulse response as defined by parameters σ_{g_j} and σ_{p_j}.

In our approach, corner error is addressed by augmenting $\mathcal{E}_c(\mathbf{\Theta_k})$ to account for localization error. Edge error is addressed by slightly modifying the decision surfaces of the classifier using a mixture density.

The corner error tends to increase as the spatial resolution of the filters becomes coarse, since small features such as sharp corners are blurred by the filter. Based on a simple estimate of error in such corners, we propose the following empirical measure of this localization error $\mathcal{E}_l(\mathbf{\Theta_k})$:[50]

$$\mathcal{E}_l(\mathbf{\Theta_k}) \approx \frac{\mathcal{N}}{k}\sum_{j=1}^{k}\frac{2(\sigma_{p_j}^2+\sigma_{g_j}^2)}{N^2}\ , \quad (29)$$

where the image dimensions are $N\times N$, $\mathcal{N}$ is the number of textures, and the term $(\sigma_{p_j}^2+\sigma_{g_j}^2)$ approximates the effective spatial localization of the combined prefilter and postfilter in channel j. The summation over k generates an average over the k channels. Although the localization error measure $\mathcal{E}_l(\mathbf{\Theta_k})$ is straightforward, we find it to produce effective filter designs in our experiments.

The total error measure $\mathcal{E}_t(\mathbf{\Theta_k})$ for filter selection is then the sum of the classification error $\mathcal{E}_c(\mathbf{\Theta_k})$ and the localization error $\mathcal{E}_l(\mathbf{\Theta_k})$:

$$\begin{aligned}\mathcal{E}_t(\mathbf{\Theta_k}) &= \mathcal{E}_c(\mathbf{\Theta_k})+\mathcal{E}_l(\mathbf{\Theta_k})\\ &\approx \frac{1}{\mathcal{N}-1}\sum_{\alpha=1}^{\mathcal{N}-1}\sum_{\beta=\alpha+1}^{\mathcal{N}}(\mathcal{P}_\alpha\mathcal{P}_\beta)^{1/2}\ e^{-B(t_\alpha,t_\beta,\mathbf{\Theta_k})}\\ &+ \frac{1}{k}\sum_{j=1}^{k}\frac{2(\mathcal{N})(\sigma_{g_j}^2+\sigma_{p_j}^2)}{N^2} \quad (30)\end{aligned}$$

where σ_{g_j} and σ_{p_j} are defined by $\mathbf{\Theta_k}$ in (20). In (30), the first term represents error due to separability of classes in feature space, and tends to favor filters with larger values of σ_g and σ_p. The second term represents error due to localization, and tends to favor filters with smaller values of σ_g and σ_p.

At several points in the development, we have employed relatively simple models to avoid increased computation. Each of these models could be replaced by more sophisticated models at the expense of greater complexity. However, our proposed models mitigate the prohibitive number of filter combinations that must be considered in the design of a multichannel system. In particular, the filtered texture was modeled as a dominant sinusoid plus noise, leading to the Rician density function in (9) characterized by only two parameters A_{ij} and N_{ij}. Second, an efficient estimate of A_{ij} and N_{ij} was implemented using the FFT to compute $P_i(u,v,\sigma_{g_j})$ for use in (14) and (15). Next, a multivariate Gaussian model $p_i(\boldsymbol{m_p},\mathbf{\Theta_k})$ for the statistics of the vector output led to the use of the Bhattacharyya distance in estimating segmentation error. The diagonal covariance matrix $\boldsymbol{C_i}$ simplifies computation and does not require computing the covariance of all filtered textures for all filter combinations. Finally, a simple empirical estimate of

localization error $\mathcal{E}_l(\boldsymbol{\Theta_k})$ is used to augment classification error $\mathcal{E}_c(\boldsymbol{\Theta_k})$. Any added complexity in the models would be compounded by the number of filter combinations that must be considered when the filter design procedure is implemented. Further details on the computational complexity of our approach are found in Weldon and Higgins.[9]

3.4 Filter-Design Method

Combining the foregoing results, the procedure for designing k channels comprised of k Gabor prefilters and k Gaussian postfilters is given below.

