



# Designing Gabor filters for optimal texture separability

David A. Clausi\*, M. Ed Jernigan

*Department of Systems Design Engineering, University of Waterloo, 200 University Avenue West, Waterloo, Ont., Canada N2L 3G1*

Received 24 April 1998; received in revised form 1 July 1999; accepted 9 August 1999

---

## Abstract

The discrimination of textures is a critical aspect of identification in digital imagery. Texture features generated by Gabor filters have been increasingly considered and applied to image analysis. Here, a comprehensive classification and segmentation comparison of different techniques used to produce texture features using Gabor filters is presented. These techniques are based on existing implementations as well as new, innovative methods. The functional characterization of the filters as well as feature extraction based on the raw filter outputs are both considered. Overall, using the Gabor filter magnitude response given a frequency bandwidth and spacing of one octave and orientation bandwidth and spacing of  $30^\circ$  augmented by a measure of the texture complexity generated preferred results. © 2000 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

*Keywords:* Gabor filter; Texture analysis; Feature extraction; Classification; Segmentation

---

## 1. Introduction

To attempt image texture segmentation, one must first determine preferred feature sets. “The development of computational formalisms for segmenting, discriminating, and recognizing image texture projected from visible surfaces are complex and interrelated problems. An important goal of any such formalism is the identification of easily computed and physically meaningful image features which can be used to effectively accomplish those tasks [1], p. 2025”. In this paper, various Gabor filter implementations for texture analysis are compared. Some of these techniques are implemented in existing publications and some are new methods for generating texture features based on Gabor filters. Usually, scientists use a particular Gabor filter configuration and feature extraction technique for texture analysis without (a) demonstrating appropriateness of the selected method and/or (b) comparing the selected method with other potential methods. In some cases, the implementation

used is inappropriate and the final results are suspect. In other cases, additional computations are performed which do not necessarily improve the quality of the texture features. The evaluation performed here provides guidelines when using Gabor filters for texture analysis in order to achieve preferred texture features.

Comparisons are performed using classification and unsupervised image segmentation. Classification is a method of choice for comparisons since calculations can be performed quickly (which provides the opportunity for a wide variety of tests) and quantitative error estimates can be determined. Features that perform better in a classification environment should have a stronger potential to perform better in an unsupervised image segmentation role.

To motivate the use of Gabor filters, Section 2 presents the relationship between the human visual system (HVS) and texture interpretation. Then, Section 3 considers how the Gabor filters can be configured to capture textural information. Descriptions of various techniques that can be used for feature extraction based on the raw Gabor filter outputs are considered in Section 4. These different scenarios are tested using both classification and segmentation (Sections 5 and 6). Discussions and conclusions follow the test results (Section 7).

---

\* Corresponding author. Tel.: + 519-885-1211; fax: + 519-746-4791.

*E-mail address:* [dclausi@engmail.uwaterloo.ca](mailto:dclausi@engmail.uwaterloo.ca) (D.A. Clausi).

## 2. Background

Multi-channel filtering is an effective consideration in the field of texture analysis. By processing the image using multiple resolution techniques, filter banks have the ability to decompose an image into relevant texture features that can be used to classify the textures accordingly. Multi-channel filtering mimics characteristics of the human visual system (HVS).

The HVS is extremely complex. Receptor cells on the back inner surface of the eyeball pass electrical impulses through various nervous pathways to the visual cortex. In this part of the brain, various cells perform different types of processing on the incoming signals. Hubel and Wiesel [2] deduced that simple cells are sensitive to specific orientations with approximate bandwidths of  $30^\circ$ . Campbell and Kulikowski [3] carried this concept further and demonstrated that humans have spatial frequency sensitivity as well as orientational sensitivity. This led to a model that the HVS is made up of “a number of independent detector mechanisms each preceded by a relatively narrow band filter ‘tuned’ to a different frequency. Each filter and detector would constitute a separate ‘channel’ ... [4], p. 564”. Experiments indicate that the frequency bandwidth of simple cells in the visual cortex is about one octave [5]. This HVS multi-channel filtering model agrees with a popular generic approach to signal decomposition, namely, wavelet analysis.

The research by Rao and Lohse [6] provides additional insight to this idea of orientation and frequency sensitivity but from a human perception slant. Subjects were asked to classify pictures from the Brodatz album [7] based solely on their own perception. Results indicate that people essentially use three high-level features for texture interpretation, namely: repetition, directionality, and complexity. Repetition and directionality are representative of frequency and orientation. Complexity relates to the consistency of the texture. For example, a purely sinusoidal texture would have low complexity and a texture without any well-defined pattern would have high complexity. These three characteristics should be considered when implementing machine vision texture interpretation algorithms.

For texture analysis, wavelet analysis has the ability to model the frequency and orientation sensitivities characteristic of the HVS. The scaling and translation properties of the wavelet transform make it an attractive tool for image processing applications. The number of oscillations the wavelet experiences is independent of the scaling and translation. The scaling parameter can be set so that the basic wavelet is compressed. In this case, more cycles occur in a shorter time frame, which indicates higher frequencies, and inherently the wavelet transform maps a shorter time interval to higher frequencies. Similarly, a longer time interval is mapped to lower frequencies. This inherent ability has tremendous potential when

segmenting natural imagery since multiple dominant frequencies are found in natural images and the wavelet filtering can take into account these multiple frequencies. Generally, the wavelet transform can take a signal, break it down into component pieces, the manipulation of which can yield features that represent characteristics of the various textures appearing in the image.

## 3. Gabor filter implementation for texture analysis

Research has demonstrated that the HVS generates a multi-resolutional decomposition. Since wavelets are intrinsically multi-resolutional, they have been utilized in texture analysis models. The Gabor function can be implemented as a multi-channel, wavelet-like filter. It has properties that make it attractive for computer vision applications. These properties include its appealing simplicity, optimum joint spatial/spatial-frequency localization, and ability to simulate the behavior of two-dimensional receptive fields of simple cells in the visual cortex (by isolating specific frequencies and orientations) [8].

When generating texture features using multi-channel filters, the following two steps should receive special consideration. First, the characterization of the filters (e.g. functionality, number, orientation, and spacing) must be selected carefully. Second, feature extraction of the raw filter outputs should be performed to improve the feature set. Here in Section 3 we discuss the Gabor filter characterization and in Section 4 feature extraction based on Gabor filter outputs is presented.

### 3.1. Gabor filter functionality

Spatially, a Gabor function is a Gaussian modulated sinusoid. The 2-d Gaussian has an aspect ratio of  $\sigma_x/\sigma_y$ . The complex exponential has a spatial frequency of  $F$  and an orientation  $\theta$  (counterclockwise with respect to the horizontal axis). A complex Gabor filter represented as a 2-d impulse response is

$$h(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left\{-\frac{1}{2}\left[\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right]\right\} \exp\{j2\pi Fx\}.$$

Ensuring that the  $x$ -axis of the Gaussian has the same orientation  $\theta$  as the frequency is a convenient implementation [9]. Rotation in the  $x$ - $y$  plane then provides for any arbitrary orientation of the filter. The corresponding representation in the spatial-frequency domain is

$$H(u, v) = \exp\{-2\pi^2[(u - F)^2\sigma_x^2 + v^2\sigma_y^2]\}.$$

As in the spatial domain, a rotation can be used to obtain any desired direction in the spatial-frequency  $u$ - $v$  plane.

The Gabor filter has also been implemented for texture analysis using only its real (even) component [10]. In this circumstance, the filter impulse response is

$$h(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left\{-\frac{1}{2}\left[\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right]\right\} \cos\{2\pi Fz\}.$$

In this case, the filter is represented in the spatial-frequency domain by two symmetrically located Gaussians:

$$H(u, v) = \exp\{-2\pi^2[(u - F)^2\sigma_x^2 + v^2\sigma_y^2]\} \\ + \exp\{-2\pi^2[(u + F)^2\sigma_x^2 + v^2\sigma_y^2]\}.$$

Using either the real or complex versions, there are six parameters that must be set when implementing these Gabor filters:  $F$ ,  $\theta$ ,  $\sigma_x$ ,  $\sigma_y$ ,  $B_F$ , and  $B_\theta$  [9]. The frequency and angular bandwidths ( $B_F$ ,  $B_\theta$ ) can be set to constant values that match psychovisual data. The frequency ( $F$ ) and orientation ( $\theta$ ) define the center location of the filter. To determine the unknowns  $\sigma_x$  and  $\sigma_y$ , two equations are established. The variable  $\sigma_x$  is determined by setting the frequency cut-off to  $-6$  db:

$$\sigma_x = \frac{\sqrt{\ln 2} (2^{B_F} + 1)}{\sqrt{2\pi F} (2^{B_F} - 1)}.$$

By setting the cut-off in the angular direction to  $-6$  db,  $\sigma_y$  is determined:

$$\sigma_y = \frac{\sqrt{\ln 2}}{\sqrt{2\pi F} \tan(B_\theta/2)}.$$

A circular Gaussian may be desired so that there is a constant spatial extent in all directions. In this case,  $\sigma_x$  is set to  $\sigma_y$  ( $\sigma = \sigma_x = \sigma_y$ ) and  $B_\theta$  is calculated.

