

# Fabric defect segmentation using multichannel blob detectors

**Ajay Kumar**

Industrial Automation Research  
Laboratory  
Department of Electrical and Electronic  
Engineering  
The University of Hong Kong  
Pokfulam Road  
Hong Kong  
E-mail: ajaykr@eee.hku.hk

**Grantham Pang**

Industrial Automation Research  
Laboratory  
Department of Electrical and Electronic  
Engineering  
The University of Hong Kong  
Pokfulam Road  
Hong Kong  
E-mail: gpang@eee.hku.hk

**Abstract.** The problem of automated defect detection in textured materials is investigated. A new algorithm based on multichannel filtering is presented. The texture features are extracted by filtering the acquired image using a filter bank consisting of a number of real Gabor functions, with multiple narrow spatial frequency and orientation channels. For each image, we propose the use of image fusion to multiplex the information from sixteen different channels obtained in four orientations. Adaptive degrees of thresholding and the associated effect on sensitivity to material impurities are discussed. This algorithm realizes large computational savings over the previous approaches and enables high-quality real-time defect detection. The performance of this algorithm has been tested thoroughly on real fabric defects, and experimental results have confirmed the usefulness of the approach. © 2000 Society of Photo-Optical Instrumentation Engineers. [S0091-3286(00)01912-7]

Subject terms: defect detection; Gabor filters; multichannel filtering; textile industry; computer vision; quality assurance; industrial automation.

Paper 200028 received Jan. 24, 2000; revised manuscript received July 7, 2000; accepted for publication July 21, 2000.

## 1 Introduction

Motivated by human beings' robust texture-segmentation capabilities and by studies of human perception, many computer vision researchers have developed a texture segmentation paradigm that is based on the Gabor filter. In this paradigm, multiscale and multiorientation Gabor filters are used for feature extraction. These features are then combined, typically using one or more polling mechanisms, to segment defects. The framework for texture segmentation based on Gabor filters has been motivated by physiological evidence that the response of orientation-selective cells in the visual cortex can be modeled using Gabor filters.<sup>1</sup> Gabor-shaped receptive fields are fundamental to biological processing of texture, but any extensions of this approach to computer vision are only appropriate within a computational framework.

In recent years, a large number of techniques for texture analysis have been based on Gabor filters. This approach is inspired by the multichannel filtering theory of the processing of visual information in the biological model of the human visual system. As proposed by Campbell and Robson,<sup>2</sup> this theory argues that the human visual system decomposes the retinal image into a number of filtered images each of which contains intensity variations over a narrow range of frequency and orientation. Subsequent electrophysiological and psychophysical experiments indicate that the brain contains large array of neurons that filter for orientation and size information at each point in visual cortex.<sup>3</sup> Each of these neurons found in the visual cortex is tuned to a particular combination of frequency and orientation in a narrow range, which is referred to as a *channel*. These channels are interpreted as bandpass filters and bear a striking resemblance to Gabor functions.<sup>1</sup> Motivated by this understanding of human perception, many computer-

vision researchers have proposed texture segmentation methods based on the filter-bank model.<sup>4,5</sup>

### 1.1 Prior Work

Fabric defect segmentation has been studied for a long time using various approaches.<sup>6-14</sup> Computing features that capture textural properties are at the heart of most of these approaches. The success of a particular feature is in its ability to describe texture that agrees with human perception. Thus Gabor functions are good descriptors of texture features. A general framework for fabric defect detection can be formulated by capturing fabric features through Gabor functions and then comparing these features with a reference (i.e., a defect-free fabric sample). Defect segmentation can be successfully achieved by statistical calculations on these features. Texture features extracted from the co-occurrence matrix,<sup>6</sup> autocorrelation of web images,<sup>7</sup> the Karhunen-Loève (KL) transform,<sup>8</sup> and means and standard deviations of subblocks<sup>9</sup> have been used for detection of fabric defects. Modeling of fabric texture using the Gauss Markov random field (GMRF) and defect detection using statistics derived from this model have been detailed in Ref. 10.

The periodicity of yarns in textile fabric provides valuable information, and therefore Fourier domain features have been suggested for detecting fabric defects.<sup>11</sup> When defects cause global distortion in textile material, Fourier analysis is most suitable. But that is not true for local fabric defects, and therefore techniques that can simultaneously measure in the spatial and the spatial-frequency domain are more useful. Therefore multiscale wavelet representation (MSWAR)<sup>12</sup> and the Gabor filter bank<sup>13,14</sup> have been used to detect fabric defects. Prior work has used complex (i.e.,

real and imaginary) Gabor functions; however, we only use real Gabor functions.