1. Construct a collection Ψ of individual candidate channels:

$$\Psi = \{\ \theta\ \} = \{\ (u, v, \sigma_g, \sigma_p)\ \} \tag{31}$$

such that:

$$\begin{aligned} \sigma_g &\in \Sigma \\ \sigma_p &\in \{\lambda\,\sigma_g \mid \lambda \in \Lambda\} \\ (u,v) &\in \left\{ \left(\frac{\eta_1}{\sqrt{8\pi^2\sigma_g^2}}, \frac{\eta_2}{\sqrt{8\pi^2\sigma_g^2}} \right) \right\} , \end{aligned}$$

where

$$\begin{gathered} -0.5 \le u < 0.5\ ,\ 0 \le v < 0.5\ , \\ \eta_1 \in \{\ldots, -1, 0, 1, 2, \ldots\}\ ,\ \eta_2 \in \{0, 1, 2, \ldots\}\ , \end{gathered}$$

Σ is a set of candidate prefilter σ_g's, and Λ is a set of constants determining candidate ratios of the postfilter σ_p relative to each value of the prefilter parameter σ_g. For each value of σ_g, an overlapping tessellation of candidate Gabor prefilters is created in the frequency half-plane. Ψ is a set of candidate filters from which the final design is constructed and differs from filter-bank approaches to texture segmentation where the filter parameters are predetermined and subdivide the frequency plane into non-overlapping regions. The filters in Ψ overlap significantly within the tesselation for each value of σ_g, and filters within a tesselation for one value of σ_g can completely overlap filters within a tesselation for another value of σ_g. Thus, our approach permits freedom in center-frequency selection due to the overlapping tesselation and permits different filter bandwidths at a center frequency by including separate tesselations at different values of σ_g. Finally, we note that candidate filters in Ψ are not limited to octave scalings of σ_g.

The candidate prefilter center frequencies (u, v) in (31) cover the closed right-half frequency plane at integer multiples of $1/\sqrt{8\pi^2\sigma_g^2}$, where $1/\sqrt{8\pi^2\sigma_g^2}$ is the displacement from center frequency at which the Gabor prefilter frequency response (2) equals $e^{-1/4} = 0.78$. This frequency separation allows significant overlap between adjacent filters, without unduly increasing the number of filters. Although filters with overlapping responses are members of Ψ, any combination of filters comprising $\boldsymbol{\Theta_k}$ is not permitted to have overlapping prefilters at the $e^{-1/2}$ frequency-response point. Overlapping responses are not permitted in an effort to reduce the likelihood of strongly correlated channel outputs. In addition, center frequencies within $1/\sqrt{8\pi^2\sigma_g^2}$ of the origin $(u, v) = (0, 0)$ are excluded to prevent inadvertent segmentation results from average-response differences.[1,2,50]

2. Compute $A_i^2(u, v, \sigma_g)$ and $N_i(u, v, \sigma_g)$ for each $\sigma_g \in \Sigma$ and each texture t_i using (14) and (15). This is done for each sample texture t_i under consideration. $P_i(u, v, \sigma_g)$ in (10) can be computed efficiently using a Fast Fourier Transform.[9,50] The result of this step is the parameters $A_i^2(u, v, \sigma_g)$ and $N_i(u, v, \sigma_g)$ at each discrete frequency (u, v) for each $\sigma_g \in \Sigma$ and for each texture t_i, $i = 1, 2, \ldots, \mathcal{N}$.

3. Find the best single filter-channel $\theta_1 \in \Psi$ as defined by

$$\mathcal{E}_t(\theta_1) \le \mathcal{E}_t(\theta_\xi)\ , \qquad \forall\ \theta_\xi \in \Psi \tag{32}$$

where $\theta_1 \in \Psi$. The predicted segmentation error associated with each filter is computed using (30). The components of $\boldsymbol{\mu_i}$ and $\boldsymbol{C_i}$ are computed from $A_i^2(u, v, \sigma_g)$ and $N_i(u, v, \sigma_g)$ using using (18) and (19) for each set of filter parameters $\theta \in \Psi$. Off-diagonal elements of $\boldsymbol{C_i}$ are set equal to zero. For this step, (30) takes a scalar form for

a single channel; i.e., the multivariate Gaussian in (21) becomes univariate.