The position ( $F$ ,  $\theta$ ) and bandwidth ( $\sigma_x$ ,  $\sigma_y$ ) of Gabor filters in the frequency domain must be carefully established to properly capture textural information. Center frequencies of channel filters must lie close to the characteristic texture frequencies or else the filter responses will fall off too rapidly. This assumes that the identifiable features for certain textures are sufficiently spaced in spatial-frequency so that cross-filter interference does not occur. Since Gabor filters are not fully bandlimited, some aliasing will occur. If Gabor filters are set up to have a well-localized measure of the local frequency, minor noise can generate outputs that do not accurately represent the underlying signal. Some form of smoothing of the feature maps is commonly used to alleviate this concern (see Section 4.1).

### 3.2. Gabor filter implementations

#### 3.2.1. Supervised methods for Gabor filter selection

Numerous methods to apply Gabor filters for texture analysis have been used. Bovik et al. [9] mention three

supervised approaches to select filter locations using empirical information based on the power spectrum characteristics of the individual textures. For strongly oriented textures, the most significant spectral peak along the dominant orientation direction is used as a filter location. Picking the lower fundamental frequency identifies periodic textures. For nonoriented textures, using the center frequencies of the two largest maxima is recommended. Since identification usually requires multiple peaks for any generic texture, these approaches are unwieldy. Identifying unique textures with small regions of support is also difficult.

Dunn and Higgins [11] have developed a method to select optimal filter parameters based on known samples of the textures. The objective of this wholly supervised approach is to identify textural boundaries using only a minimum number of filters. Only “the” particular filter that optimally separates two texture classes is used to partition an image. This “optimal” filter may reflect strong textural characteristics of one class but may actually express a lack of textural information of the other class. To use just a single filter to discriminate each texture pair may be difficult because each texture is subject to spatial variability and the texture may contain multiple dominant components.

#### 3.2.2. Unsupervised methods for Gabor filter selection

Obviously, automated methods that identify texture characteristics are more appealing. Instead of trying to identify unique peaks that belong to unique textures, another approach is to use a filter bank containing filters spread throughout the spatial-frequency domain field. By providing near uniform coverage of the spatial-frequency domain with Gabor filters, the issue of selecting only texture-dependent center frequencies is avoided.

If a constant value of  $\sigma$  were to be selected for every  $F$ , then the decomposition would be that of a short-time Fourier transform. From a practical perspective, it makes sense to increase the spatial-frequency bandwidth of the filter with increasing frequency since the power spectrum typically decreases in a logarithmic fashion and since this corresponds to narrower impulse responses for higher frequencies. Given that natural imagery can often be considered to be non-stationary and wideband, the Gabor decomposition can capture information at different scales and orientations with the proper matching frequency.

Jain and Farrokhnia [10] implemented real Gabor filters for texture segmentation using frequency bandwidth ( $B_F$ ) of one octave, center frequency spacing ( $S_F$ ) of one octave, angular bandwidth ( $B_\theta$ ) of  $45^\circ$ , and angular spacing ( $S_\theta$ ) of  $45^\circ$ . Unit octave spacing matches the experimentally determined HVS ability, however,  $45^\circ$  angular bandwidths are not in agreement. As mentioned earlier, a smaller bandwidth of about  $30^\circ$  is more characteristic of the HVS. To provide minimal overlap and still

maintain reasonable coverage in the spatial-frequency domain,  $B_\theta$  is set equal to  $S_\theta$ . If  $B_\theta$  is greater than  $S_\theta$  redundancies occur which can deteriorate the feature set.

#### 4. Feature extraction of Gabor filter outputs

A host of different feature extraction methods can be applied to the Gabor filter outputs. Considerable information can be produced by the Gabor filter analysis and it is the efficient manipulation of this data that will provide appropriate texture features. The objective here is to compare the following published and novel methods for their ability to create unique texture signatures based on Gabor filter outputs. Although the published methods are commonly employed for texture analysis, no known publication has ever considered which method creates more separable texture features. The investigated methods are listed here and described immediately afterwards.

##### *Journal published methods:*

- using the magnitude response [9]
- applying spatial smoothing [9]
- using only the real component [10]
- applying full-wave rectification [12]
- using a sigmoidal function [10]
- creating moments (geometric and central) based on the spatial-frequency plane [13]

##### *Novel methods:*

- using a local filter variance measure to characterize texture complexity
- using a measure of consistent local filter response to characterize texture complexity
- implementing a syntactic approach to characterize the filter outputs.

#### 4.1. Journal published methods for Gabor filter feature extraction

##### 4.1.1. Magnitude response

Texture identification can be performed based on the magnitude of the output of the Gabor functions [9]. In the case of a filter that “matches” the particular texture, the magnitude of the output is large to enable identification. Filters that do not match the spatial-frequency components of dominant texture characteristics should have a negligible response and can be safely ignored as characteristic of that particular texture. Low responses to any filter may be important to identifying a texture if the responses are consistent.

##### 4.1.2. Spatial smoothing

Gaussian smoothing is known to improve the performance of Gabor filters for texture analysis. Bovik et al. [9] understand that textures which do not have sufficiently narrow bandwidths suffer from “leakage”. They reduce the effects of leakage by postfiltering the channel amplitudes with Gaussian filters having the same shape as the corresponding channel filters but greater spatial extents. The extent is controlled by  $\gamma$ . If  $g(x, y)$  is the Gaussian function of the Gabor filter, i.e.:

$$h(x, y) = g(x, y) \exp(2\pi j F x)$$

then the function that smoothes the filter’s magnitude response is  $g(\gamma x, \gamma y)$ . Bovik recommended setting  $\gamma$  to  $\frac{2}{3}$ . The smaller the value of  $\gamma$ , the greater the smoothing. To match the smoothing of the Gabor outputs with the Gaussian of the Gabor filter makes sense since this should yield results that are still spatially well localized. Jain and Farrokhnia [10] utilize uniform smoothing filters with window sizes as a function of the filter frequency. Gaussian filters should be better suited to maintaining textural boundaries.

There exists a physiological reason for utilizing smoothing since it mimics characteristics of the HVS. Hall and Hall [12] describe the existence of sustained channels in the visual system, indicating that the HVS not only considers pixels in the field of view, but also pixels in the vicinity.

##### 4.1.3. Real component only

Jain and Farrokhnia [10] use a bank of even-symmetric Gabor filters to characterize the channels. The authors justify using only even-symmetric Gabor filters on psychophysical grounds, but no full explanation of this basis is provided. Although less computationally intensive to spatially filter than the complex version (only one convolution mask instead of two), no such gains are offered for filtering in the spatial-frequency domain.

##### 4.1.4. Rectification

Instead of using the magnitude response of the Gabor filter, there has been speculation that a non-linear operation, such as rectification, may be more suitable for feature extraction. Many HVS models consider the involvement of non-linear behavior [12]. Full-wave rectification (summing the absolute value of the real and imaginary responses) is a nonlinear method that can be used to process the complex filter outputs. Such a process was advocated by Bergen and Adelson [14] for texture perception models.

##### 4.1.5. Sigmoidal function

Jain and Farrokhnia [10] subject each filtered image to a nonlinear transformation that can be interpreted as

a blob detector. This nonlinear transformation, reminiscent of a sigmoidal activation function used in artificial neural networks, is indicated by

$$\varphi(t) = \tanh(\alpha t) = \frac{1 - e^{-2\alpha t}}{1 + e^{-2\alpha t}}$$

where  $\alpha$  is a constant. They use an empirical value of  $\alpha = 0.25$ . This sigmoidal function continues to be used in practice [15], although no comparison with other Gabor filter feature extraction techniques has been identified.