4. Search for subsequent filters using a forward sequential selection method,[49]

$$\mathbf{\Theta}_\delta = \begin{bmatrix} \mathbf{\Theta}_{\delta-1} \\ \theta_\delta \end{bmatrix} \tag{33}$$

such that

$$\mathcal{E}_t\left(\begin{bmatrix} \mathbf{\Theta}_{\delta-1} \\ \theta_\delta \end{bmatrix}\right) \leq \mathcal{E}_t\left(\begin{bmatrix} \mathbf{\Theta}_{\delta-1} \\ \theta_\xi \end{bmatrix}\right), \quad \forall\, \theta_\xi \in \Psi$$

where row vector $\theta_\delta \in \Psi$, and where $\mathbf{\Theta}_{\delta-1}$ is a fixed $\delta - 1$ row by 4 column matrix, having been established at step $\delta - 1$ of the procedure. The forward-sequential method terminates when a desired error level $\mathcal{E}_t(\mathbf{\Theta_\delta})$ is achieved or when a desired number of channels k is reached. The final filter design is then denoted $\mathbf{\Theta_k}$. Although a single channel θ_δ is added at each step in (33), the criteria for selecting the added channel is based on the combined effect of all δ channels through the vector error measure $\mathcal{E}_t(\mathbf{\Theta_\delta})$. Thus, the vector output statistics of all δ channels are considered as each new channel is added. The final filter design is then $\mathbf{\Theta_k}$. Alternatively, the procedure can terminate when the predicted segmentation error $\mathcal{E}_t(\mathbf{\Theta_\delta})$ reaches some desired level.

4 Results

Before proceeding to the experimental results, we briefly outline the methods used to design the overall texture-segmentation system in Fig. 1. First, the total segmentation-error measure $\mathcal{E}_t(\mathbf{\Theta_k})$ in (30) is used as the criterion for the design of the channels. Given a set of texture samples, the filter parameters are chosen to minimize total segmentation error $\mathcal{E}_t(\mathbf{\Theta_k})$ using the filter design procedure outlined in Section 3.4. Second, the predicted multivariate Gaussian probability density $p_i(\boldsymbol{m_p}, \mathbf{\Theta_k})$ of (21) is used to construct a Bayesian classifier for the vector classifier in Fig. 1. Predicted statistics are used rather than directly-measured statistics to confirm the effectiveness of the predicted statistics in (21). Finally, postprocessing is used to remove narrow misclassified regions at texture boundaries.

We have tested our texture-segmentation method on a wide range of synthetic and Brodatz textures.[51] In Fig. 2, we show results for a 256×256 pixel 8-bit gray-scale image comprised of 8 textures. To prevent biased segmentation results due to leakage of the DC component through the Gabor prefilters, the average gray-scale of all textures were equalized.[1,2,50] The input image of Fig. 2(a) contains eight Brodatz textures, counterclockwise from the upper left: d77 "cotton canvas," d84 "raffia," d55 "straw matting," d17 "herringbone weave," d68 "wood grain," d57 "handmade paper," d21 "French canvas," d24 "pressed leather".[51] The parameters $\Sigma = \{2, 3, 4.5\}$ and $\Lambda = \{1.5\}$ were used to construct the collection of candidate filters Ψ. The segmented images in Figs. 2(b), (c), and (d) show progressive improvement as the number of channels increases from 2, to 4, to 6. Even with two filters, we find that our approach can generate an effective design for segmenting an eight-texture image. Although such effective results are not always achievable with so few filters and so many textures, they do illustrate the potential of our methods.

Fig. 2(e) is a plot of the predicted single-channel error as a function of frequency. The gray level in this plot is proportional to segmentation error, with black representing zero error and white representing 100 % error. The center of Fig. 2(e) corresponds to a prefilter center frequency $(u, v) = (0, 0)$. The u axis ranges from -.5 at the top to +.5 at the bottom, and the v axis ranges from -.5 at the left to +.5 cycles-per-pixel at the right of the image. The locations of the center frequencies of the 6 filters used in the segmentation of Fig. 2(d) are shown as white squares superimposed on Fig. 2(e). The center frequencies of the six filters in Fig. 2(e) tend to fall within darker regions corresponding to lower predicted error for a single filter. It is unlikely that any single filter could segment all eight textures with little error, and this is manifested in Fig. 2(e) by the absence of any dark black regions that would suggest a very low predicted segmentation error for a single filter. Finally, in Fig. 2(f) the misclassified pixels from the 6-channel segmentation of Fig. 2(d) are shown in black.

As another example, the image shown in Fig. 3(a) is composed of five samples from the Brodatz texture album and resembles the "Nat-5" image used by previous investigators to test texture-segmentation methods.[3,42,51] The parameters $\Sigma = \{3, 6, 12\}$ and $\Lambda = \{1.5, 2\}$ were used to construct the collection of candidate filters Ψ. For this example, there are ap-

proximately 7000 different prefilters in Ψ, each with two possible postfilters, resulting in approximately 14,000 candidate filter channels. There are then approximately $(14,000)^4 \approx 4 \times 10^{16}$ combinations of 4 filters. Fig. 3(b) is the result of a $k = 2$ channel segmentation, (c) is the result of a $k = 4$ channel segmentation. The measured segmentation error decreases from 0.13 to 0.05 as the number of channels increases. By comparison, Jain and Farrokhnia obtained similar segmentation results using 13 filters selected from a predetermined filter bank of 20 filters.[3] Randen and Husøy also achieved similar results using 13 to 40 filters.[42]

5 Discussion

We have presented a mathematical framework for the design of multiple Gabor filters to segment multiple textures. The framework provides relationships between predicted segmentation error $\mathcal{E}_t(\boldsymbol{\Theta_k})$, Gabor prefilter parameters (u, v, σ_{g_j}), Gaussian postfilter parameters σ_{p_j}, and sample-texture power spectra $S_i(u, v)$. In addition, the framework gives predicted multivariate output statistics $p_i(\boldsymbol{m_p}, \boldsymbol{\Theta_k})$ for multiple channels.

A multi-channel filter design procedure was developed using the mathematical framework. The experimental results support the framework, since both the designed filters and the Bayesian classifier used in the segmentations were based on it. The results showed effective segmentation of 8 textures in a single image using 2-6 channels. Compared to earlier efforts, the results suggest that our methods can generate effective segmentations using fewer filters.

In our framework, the bandpass-filtered texture was modeled as a dominant sinusoid with remaining energy allocated to noise. Although imperfect, this model has been found to be remarkably effective in our experiments. The generality of this notion also suggests that our methods should be applicable to other types of bandpass filters. Such an adaptation can be accomplished by replacing the kernel $G_j(u, v)$ in (10) by a different kernel $K(u, v)$. ($K(u, v)$ could also accommodate asymmetric filter variants.) For a direct substitution, $K(u, v)$ should satisfy $[K(u, v)]^2 = |K(u, v)|^2$, and $K(u, v)$ should be non-negative to assure that the spectral estimate given by (10) is non-negative.

Since our classifier is designed using the sample textures, it will assign all regions of the textured image to one of the original texture classes. Thus, a texture different from the original textures used in the design of the system will be assigned to one of the original textures. However, the absence of a texture from the image will not present difficulties since the feature vector at the filter-bank output should not be found in regions of feature space assigned to this texture class. In our experimental results, the successful segmentation of the large homogeneously textured regions is representative of this second situation.

The multiple filter design problem is deceptively complex. To illustrate this, consider the problem of designing k filters to segment an $N \times N$ pixel image. If there are $N^2/2$ candidate-filter center frequencies, $(log_2 N)/2$ filter bandwidths, and a set of only $k = 4$ channels, then there are approximately $[(N^2\, log_2 N)/4]^k$ possible filter-channel combinations, or 3×10^{20} combinations for $N = 256$. Therefore, a brute-force approach without judicious selection of candidate filters and careful attention to computational efficiencies quickly becomes prohibitive. Similarly, methods requiring explicit filtering of textures or multivariate statistical characterization of the outputs would only exacerbate the problem. For $\mathcal{N}$ textures and $(log_2 N)/2$ filter bandwidths, the computational complexity of our method is roughly $3\mathcal{N}[(log_2 N)/2](N^2 log_2 N) = 1.5\mathcal{N}(N log_2 N)^2$, due primarily to three $N \times N$ FFT's used in the computation of the mean and covariance statistics for each texture and each bandwidth.[9] This gives a complexity of $\approx 3 \times 10^7$ for $\mathcal{N} = 5$ textures and 256×256 pixel images.

Finally, the images and a demonstration software package called "teXan" are available.[52] The package is a graphic-interface-based program for interactive texture segmentation. Given an image, the user marks samples of the different textures to be segmented. The program then uses these samples to design the filters, design the classifier, and segment the image automatically using the techniques presented in this paper.