#### 4.1.6. Moments based on spatial-frequency plane

Instead of using the magnitude responses, Bigun and du Buf [13] use moments of the Gabor responses to reduce the number of texture features generated. These moments are based on the power spectrum Gabor filter responses. They refer to the squared magnitude response as the “local power spectrum”. Geometric moments are defined by

$$m_{pq} = \sum_{i,j} \omega_i^p \omega_j^q |g(\omega_i, \omega_j)|^2,$$

where  $p$  and  $q$  are integers ( $\geq 0$ ) which represent the moment order. The authors use  $p + q < 5$  to reduce their feature set from 30 filter outputs down to 15 features. The parameters  $(i, j)$  index the power spectrum. The filter response is  $g(\omega_i, \omega_j)$ , defined in terms of Cartesian spatial frequency coordinates. Central moments are defined by

$$\mu_{pq} = \sum_{i,j} (\omega_i - \omega_x)^p (\omega_j - \omega_y)^q |g(\omega_i, \omega_j)|^2,$$

where  $\omega_x = m_{10}/m_{00}$  and  $\omega_y = m_{01}/m_{00}$ . Bigun and du Buf provide no analysis comparing the Gabor filter magnitude responses with these moment features. In addition, only central moments are employed in their segmentation studies, without any explanation for the preference. Note that another form of a moment can be generated using the magnitude filter response instead of the power spectrum.

#### 4.2. Novel methods for Gabor filter feature extraction

As mentioned earlier, there are seemingly three important factors that the HVS uses to identify texture: frequency, orientation, and complexity. Orientation and frequency are inherently accounted for in the Gabor filters. In the case where textures are not very regular, the complexity (or regularity) of the texture is expected to be an important issue. Irregular or complex textures are not as detectable as regular textures by the Gabor filters because of filter leakage caused by the spatial texture variability. Additional features may be necessary to complement the existing feature set to address complexity.

##### 4.2.1. Local variance of the filter response as a complexity measure

A regular (non-complex) texture has a consistent appearance. The strongest Gabor filter responses to a particular texture should be consistent. A complex texture will have dominant Gabor responses that fluctuate. Accordingly, the local spatial variance of a certain filter response can be calculated to determine the degree to which the filter response is changing. Those textures with low variances have a low degree of complexity and those textures generating high variances are complex. This characteristic can be used as an additional feature for the Gabor feature set. It will be referred to as the “variance complexity” texture measure.

##### 4.2.2. Filter response consistency with increasing Gaussian envelope as a complexity measure

A second method to determine complexity involves applying multiple Gabor filters by incrementally increasing the spatial bandwidth (keeping  $F$  and  $\theta$  constant). If the response across the multiple filters is consistent, then the texture is regular since it has the same response over a wider spatial extent. The slope of the responses can be used as a texture feature. The more horizontal the slope, the more consistent the filter response, and thus, the more regular the texture. This feature can also be used to augment the existing Gabor filter bank feature set. It will be referred to as the “consistency complexity” texture measure.

##### 4.2.3. Syntactic characterization

The final method for manipulating the Gabor filter outputs is a syntactic approach. If all the filter outputs are ranked in order of magnitude, then one would expect that the ranking is consistent for the same texture. By assigning each filter its own symbol (an integer value), a syntactic string is created. There are existing methods for string recognition that can be implemented for cluster analysis [16]. Here, the strings representing the filter outputs are unique since they are all of the same length and each symbol is only represented once. A “distance” between two strings  $A = a_1 a_2 \dots a_M$  and  $B = b_1 b_2 \dots b_M$  can be determined by finding  $j$  for which  $a_i = b_j$ ,  $i = 1$  to  $M$ . Then the distance measure can be found by summing all of the differences of  $i$  and  $j$ :

$$d(A, B) = \sum |i - j| \text{ when } a_i = b_j \text{ for } i = 1 \text{ to } M.$$

The squared distance has also been used, but this did not significantly affect the overall results.

## 5. Testing methodology

In this section, a description of the data sets and classifiers used is presented.

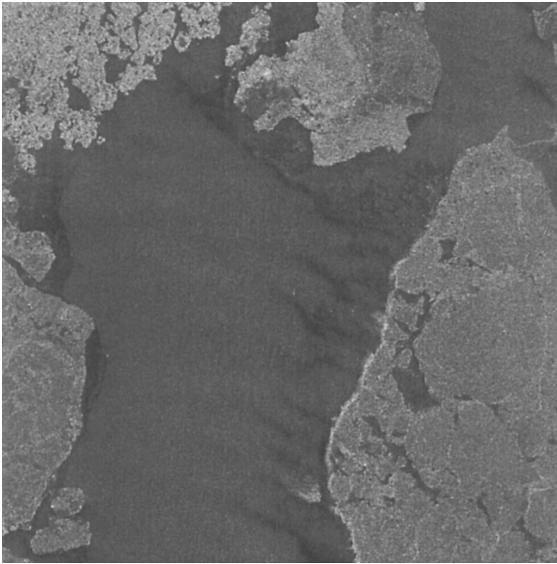


Fig. 1. Aerial SAR image obtained during the Labrador Ice Margin Experiment (Limex) during springtime 1989. The image has pixels with 100 meter spacing and was taken in nadir mode at incidence angles of between 50 and 70°. This image contains brash ice (top left hand corner), open water (dark regions in center; wind blowing left to right), and first year ice (right and left hand sides).

## 5.1. Classification testing

### 5.1.1. Data sets

Two data sets are used for the classification testing: textures taken from a SAR aerial image [17] (Fig. 1) and textures obtained from the Brodatz album [7] (Fig. 2). The SAR image (C-band, HH, obtained during the Labrador Ice Margin Experiment) contains three distinct classes: brash ice, first year smooth ice, and open water. Two sets of sixty-four  $8 \times 8$  samples of each class are selected for training and testing. Brodatz imagery is undoubtedly the most common test imagery used in the texture interpretation literature. This imagery provides opportunity for testing using a variety of classes and for comparing results with other research. Also, the training and test image samples are assured to contain one class only; the same cannot hold absolutely true for the SAR image samples, although efforts are made to accomplish this.

Eight different Brodatz textures are used for classification testing: cloth (D19), cork (D4), cotton (D77), grass (D9), paper (D57), pigskin (D92), stone (D2), and wood (D68). (The 'D\*' represents the numbering system assigned in the Brodatz album.) These textures are chosen since they have a noticeable but not necessarily regular textural pattern and several of the textures are similar in nature, comparable to textures found in natural imagery.

The cotton texture, used as a control, is the only one that has a well-defined pattern. Prior to classification, each  $256 \times 256$  image is normalized to 256 grey levels. Training samples are selected by dividing the upper left-hand quadrant into 64  $16 \times 16$  images. Test samples are selected by dividing the bottom right-hand quadrant into 64  $16 \times 16$  images. It is interesting to note that the  $16 \times 16$  samples of Brodatz imagery are not easy to visually discriminate. From an implementational perspective, it is easier to amalgamate the 64  $8 \times 8$  SAR samples into a single  $64 \times 64$  image and perform the filtering on the entire image using the spatial-frequency domain. Features based on pixels located at centres of the samples are used for the classification purposes. Feature vectors are dimensioned to the number of Gabor filters used.

### 5.1.2. Classifier

There are many methods to perform supervised classification. The method of choice here is the Fisher linear discriminant (FLD) [18]. The FLD provides a non-parametric method to reduce a multidimensional feature space down to a one-dimensional (1-d) feature space. The 1-d discriminant is able to classify two classes using the multidimensional features. If there are more than two classes, then an exhaustive, competitive pairwise comparison is performed. The advantages of using this classifier include: low computational load, optimal reduction of a multidimensional space to a 1-d space, and inherent normalization of the distance measures between the classes regardless of the scaling of the feature dimensions.

For the syntactic characterization classification, the measure only compares distances between individual strings. As a result, the  $k$ -nearest-neighbor classifier is implemented [18]. The computational demands of the  $k$ -nearest-neighbor routine are much higher than the demands of the FLD.

## 5.2. Unsupervised image segmentation

### 5.2.1. Data set

The image (Fig. 3) used for segmentation testing of various Gabor filter feature extraction techniques was originally published by Jain and Farrokhnia [10] and republished by Jain and Mao [19]. Five Brodatz textures appear in the image.