Acknowledgments

This work was partially supported by NIH FIRST award #CA53607 from the National Cancer Institute of the National Institutes of Health. Preliminary versions of portions of this work have appeared at conferences:[27,28,53]

References

1. A. C. Bovik, M. Clark, and W. S. Geisler, "Multichannel texture analysis using localized spatial

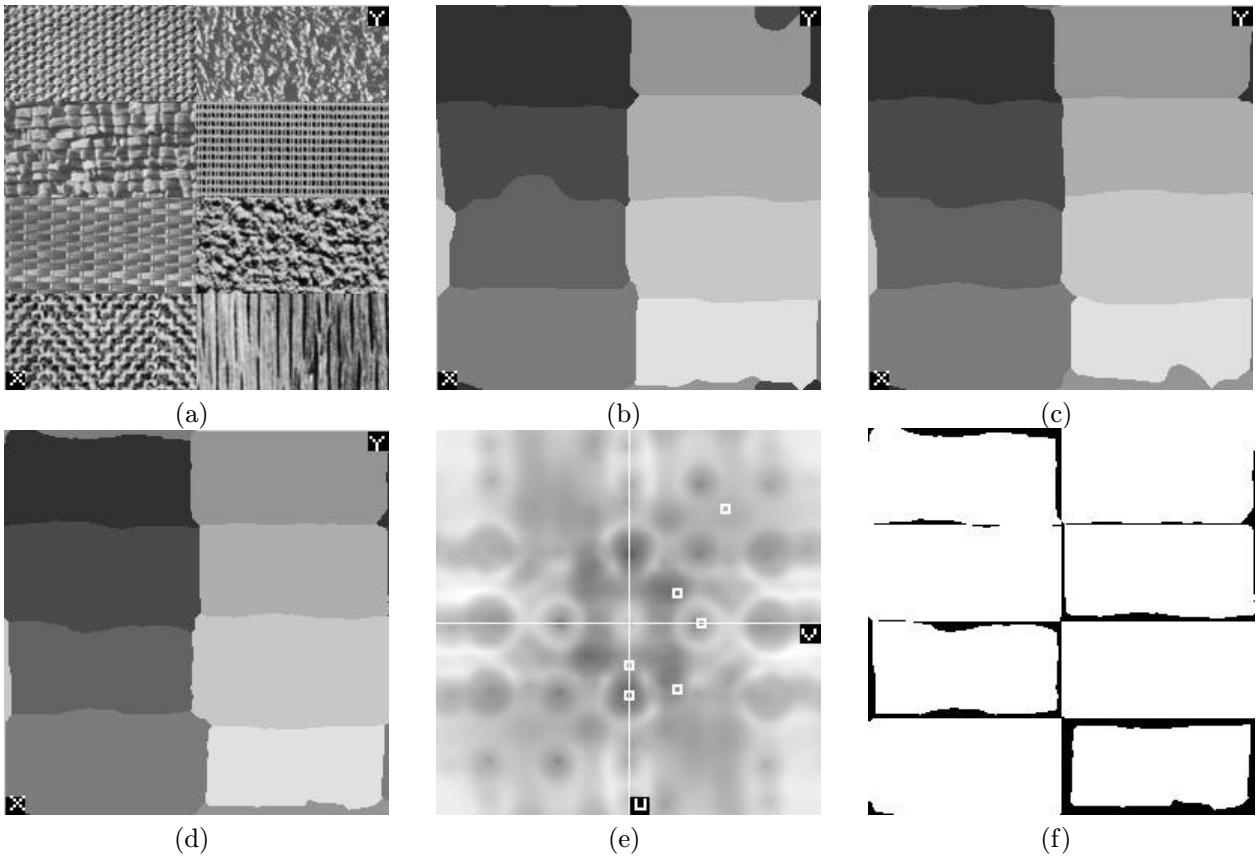

Figure 2: Segmentation results for $\Sigma = \{2, 3, 4.5\}$, $\Lambda = \{1.5\}$. (a) Input composite 256×256 image, d77, d84, d55, d17, d68, d57, d21, d24. (b) 2-channel segmentation, measured segmentation error = 0.09. (c) 4-channel segmentation, measured segmentation error = 0.08. (d) 6-channel segmentation, measured segmentation error = 0.08. (e) Filter center frequencies (white squares) plotted on error map. (f) Segmentation error for six-channel segmentation, misclassified pixels in black. The parameters for the six filters are $\mathbf{\Theta_6} = [\ (0.19, 0, 4.5, 6.8),\ (0.17, 0.13, 4.5, 6.8),\ (-0.08, 0.13, 4.5, 6.8),\ (0.11, 0, 4.5, 6.8),\ (-0.30, 0.25, 2, 3),\ (0, 0.19, 3, 4.5)\ ]$

filters," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, no. 1, pp. 55–73, Jan. 1990.