### 5.2.2. Clustering algorithm

The clustering algorithm used is the KIF (K-means Iterative Fisher) clustering algorithm [20]. KIF was designed for identifying classes in the feature space with hyperellipsoidal characteristics and Gabor texture features satisfy this characteristic [19]. This unsupervised algorithm is quite straightforward and very effective for the problem at hand. Only a synopsis is provided here, however, a full description is provided by Clausi [20]. The algorithm utilizes a binary hierarchical tree for

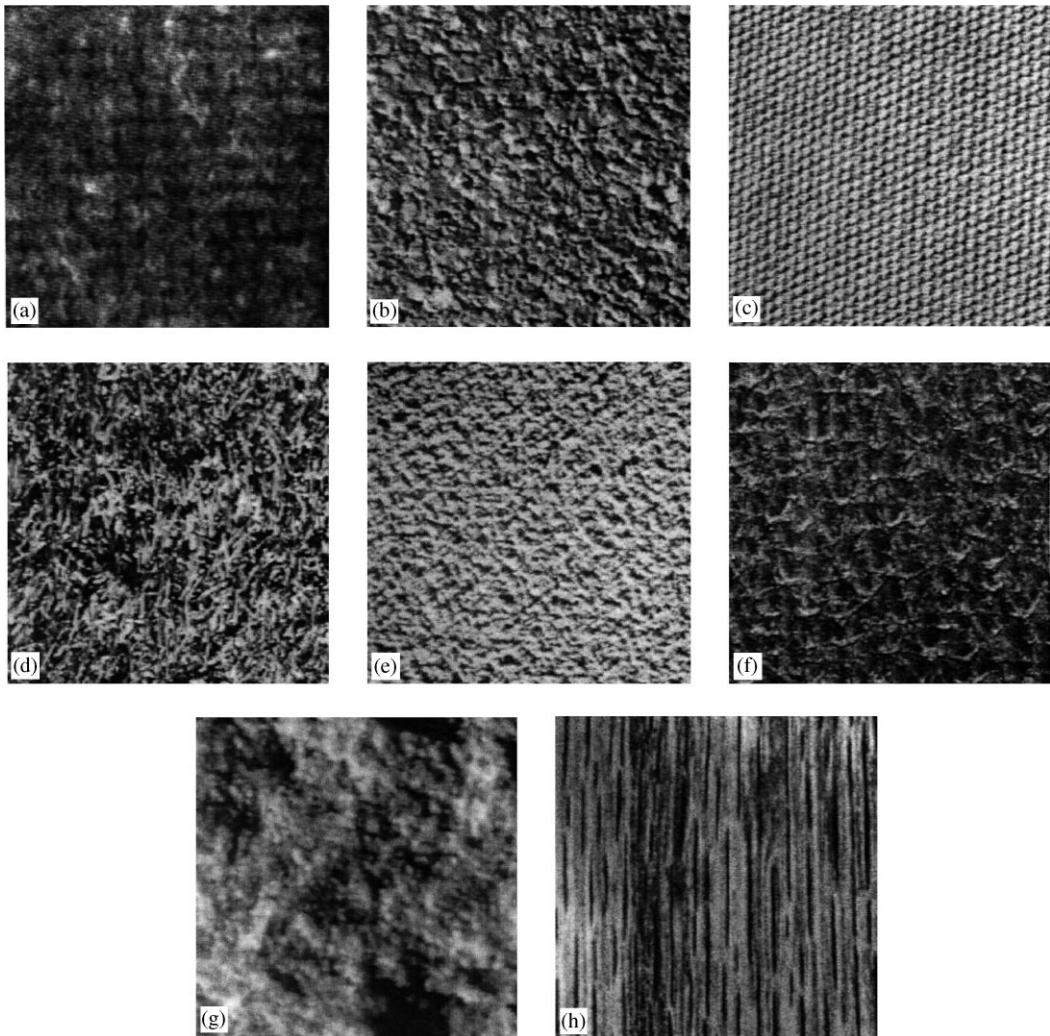


Fig. 2. Brodatz images used for classification testing. (a) Cloth [D19], (b) Cork [D4], (c) Cotton [D77], (d) Grass [D9], (e) Paper [D57], (f) Pigskin [D92], (g) Stone [D2], (h) Wood [D68].

breaking down the clusters in the feature space into appropriate classes. At each node in the binary tree, the following steps are performed to decide whether or not the cluster can be broken into two valid subclusters:

*Step 1:* K-means [21] is used to identify two dense regions in the cluster under consideration. Each dense region represents a subcluster. Each subcluster can contain one or more classes.

*Step 2:* A class pairwise Fisher linear discriminant [18,22] is used iteratively (iterative Fisher linear discriminant or iFLD) to improve the subclusters generated by K-means by taking account the subcluster covariances. Instead of using scatter ma-

trices, actual covariance matrices are used since an estimate of the number of samples per class is known. The iterations continue until the Fisher distance ( $\tau$ ) begins to reduce or until five iterations is reached.

*Step 3:* The Fisher distance ( $\tau$ ) represents a normalized distance separating the two subclusters. Two possible outcomes may occur. (i) If  $\tau$  exceeds some constant threshold ( $\tau'$ ), then the cluster will be split and the subclusters retained. At this point, each subcluster contains one or more true classes. Each subcluster is then processed through Steps 1–3. (ii) If the distance between the two subclusters is below  $\tau'$ , then the two subclusters represent a single class and the cluster is not split.

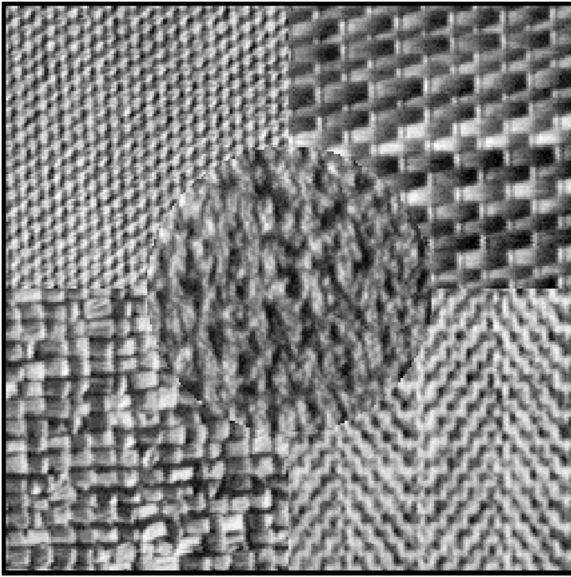


Fig. 3. Five texture Brodatz image originally published by Jain and Farrokhnia.

Both K-means and the FLD are nonparametric methods that make no explicit statistical assumptions about the data distribution for each cluster. The KIF method addresses the difficult problem of identifying the number of classes [21] when a suitable  $\tau'$  is selected for the image under analysis. Any  $\tau'$  in a certain range  $[\tau_{\min}, \tau_{\max}]$  has been demonstrated to be effective for a variety of imagery when Gabor filters are used thus generating a completely unsupervised image segmentation algorithm. The peak Fisher distance provides an indication of the separability of a pair of clusters. The greater the Fisher distance, the greater the separability of the clusters.

## 6. Classification and segmentation testing

Here, testing of the functional characterization of the Gabor filters is performed using classification. Testing each feature extraction method to determine relative classification ability follows. Unsupervised image segmentation is performed on different feature extraction methods based on the preferred functional characterization.

### 6.1. Classification testing of Gabor filter functional characterization

Since physiologically derived frequency bandwidths ( $B_F$ ) are approximately one octave, it makes sense to space center frequencies ( $S_F$ ) by the same amount to

minimize overlap of filters and maximize coverage of the spatial-frequency domain. Similarly, it makes sense to set the orientation bandwidth ( $B_\theta$ ) to the orientation spacing ( $S_\theta$ ). Since scientists generally use  $S_\theta = 45^\circ$ , yet  $S_\theta = 30^\circ$  agrees more closely to physiologically derived HVS characteristics, these two orientation bandwidths are compared. To compare the case of overlapping filters, setting  $S_\theta$  to  $30^\circ$  and  $B_\theta$  to  $45^\circ$  is performed. Bandwidths can also be constrained by assuming circular Gaussians ( $\sigma_x = \sigma_y$ ) as opposed to elliptical Gaussians ( $B_\theta = S_\theta$ ). Frequency radii ( $F$ ) of  $\sqrt{2}$ ,  $2\sqrt{2}$ ,  $4\sqrt{2}$  cycles per image (cpi) are used as center frequencies to filter the Brodatz samples and  $\sqrt{2}$ ,  $2\sqrt{2}$  cpi are used to filter the smaller Limex samples. Thus, when  $S_\theta = 45^\circ$ , there are 12 features for the Brodatz imagery and eight features for the Limex imagery. For  $S_\theta = 30^\circ$ , there are 18 and 12 features, respectively. The DC component is set to zero to prevent contribution of the mean local grey level. All filtering (Gabor and Gaussian smoothing) is performed in the spatial-frequency domain to prevent filter truncation that may occur in the spatial domain and to expedite the filtering process.