2. A. C. Bovik, "Analysis of multichannel narrowband filters for image texture segmentation," *IEEE Trans. Signal Processing*, vol. 39, no. 9, pp. 2025–2043, Sept. 1991.

3. A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using Gabor filters," *Pattern Recognition*, vol. 23, no. 12, pp. 1167–1186, Dec. 1991.

4. A. C. Bovik, N. Gopal, T. Emmoth, and A. Restrepo, "Localized measurements of emergent image frequencies by Gabor wavelets," *IEEE Trans. Inform. Theory*, vol. 38, no. 2, pp. 691–711, Mar. 1992.

5. M. Unser, "Texture classification and segmentation using wavelet frames," *IEEE Trans. Image Proc.*, vol. 4, no. 11, pp. 1549–1560, Nov. 1995.

6. D. F. Dunn and W. E. Higgins, "Optimal Gabor filters for texture segmentation," *IEEE Trans. Image Proc.*, vol. 4, no. 7, pp. 947–964, July 1995.

7. D. Dunn, W. Higgins, and J. Wakeley, "Texture segmentation using 2-D Gabor elementary functions," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 16, no. 2, pp. 130–149, Feb. 1994.

8. J. Bigun and J. M. H. du Buf, "N-folded symmetries by complex moments in Gabor space and their application to unsupervised texture segmentation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 16, no. 1, pp. 80–87, Jan. 1994.

(a) (b) (c)

Figure 3: Results for $\Sigma = \{3, 6, 12\}$, $\Lambda = \{1.5, 2\}$. (a) Input composite image "Nat-5," clockwise from top left: d77 "cotton canvas," d55 "straw matting," d17 "herringbone weave," d84 "raffia," d24 "pressed calf leather" in center. (b) 2-channel segmentation, measured error=0.13. (c) 4-channel segmentation, measured error=0.05. The parameters for the four filters are $\mathbf{\Theta_4} = [\ (-0.078, 0.11, 6, 12),\ (0.13, 0.02, 6, 12),\ (0, 0.19, 3, 6),\ (0.2, 0, 3, 6)\ ]$

9. T. P. Weldon, W. E. Higgins, and D. F. Dunn, "Gabor filter design for multiple texture segmentation," *Optical Eng.*, vol. 35, no. 10, pp. 2852–2863, Oct. 1996.

10. T. P. Weldon, W. E. Higgins, and D. F. Dunn, "Efficient Gabor filter design for texture segmentation," *Pattern Recognition*, vol. 29, no. 12, pp. 2005–2015, Dec. 1996.

11. T. N. Tan, "Texture edge detection by modelling visual cortical channels," *Pattern Recognition*, vol. 28, no. 9, pp. 1283–1298, Sept. 1995.

12. C. S. Lu, P. C. Chung, and C. F. Chen, "Unsupervised texture segmentation via wavelet transform," *Pattern Rec.*, vol. 30, no. 5, pp. 729–742, May 1992.

13. B. S. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of data," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, no. 8, pp. 837–842, Aug. 1996.

14. A. K. Jain and S. Bhattacharjee, "Text segmentation using Gabor filters for automatic document processing," *Machine Vision and Applications*, vol. 5, pp. 169–184, 1992.

15. D. F. Dunn, T. P. Weldon, and W. E. Higgins, "Extracting halftones from printed documents using texture analysis," *Optical Eng.*, vol. 36, no. 4, pp. 1044–1052, Apr. 1997.

16. J. G. Daugman, "Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, no. 7, pp. 1169–1179, July 1988.

17. T. Ebrahimi and M. Kunt, "Image compression by Gabor expansion," *Optical Eng.*, vol. 30, no. 7, pp. 873–880, July 1991.

18. J. G. Daugman, "High confidence visual recognition of persons by a test of statistical independence," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, no. 11, pp. 1148–1160, Nov. 1993.

19. A. Shustorovich, "A subspace projection approach to feature extraction: The two-Dimensional Gabor transform for character recognition," *Neural Networks*, vol. 7, no. 8, no. 8, pp. 1295–1301, 1994.