Results for Gabor filter features based on different functional characterizations are displayed in Table 1. Better results are obtained with the finer orientation spacing  $S_\theta = 30^\circ$ . Note that the performance of the classification is poorer when  $B_\theta = 45^\circ$  than  $B_\theta = 30^\circ$  given the same spacing of  $S_\theta = 30^\circ$ . In this case, one might consider that the additional overlap in the spatial-frequency could be conducive to improving the classification rate, however, the redundancy actually decreases the effectiveness of the features. Improvements are made by setting  $B_\theta = S_\theta$  (elliptical Gaussian) as opposed to setting  $\sigma_x = \sigma_y$  (circular Gaussian).

### 6.2. Classification testing of Gabor filter feature extraction

Instead of attempting to improve all the filter configurations presented in Table 1, only the most successful filter configuration ( $S_\theta = B_\theta = 30^\circ$ ) will be

Table 1

Classification accuracies (%) using Gabor filter outputs for different functional characterizations

$S_\theta$	Bandwidth	Brodatz		Limex	
		Train (%)	Test (%)	Train (%)	Test (%)
$45^\circ$	$B_\theta = 45^\circ$	72.9	68.2	84.4	86.5
$30^\circ$	$B_\theta = 45^\circ$	77.3	71.1	87.0	85.4
$30^\circ$	$B_\theta = 30^\circ$	81.6	77.5	91.2	87.5
$45^\circ$	$\sigma_x = \sigma_y$	75.8	72.1	87.0	85.4
$30^\circ$	$\sigma_x = \sigma_y$	79.7	74.6	88.5	87.5

Table 2

Classification accuracies of feature extraction performed on Gabor filter outputs using  $S_\theta = B_\theta = 30^\circ$

	Brodatz		Limex	
	Train (%)	Test (%)	Train (%)	Test (%)
Magnitude response	81.6	77.5	91.2	87.5
Real component only	70.7	59.6	83.9	82.8
Full-wave rectification	81.6	75.6	91.7	88.5
Sigmoidal function	78.5	75.2	91.7	88.5
Geometric moments	79.1	76.4	91.2	88.5
Central moments	66.2	58.2	92.7	86.5
Complexity (Variance)	82.6	77.7	94.8	88.0
Complexity (Consistency)	82.6	76.7	91.7	88.5

Table 3

Classification accuracies of feature extraction performed on Gabor filter outputs using  $S_\theta = B_\theta = 30^\circ$  and Gaussian smoothing ( $\gamma = 2/3$ )

	Brodatz		Limex	
	Train (%)	Test (%)	Train (%)	Test (%)
Magnitude response	98.1	95.3	100	98.4
Real component only	98.2	95.3	100	99.5
Full-wave rectification	98.1	94.5	100	98.4
Sigmoidal function	96.7	93.0	100	99.5
Geometric moments	96.5	95.5	100	99.5
Central moments	87.5	76.6	100	92.7
Complexity (Variance)	96.3	92.6	100	98.4
Complexity (Consistency)	98.4	95.5	100	95.8

post-processed. Post-processing results have been calculated for the next best case ( $S_\theta = 30^\circ$ ;  $\sigma_x = \sigma_y$ ) and these are always slightly poorer than post-processing using  $S_\theta = B_\theta = 30^\circ$ . Results are presented in Table 2 (unsmoothed filter outputs) and Table 3 (smoothed filter outputs). The objective is to identify methods that are consistently strong performers across these diverse data sets.

Note that the results for the central and geometric moments utilize the magnitude response, not the power spectrum response as used by Bigun and du Buf. The classification results using the local power spectrum moments were poorer than those for the magnitude moments of the Brodatz imagery and about the same or poorer for the Limex imagery. The moment order was set to  $p + q < 4$ , reducing the number of features from either 18 (Brodatz) or 12 (Limex) to 10.

Both the variance and consistency complexity measures have been estimated based only on the filter

output with the highest magnitude response for each pixel. This filter has the frequency and orientation that is assumed to dominate the response of the HVS and thus controls the regularity that we observe and interpret. The local variance estimate of complexity has been implemented using  $3 \times 3$  windows on the magnitude plane for the filter with the peak response. The consistency feature is determined by increasing the filter bandwidth by 5% twice. Different window sizes and percentage increases have been used and results are consistent. The slope of the three values (determined by linear regression) is used to represent the consistency feature. In both cases (variance and consistency), the feature augments the existing feature set created by the magnitude response determined by the Gabor filter bank. If the additional feature is able to improve the classification accuracy, then this feature provides a distinguishing characteristic not captured in the existing feature set.

### 6.2.1. Smoothing versus no smoothing

Various values of  $\gamma$  for smoothing purposes were attempted and  $2/3$  performed well all the time so it was used in the results presented here. The smoothed responses (Table 3) definitely yield results that are substantially better than their unsmoothed counterparts (Table 2). The methods with the strongest classification rate improve test classification accuracy from high 70s to mid 90s for the Brodatz imagery, an increase of over 20%. Increases to near perfect accuracy are generated for the Limex imagery.

### 6.2.2. Feature extraction of raw filter outputs (no smoothing)

Without any smoothing, the magnitude response, full wave rectification, sigmoidal function, geometric moments, and complexity measures all produce strong, similar classification accuracies (Table 2). Central moments and the real component underperform. That the real unsmoothed results are poor makes sense since a matched real Gabor filter response to a sinusoidal signal generates a sinusoid in the spatial domain. The sinusoidal variations are not conducive for accurate classification. In contrast, the complex filter response to a sinusoid is constant, generating a consistent measurement that assists accurate classification. This aspect is demonstrated by considering the 1-d case. The time domain representation of a real Gabor filter is

$$h(t) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\left[\frac{t^2}{\sigma^2}\right]\right\} \cos\{2\pi Ft\}.$$

The frequency representation of the real component of the Gabor filter is defined as the convolution of two spectra:

$$H(u) = \exp\{-2\pi^2 u^2 \sigma^2\} * \frac{1}{2} [\delta(u + F) + \delta(u - F)]$$

or

$$H(u) = \frac{1}{2} [\exp\{-2\pi^2[(u-F)^2\sigma^2]\} + \exp\{-2\pi^2[(u+F)^2\sigma^2]\}].$$

A pure sinusoid represented in the frequency domain is

$$X(u) = \frac{1}{2} [\delta(u+F) + \delta(u-F)].$$

Multiplying these two signals yields the following filter response:

$$Y(u) = \frac{1}{4} [\delta(u+F) + \delta(u-F)]$$

which generates a pure sinusoid in the time domain:

$$y(t) = \frac{1}{2} \cos\{2\pi Ft\}.$$

Using the same input signal  $X(u)$  and filtering with a complex Gabor:

$$h(t) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\left[\frac{t^2}{\sigma^2}\right]\right\} \exp\{j2\pi Ft\}$$

which is represented in the frequency domain as

$$H(u) = \exp\{-2\pi^2[(u-F)^2\sigma^2]\}.$$

yields the following output:

$$Y(u) = \frac{1}{2}\delta(u-F)$$

which has a magnitude that is simply a constant in the time domain:

$$|y(t)| = \frac{1}{2}.$$

This result is illustrated in Fig. 4. Fig. 4(a) indicates an input signal,  $x(t)$ , composed of two sinusoids in sequence, the latter sinusoid being half the frequency of the first. Fig. 4(b) shows the response of  $x(t)$  to a real Gabor filter tuned to the first sinusoid. This response has obvious sinusoidal variations. Fig. 4(c) shows the response of  $x(t)$  to a complex Gabor filter tuned to the first sinusoid. This response is a constant value of  $\frac{1}{2}$  for the signal region with the tuned frequency  $F$  and a lesser constant value for the region with frequency  $F/2$ . Classification based on constant responses would have more consistent results than those generated by sinusoidal responses. This advocates the use of complex Gabor filters over real Gabor filters for the purpose of texture analysis.

### 6.2.3. Feature extraction of smoothed filter outputs

The same group of methods that performed best without smoothing — plus the addition of the real component method — also performed best following smoothing (Table 3). Using a wider Gaussian than the Gaussian in the Gabor filter smooths the sinusoidal variation generated by the real component sufficiently to generate a similar classification result. Once again, the central moments underperform the other methods.