20. D. P. Casasent and J. S. Smokelin, "Neural net design of macro Gabor wavelet filters for distortion-invariant object detection in clutter," *Optical Eng.*, vol. 33, no. 7, pp. 2264–2271, July 1994.

21. D. P. Casasent and A. Ye, "Detection filters and algorithm fusion for ATR," *IEEE Trans. Image Proc.*, vol. 6, no. 1, pp. 114–125, Jan. 1997.

22. B. J. Super and A. C. Bovik, "Localized measurement of image fractal dimension using Gabor filters," *Journal of Visual Comm. and Image Representation*, vol. 2, no. 2, pp. 114–128, June 1991.

23. R. Mehrotra, K. R. Namuduri, and N. Ranganathan, "Gabor filter-based edge detection," *Pattern Recognition*, vol. 25, no. 12, pp. 1479–1493, Dec. 1992.

24. R. Buse, Z. Liu, and T. Caelli, "Using Gabor filters to measure the physical parameters of lines," *Pattern Recognition*, vol. 29, no. 4, pp. 615–625, Apr. 1996.

25. M. P. Anderson, M. H. Loew, and D. G. Brown, "Gabor function-based medical image compression," *Image and Vision Computing*, vol. 13, no. 7, pp. 535–541, Sept. 1995.

26. M. Porat and Y. Zeevi, "The generalized Gabor scheme of image representation in biological and machine vision," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 10, no. 4, pp. 452–468, July 1988.

27. T. Weldon and W. Higgins, "Design of multiple Gabor filters for texture segmentation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. IV, (Atlanta, GA), pp. 2245–2248, 7-10 May 1996.

28. T. P. Weldon and W. E. Higgins, "Integrated approach to texture segmentation using multiple Gabor filters," in *Proc. IEEE Int. Conf. on Image Processing*, vol. III, (Lausanne, Switzerland), pp. 955–958, 16-19 Sept. 1996.

29. T. Randen and J. H. Husøy, "Novel approaches to multichannel filtering for image texture segmentation," in *Proc. SPIE Visual Comm. Image Processing 1994*, vol. 2094, pp. 626–636, 1994.

30. P. P. Raghu and B. Yegnanarayana, "Segmentation of Gabor-Filtered textures using deterministic relaxation," *IEEE Trans. Image Proc.*, vol. 5, no. 12, pp. 1625–1636, Dec. 1996.

31. X. Wu and B. Bhanu, "Gabor wavelet representation for 3-D object recognition," *IEEE Trans. Image Proc.*, vol. 6, no. 1, pp. 47–64, Jan. 1997.

32. B. S. Manjunath and R. Chellapa, "A unified approach to boundary perception: Edges, textures, and illusory contours," *IEEE Trans. Neural Networks*, vol. 4, no. 1, pp. 96–108, Jan. 1993.

33. H. W. Tang, V. Srinivasan, and S. H. Ong, "Texture segmentation via nonlinear interactions among Gabor feature pairs," *Optical Eng.*, vol. 34, no. 1, pp. 125–134, Jan. 1995.

34. M. R. Turner, "Texture discrimination by Gabor functions," *Biological Cybernetics*, vol. 55, pp. 71–82, 1986.

35. J. Malik and P. Perona, "Preattentive texture discrimination with early vision mechanisms," *J. Opt. Soc. Amer. A*, vol. 7, no. 5, pp. 923–932, May 1990.

36. J. M. H. du Buf, "Abstract processes in texture discrimination," *Spatial Vision*, vol. 6, pp. 221–242, 1992.

37. T. Chang and C. C. J. Kuo, "Texture analysis and classification with tree-structured wavelet transform," *IEEE Trans. Image Proc.*, vol. 2, no. 4, pp. 429–441, Oct. 1993.

38. R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. John Wiley and Sons, 1973.

39. A. Teuner, O. Pichler, and B. H. Hosticka, "Unsupervised texture segmentation of images using tuned matched Gabor filters," *IEEE Trans. Image Proc.*, vol. 4, no. 6, pp. 863–870, June 1995.

40. D. Casasent and J. Smokelin, "Wavelet and Gabor transforms for detection," in *Proc. SPIE Wavelet Applications 1994*, vol. 2242, pp. 678–689, 1994.

41. T. Caelli and D. Carevic, "Application of partial modelling techniques for texture segmentation," *J. Opt. Soc. Am. A*, vol. 14, no. 11, pp. 2924–2937, Nov. 1997.

42. T. Randen and J. H. Husøy, "Multichannel filtering for image texture segmentation," *Optical Eng.*, vol. 33, no. 8, pp. 2617–2625, Aug. 1994.

43. S. Rice, "Mathematical analysis of random noise," *Bell System Tech. Journal*, vol. 23, pp. 282–333, July 1944.

44. S. Rice, "Mathematical analysis of random noise," *Bell System Tech. Journal*, vol. 24, pp. 96–157, Jan. 1945.

45. M. Schwartz, *Information Transmission, Modulation, and Noise*. New York, NY: McGraw-Hill, third ed., 1980.

46. T. P. Weldon, W. E. Higgins, and D. F. Dunn, "Gabor filter design for multiple texture segmentation," *Optical Eng.*, vol. 35, no. 10, pp. 2852–2863, Oct. 1996.

47. W. K. Pratt, *Digital Image Processing*. John Wiley and Sons, second ed., 1991.

48. D. Lainiotis, "A class of upper bounds on probability of error for multihypotheses pattern recognition," *IEEE Trans. Inform. Theory*, vol. 15, pp. 730–731, Nov. 1969.

49. L. Kanal, "Patterns in pattern recognition: 1968-1974," *IEEE Trans. Inform. Theory*, vol. 20, no. 6, pp. 697–722, Nov. 1974.

50. T. P. Weldon, *Multiresolution Design of Multiple Gabor Filters for Texture Segmentation*. PhD thesis, The Pennsylvania State University, 1995.

51. P. Brodatz, *Textures: A Photographic Album for Artists and Designers*. New York, NY: Dover, 1966.

52. T. P. Weldon, "The teXan software package." Available: http://ws351.uncc.edu/tpw/texan/ or http://cobb.ee.psu.edu/img_proc_lab.html.

53. T. P. Weldon and W. E. Higgins, "An algorithm for designing multiple Gabor filters for segmenting multi-textured images," in *Proc. IEEE Int. Conf. on Image Processing*, (Chicago, IL), 4-7 Oct. 1998.

54. I. Bloch, "Fuzzy connectivity and mathematical morphology," *Pattern Recog. Lett.*, vol. 14, no. 6, p. 483, June 1993.

Thomas Weldon received the B.S., M.Eng., and Ph.D. in electrical engineering from Penn State in 1979, 1989, and 1995. From 1979 to 1982 he designed portable communications equipment at Motorola in Plantation, Florida. From 1982 to 1984 he designed microelectronic circuits at Alpha Industries in Lansdale, Pennsylvania. From 1984 to 1990 he developed signal processing and microwave communication systems as a senior engineer at American Electronic Laboratories in Lansdale, Pennsylvania. From 1990 to 1995 he was engaged in signal-processing and image-processing research at Penn State while completing his Ph.D. In 1995 he joined the University of North Carolina at Charlotte as an assistant professor of electrical engineering, where he is presently engaged in research on image processing and biotelemetry systems. There, he was awarded the 1995-96 Tau Beta Pi Outstanding Professor Award. His current research interests are image processing, signal processing, biomedical imaging, biotelemetry, and communication systems. He is a member of the SPIE, IEEE, and the Pattern Recognition Society.

William E. Higgins obtained the B.S. degree in electrical engineering (1979) at the Massachusetts Institute of Technology and the M.S. (1981) and Ph.D. (1984) degrees in electrical engineering at the University of Illinois, Urbana. Previously, he worked as a principal research scientist at the Honeywell Systems and Research Center (1984-1987), Minneapolis, MN, working in the field of automatic target-recognition systems. He later was a senior research fellow in the Biodynamics Research Unit of the Mayo Clinic (1987-1989), Rochester, MN, working in the areas of multidimensional image analysis and visualization. Since 1989, he has been at the Pennsylvania State University, University Park, PA, where he is currently a professor in the departments of electrical engineering, computer science and engineering, and bioengineering. He is also an adjunct professor of radiology at the University of Iowa. His research interests are in image processing, computer vision, scientific visualization, and medical image analysis. He is a member of the SPIE, IEEE, Sigma Xi, Tau Beta Pi, Eta Kappa Nu, and the Pattern Recognition Society.