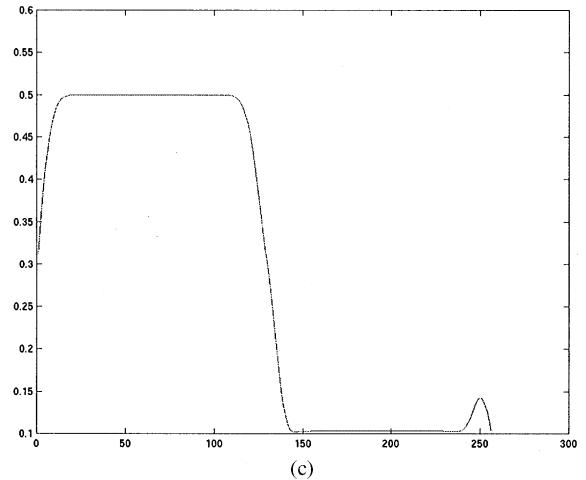
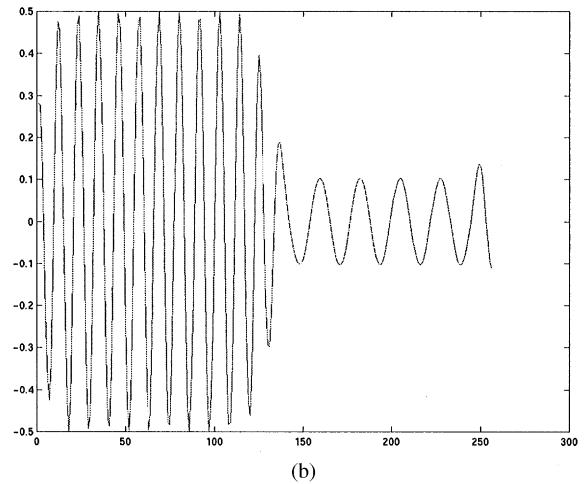
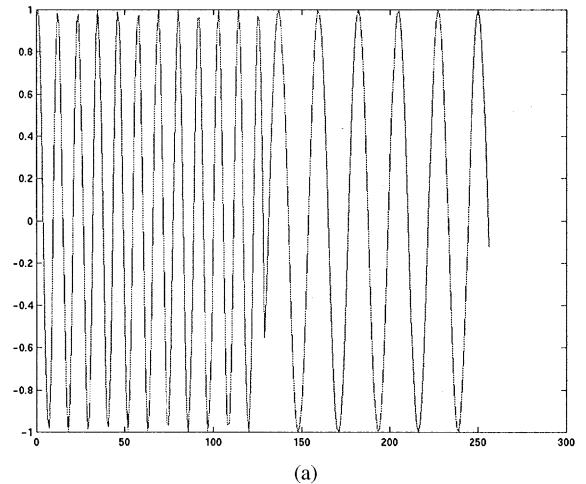


Fig. 4. (a) Input signal  $x(t)$  composed of two sinusoids in sequence, the latter with half the frequency of the first. (b) Response of  $x(t)$  to a real Gabor filter tuned to the first sinusoid. (c) Response of  $x(t)$  to a complex Gabor filter tuned to the first sinusoid.

#### 6.2.4. General results

For most cases, especially when no smoothing is performed, augmenting the feature set with a complexity measure provides a minor increase in classification success. These complexity measures do take into account some additional local information, in the same manner that smoothing takes into account local information. However, the complexity measures provide very localized accounts of the response of the adjacent pixels and are perhaps better suited for image segmentation of regions with jagged boundaries.

Results using the syntactic approach are poor (<60% for Brodatz and <50% for Limex; results only for smoothed training data sets). When the filter orientations are spaced by  $45^\circ$ , contrary to the other testing, classifica-

tion accuracy of the Brodatz imagery increases, but the results are still below 80% (Limex results remain below 50%). For brevity, these results are not presented in the tables. The syntactic approach does not seem appropriate for textures that are not very distinguishable, however, in the case of distinct textures, they can be easily used to identify and code a texture in a very compact manner. For example, none of the cotton samples are misclassified using the syntactic technique.

#### 6.3. Segmentation testing

The original image contains five Brodatz textures. All results displayed in Fig. 5 utilize smoothed feature sets given the parameters ( $B_\theta = S_\theta = 30^\circ$ ,  $B_F = S_F = 1$

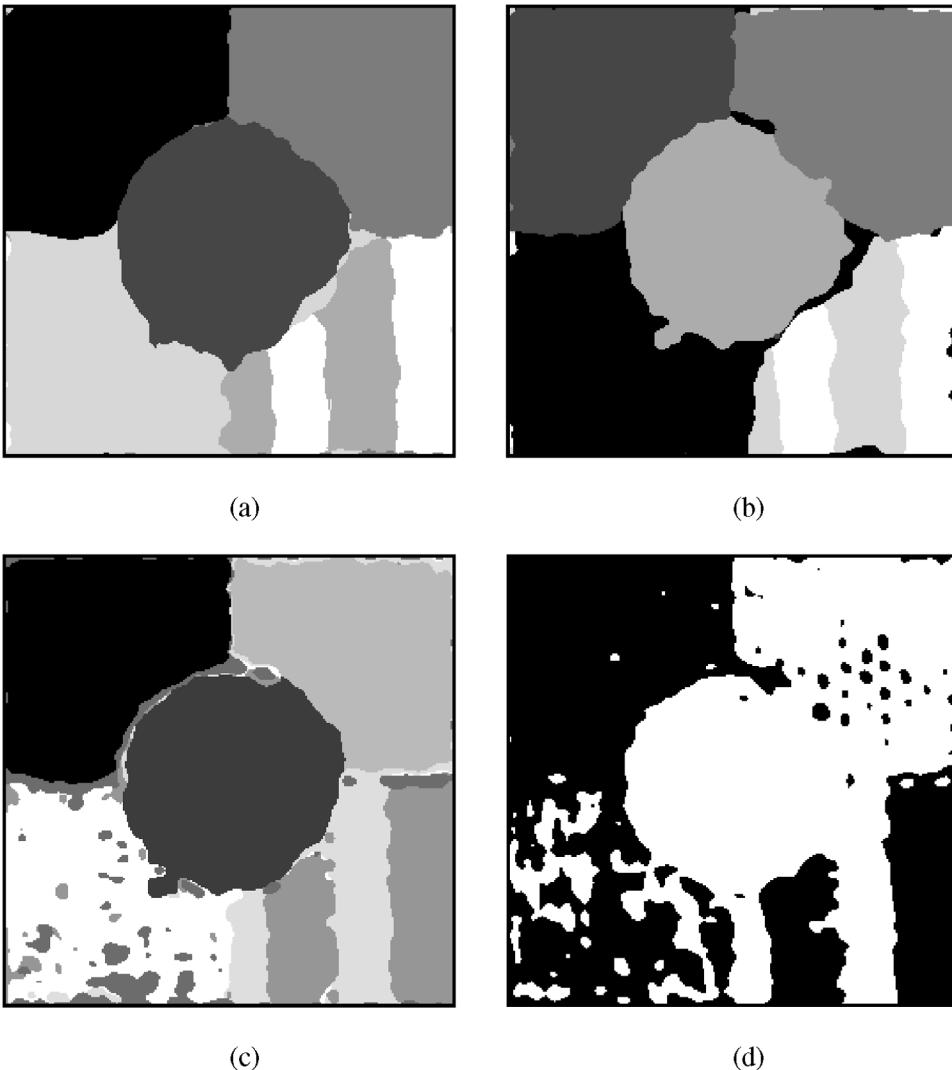


Fig. 5. Examples of unsupervised image segmentations using various Gabor filter feature extractions: (a) Magnitude response and consistency complexity measure. (b) Sigmoidal activation. (c) Geometric moments. (d) Central moments.

octave). Given that the Gabor filters are not rotationally invariant and that an unsupervised segmentation algorithm is used, the herringbone texture in the bottom right-hand corner should be segmented into two regions based on the grating orientation [23]. This is the result observed in Fig. 5(a) where the smoothed magnitude responses augmented with the consistency complexity feature is used as the feature set. For brevity, since similar segmentations are obtained using the smoothed versions of the magnitude responses alone, real responses, full rectification, and magnitude responses augmented with local variance complexity measures, these segmentations are not displayed. Sigmoidal features are also able to segment the image properly (Fig. 5(b)), but, the segmentation is not as successful as the results using the magnitude response and the consistency complexity measure (Fig. 5(a)). Geometric moments are able to segment the image and determine the proper number of classes (Fig. 5(c)), but the quality is noticeably poorer than the previous two examples (Fig. 5(a) and (b)). Central moments are unable to create features that segment properly since the class clusters are not well separated in the feature space (Fig. 5(d)). In this case, the two clusters identified each have considerable error in the class assignments.

Ranked allowable ranges of  $\tau$  for all represented results in Fig. 5 are presented in Table 4. The magnitude response alone has an allowable range of 11.4–14.0. Although the sigmoidal method achieves a fairly accurate segmentation, the range of  $\tau$  is limited (11.2–11.6). Geometric moments require a range of  $\tau$  with values quite lower (6.77–8.29), indicating the class clusters are closer in feature space. The method with the preferred range of

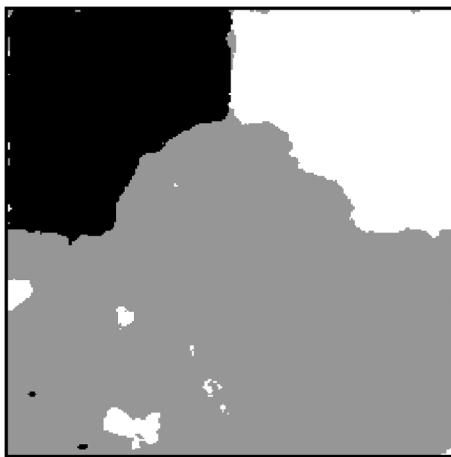
$\tau$  uses the magnitude response coupled with the consistency complexity measure (10.4–14.1).

Fig. 6 illustrates the drawback of not smoothing the feature maps. Fig. 6(a) depicts a segmentation generated by using the raw magnitude response coupled with the consistency complexity measure. Only three classes are identified. The results of using the raw magnitude response and using the magnitude response with the local variance complexity measure achieved similar results, so these results are not displayed. Using only the raw real response (Fig. 6(b)) also identifies only three classes. Here, inconsistent regional classifications are noticeable (note the rippling in the top right hand corner). This is a result of sinusoidal functions being generated as responses to the real Gabor filter, as demonstrated in Section 6.2.

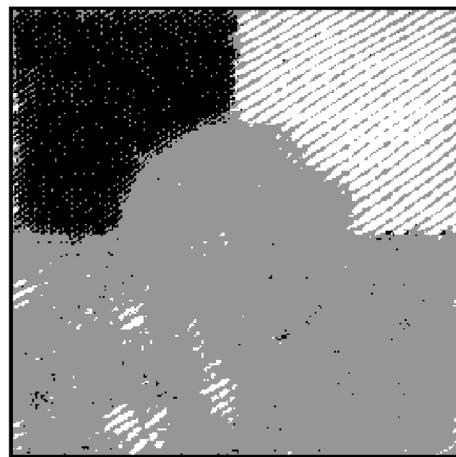
Table 4

Allowable ranges of  $\tau$  to achieve unsupervised image segmentation for various Gabor filter feature extraction methods

Feature extraction technique	Allowable range of $\tau$
Magnitude response and consistency	10.4 – 14.1
Magnitude response and variance	10.7 – 14.0
Magnitude response	11.4 – 14.0
Full rectification	11.3 – 13.9
Real response	11.4 – 13.9
Sigmoidal activation	11.2 – 11.6
Geometric moments	6.77 – 8.29



(a)



(b)

Fig. 6. Examples of unsupervised image segmentations without smoothing of feature maps: (a) Magnitude response and consistency complexity measure. (b) Real response.

## 7. Discussion and conclusions

### 7.1. Summary

Since better results are obtained using  $B_\theta = S_\theta = 30^\circ$ , this is the choice for any texture classification/segmentation studies. There is some concern that the added dimensionality of  $30^\circ$  spacing may not be warranted due to the increased computational demands. That is, using  $30^\circ$  spacing instead of  $45^\circ$  spacing increases the feature set by 1.5 times and there is a corresponding increase in memory, disk space, completion times, and swap space to perform the analysis. For some test texture images, setting  $S_\theta$  to  $45^\circ$  will generate a proper segmentation, however, to generate a more robust and universal feature set,  $S_\theta = 30^\circ$  is recommended.

With higher smoothing, better classification results occur. This may be detrimental when textures representing small spatial regions are to be segmented. Using the complexity measures provides a method to utilize local information using a smaller spatial extent and may be preferable under such circumstances. Additional computations are required to capture the complexity measures.

Using the smoothed magnitude response augmented with either of the complexity measures generated the preferred unsupervised texture segmentation. The drawback to using the consistency complexity measure is the three-fold increase in the number of filters required. Full wave rectification also generates preferred Gabor texture features for texture classification or segmentation. Using the real component requires smoothing to achieve reasonable results, so this method is not advocated. The smoothed real responses do generate strong discriminations, however, if less smoothing is used, how these features will respond is uncertain. Using the sigmoidal function or geometric moments, although successful for classification, did not generate appropriate results for segmentation. If a reduced feature set is a necessity, then geometric moments may be used. Central moments do not achieve the same classification and segmentation success as the other methods and this method is not recommended.

Generally, magnitude, full wave rectification, sigmoidal activation, and geometric moment responses all perform consistently the best in classification. For segmentation, magnitude and full wave rectification performed the best. Given these results, there is no added value for the feature set to performing additional calculations for sigmoidal activation and geometric and central moments. Added value is obtained by augmenting the feature set with one of the complexity texture measures.

Further investigation is warranted for the syntactic approach to ascertain its ability to code in a very compact manner distinct textures. If the discrimination of highly distinct textures using compact codes is required,

and the syntactic approach may be well suited for such applications.

### 7.2. Examples utilizing Gabor filters for texture analysis

Augusteijn et al. [24] classified a Thematic Mapper (TM) set of images using a number of different methods, including co-occurrence matrices, power spectrum, and Gabor features. Power spectrum features (a FIR method for estimating local frequency) and Gabor filters are found to be preferred choices. Only the real component of the Gabor filters is utilized, without smoothing. The Gabor filters are implemented using a constant bandwidth for all frequencies, which generates non-orthogonal (redundant) spatial-frequency domain coverage and does not take advantage of the wavelet capabilities of the Gabor filter. Oddly, no information is provided concerning the nature of the orientation of the Gabor filters. Classification improvements could have been made using a smoothed complex Gabor filter with octave spacing/bandwidths and  $30^\circ$  spacing/bandwidths.

Ohanian and Dubes [25] compare four different techniques for their classification ability: Markov random field parameters, Gabor filters, fractal based features, and co-occurrence features. Here, Gabor filters do not perform as well as the co-occurrence measures. Gabor filtering is probably not performing optimally since a real filter is used with constant spatial-frequency bandwidth, with  $45^\circ$  spacing/bandwidth, and without any local smoothing.

Strand and Taxt [26] compare Gabor and co-occurrence supervised texture segmentation with their new method (one that essentially calculates a local frequency measurement). The Gabor technique is implemented ineffectively for a number of reasons. For example, their method parallels Jain and Farrokhnia's which is limited to  $45^\circ$  filter spacing and bandwidth and uses a sigmoidal activation function for feature extraction. Most importantly, the implemented Gabor approach uses center frequencies that are clearly too low to be helpful for discrimination given the textures used in their study (for example,  $\sqrt{2}$  cycles per image (cpi) and  $2\sqrt{2}$  or cpi for  $256 \times 256$  images). Also, the two highest octave frequency bands for a  $256 \times 256$  image ( $5\sqrt{2}$  cpi and  $6\sqrt{2}$  cpi) are not included. These would have definitely assisted the segmentation since the cotton and herringbone Brodatz textures are better recognized using these higher frequencies. Filters are restricted to only 0 and  $90^\circ$  orientation, even though the Brodatz herringbone texture (which has strong 45 and  $135^\circ$  features) is used in the test image. The Gabor filters in the 0 and  $90^\circ$  directions will be weakly sensitive to the 45 and  $135^\circ$  texture characteristics, thus, little or no discriminating information is captured by these filters. Gabor filters are implemented spatially using fixed window sizes of  $32 \times 32$  causing the filters with  $\sqrt{2}$  cpi and  $2\sqrt{2}$  cpi to be dramatically truncated.

## 8. Conclusions

For the above three publications, using our recommended Gabor filter configuration and feature extraction would have probably improved results. Successful results identifying textures using Gabor filters have been generated by:

- using a pseudo-wavelet implementation.
- providing full coverage of the spatial-frequency domain, without significant overlap between individual filters.
- setting the filter's frequency spacing and bandwidth to one octave and orientation spacing and bandwidth to  $30^\circ$ .
- setting the DC gain to zero to prevent classification based on tone.
- utilizing the magnitude response augmented with the consistency complexity feature.
- smoothing outputs using a Gaussian filter with the same shape as the corresponding channel filter but greater spatial extent.
- performing all filtering operations (Gabor and Gaussian smoothing) in the spatial-frequency domain for improved computational performance.

## 9. Summary

Gabor filters have tremendous potential to generate texture features from digital imagery. Typically, these filters are implemented without any attention paid to properly configuring the filter parameters and manipulating the raw outputs to generate optimal texture features for discriminating the regions of interest. This paper provides guidance to ensure that the Gabor filters are implemented to generate preferred texture discrimination ability. Both previously implemented and new methods to extract meaningful texture information using Gabor functions are described and evaluated using both classification and unsupervised segmentation data sets.

The human visual system (HVS) is known to identify textures based on three primary characteristics: frequency, orientation, and complexity. Gabor filters mimic the HVS since they are able to localize frequency and orientation characteristics, however, complexity is not as easily defined. This concept is related to the ease that the texture is described. For example, a texture represented by a single pure sinusoid has low complexity and a texture with multiple sinusoidal components (with varying frequencies and orientations) has higher complexity. It is the discrimination of textures based on frequency, orientation, and complexity that make the Gabor filter technique an excellent choice for this task. Two methods to address the complexity issue using Gabor filters are described and implemented. The first method involves

measuring the local spatial variance of a filter's response. Secondly, the slope of the responses of a filter with increasing spatial width is measured.

The Gabor filter bank may be constructed using an infinite number of filters. Here, various filter configurations have been compared by using different filter orientation and bandwidths. The filter bank approach that modeled the HVS ( $30^\circ$  and 1 octave filter bandwidth and spacing) in a wavelet manner and still created a near orthogonal system produced the preferred Gabor filter configuration. This preferred method was used to evaluate different feature extraction techniques: magnitude response, spatial smoothing, only the real component, full-wave rectification, sigmoidal function, spatial-frequency based moments, two complexity measures, and a syntactic approach.

Tests were performed using a SAR sea ice image and Brodatz images for classification and a previously published five-class Brodatz image for unsupervised segmentation. Smoothing of the texture feature maps significantly improves the classification rate and the segmentation accuracy. From a signal processing perspective, it is demonstrated that raw outputs based on only the real Gabor filter component are inappropriate for texture feature extraction. Using the magnitude response augmented with the consistency complexity measure generated the most consistently accurate results across both classification and segmentation testing.

## References

- [1] A.C. Bovik, Analysis of multichannel narrow-band filters for image texture segmentation, *IEEE Trans. Signal Proc.* 39 (9) (1991) 202–2043.
- [2] D.H. Hubel, T.N. Wiesel, Receptive fields and functional architecture in two nonstriate visual areas 18 and 19 of the cat, *J. Neurophysiol.* 28 (1965) 229–289.
- [3] F.W. Campbell, J.J. Kulikowski, Orientational selectivity of the human visual system, *J. Physiol.* 187 (1966) 437–445.
- [4] F.W. Campbell, J.G. Robson, Application of Fourier analysis to the visibility of gratings, *J. Physiol.* 197 (1968) 551–565.
- [5] D.A. Pollen, S.F. Ronner, Visual cortical neurons as localized spatial frequency filters, *IEEE Trans. Syst., Man, Cybern. SMC-13* (5) (1983) 907–916.
- [6] A.R. Rao, G.L. Lohse, Identifying high level features of texture perception, *CVGIP: Graphical Models and Image Proc.* 55 (3) (1993) 5218–5233.
- [7] P. Brodatz, *Textures: A Photographic Album for Artists and Designers.*, Dover, New York, 1966.
- [8] J.G. Daugman, Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortex filters, *J. Opt. Soc. Am. A* 2 (7) (1985) 1160–1169.
- [9] A.C. Bovik, M. Clark, W.S. Geisler, Multichannel texture analysis using localized spatial filters, *IEEE Trans. Pattern Anal. Machine Intell.* 12 (1) (1990) 55–73.

- [10] A.K. Jain, F. Farrokhnia, Unsupervised texture segmentation using Gabor filters, *Pattern Recognition* 24 (12) (1991) 1167–1186.
- [11] D.F. Dunn, W.E. Higgins, Optimal Gabor-filter design for texture segmentation, ICASSP 0-7803-0946-4 (1993) V37–V40.
- [12] C.F. Hall, E.L. Hall, A nonlinear model for the spatial characteristics of the human visual system, *IEEE Trans. Systems Man, Cybern.* SMC-7 (3) (1977) 161–170.
- [13] J. Bigun, J.M. du Buf, N-folded symmetries by complex moments in Gabor space and their application to unsupervised texture segmentation, *IEEE Trans. Pattern Anal. Machine Intell.* 16 (1) (1994) 80–87.
- [14] J.R. Bergen, E.H. Adelson, Early vision and texture perception, *Nature* 333 (26) (1988) 363–364.
- [15] R. Panda, B.N. Chatterji, Unsupervised texture segmentation using tuned filters in Gaborian space, *Pattern Recognition Lett.* 13 (1997) 445–453.
- [16] K. Sun Fu, *Syntactic Pattern Recognition and Applications*, Prentice-Hall, Toronto, 1982.
- [17] T. Hirose, T. Heacock, R. Duncan, A report on the evaluation of ice classification algorithms, Prepared by Noetix Research Inc. for Environment Canada Ice Branch and Ice Applications Group, Canada Centre for Remote Sensing, 1993.
- [18] R.O. Duda, P.E. Hart, *Pattern Classification and Scene Analysis*, Wiley, Toronto, 1973.
- [19] R.A. Fisher, The use of multiple measurements in taxonomic problems, *Annals of Eugenics* 7 (1936) 179–188.
- [20] D.A. Clausi, *Texture Segmentation of SAR Sea Ice Imagery*, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1, 1996.
- [21] A.K. Jain, R.C. Dubes, *Algorithms for Clustering Data*, Prentice-Hall, Englewood Cliffs, NJ, 1988.
- [22] A.J. Jain, J. Mao, K.M. Mohiuddin, Artificial neural networks: A tutorial, *IEEE Computer* 29 (3) (1996) 31–44.
- [23] D.A. Clausi, Algorithm for clustering hyperellipsoidal texture feature classes, in *IASTED/IEEE Signal and Image Processing SIP/98*, Las Vegas, NV, 1998, pp. 374–377.
- [24] M.F. Augusteijn, L.E. Clemens, K.A. Shaw, Performance evaluation of texture measures for ground cover identification in satellite images by means of a neural network classifier, *IEEE Trans. Geosci. Remote Sensing* 33 (3) (1995) 616–626.
- [25] P.P. Ohanian, R.C. Dubes, Performance evaluation for four classes of textural features, *Pattern Recognition* 25 (8) (1992) 819–833.
- [26] J. Strand, T. Taxt, Local frequency features for texture classification, *Pattern Recognition* 27 (10) (1994) 1397–1406.

**About the Author**—DAVID CLAUSI earned his B.A.Sc. (1990), M.A.Sc. (1992), and Ph.D. (1996) at the University of Waterloo (Waterloo, Ontario, Canada). He initiated his career by working in the imaging industry (Mitra Imaging Inc.) and then started his faculty career as an Assistant Professor in the Department of Geomatics Engineering at the University of Calgary (Calgary, Alberta, Canada). In 1999, he returned to the University of Waterloo to his current position as an Assistant Professor in Systems Design Engineering. His research interests include automated image interpretation, digital image processing, pattern recognition with applications in remote sensing and medical imaging.

**About the Author**—ED JERNIGAN received his B.S. (1969), M.S. (1971), and Ph.D. (1975) degrees from the Massachusetts Institute of Technology (MIT). After a post-doctoral year at MIT in the Cognitive Information Processing Group, he moved to the Department of Systems Design Engineering at the University of Waterloo, Ontario, Canada, where he is currently professor and director of the Vision and Image Processing Laboratory. His research interests include signal and image processing, pattern recognition, and perception, particularly in medical and remote sensing applications. Professor Jernigan is a recipient of the University of Waterloo's Distinguished Teacher Award, has been Program Director for the Shad Valley Summer Program for over ten years, was Director of Engineering Admissions, and became chair of Systems Design Engineering in September 1998.