## 1.2 Present Work

In this paper, we present a multichannel filtering technique based on real Gabor functions for segmentation of local texture defects. The technique has been developed and evaluated for on-line detection of local defects in textile webs. One of the advantages of the multichannel filtering approach<sup>4,5</sup> over other textural feature extraction approaches that use a small window size is its ability to segment both fine and coarse texture defects.<sup>15</sup> This is accomplished by segmenting fine and coarse texture defects on different scales (multichannel). We now detail how our approach is motivated by earlier work on mechanisms in the visual cortex of mammals.

Psychophysically observed spatial-frequency channels<sup>16</sup> and neurophysiologically observed blob-, bar-, and edge-sensitive neurons<sup>17</sup> have been used to explain texture perception in human vision. Malik and Perona<sup>18</sup> have presented a model for preattentive\* texture discrimination based on human visual mechanisms. They have shown that odd-symmetric (imaginary Gabor function) and even-symmetric (real Gabor function) filters are not treated identically in texture discrimination. One of the important conclusions of their work is that odd-symmetric filters are not useful in texture discrimination but even-symmetric filters are. They excluded odd-symmetric filters from their model, since they could not find any texture for which an odd-symmetric mechanism was necessary. Therefore their work<sup>18</sup> supports our defect detection scheme using only real (even-symmetric) Gabor functions and ignoring imaginary (odd-symmetric) ones.

Some researchers<sup>20,21</sup> have described how a well-known nonlinearity of visual system plays a significant role in texture discrimination. This nonlinearity is due to retinal adaptations,<sup>22</sup> and it follows a simple-cell-response (Gabor-shaped) stage. This nonlinearity enables the human visual system to respond to local contrast over 10 log units of illumination changes. As detailed in Ref. 18, there are at least two physiologically plausible causes for this nonlinearity: (a) a nonlinear contrast response function that typically has sigmoid shape for neurons that exhibit a threshold effect for low contrast and a saturation effect for higher contrast, (b) intracortical inhibition that could occur within and among responses in different channels. Malik and Perona,<sup>18</sup> in their texture perception model, have chosen to use (b), while ignoring (a). In our work we use the nonlinearity suggested in (a). The main contributions of this paper are summarized as follows:

1. This algorithm uses real Gabor functions instead of the complex Gabor functions used in Refs. 13 and 14. This is because the real Gabor function can act as a blob detector.<sup>5,26–29</sup> More justification is given in Sec. 1.2 and Sec. 3.
2. Computational and performance gain have been en-

hanced by the use of local nonlinear functions instead of the multiresolution pyramid used in Ref. 13.

3. An image fusion technique based on Bernoulli's rule of combination<sup>12</sup> is proposed to integrate information from different channels. This approach offers high detection rate and low false-alarm rate.
4. This algorithm is computationally economic, with the use of threshold tradeoff and smaller convolution masks. Performance analysis of this algorithm as a function of sensitivity and mask size is also presented.
5. A simple thresholding method is suggested to remove isolated noisy pixels, which does not require any morphological operations.

Prior texture segmentation work<sup>5</sup> using real Gabor functions has been concentrated on Brodatz's album.<sup>23</sup> However, the present work on defect segmentation pertains to real, meaningful patterns from the textile industry.

In this paper, "complex Gabor function" or "Gabor function" or "Gabor filter" refers to the real and imaginary parts (combined) of a Gabor function. Similarly, "real Gabor function" means only the real part of the Gabor function, and "imaginary Gabor function" means only the imaginary part.

The organization of the rest of this paper is as follows. In Sec. 2, preliminary theoretical foundations of this work are introduced. Those include spatial- and frequency-domain description of Gabor filters, and a review of prior work using real Gabor functions. In Sec. 3, a class of 2-D real-Gabor-function masks are introduced, which are used to encode images into multiple cosine-modulated subimages. That section includes a discussion of the selection of salient Gabor-function parameters for defect segmentation, followed by a brief discussion of supervised defect detection. Extraction of textural blob descriptors for online defect segmentation is described in Sec. 4. Section 5 describes statistical calculations on these texture descriptors. In Sec. 6, experimental results on synthetic as well as real fabrics are reported. Section 7 gives a discussion, followed by conclusions in Sec. 8.

## 2 Theoretical Foundations

In this section, we briefly review the basic properties of Gabor functions. This is followed by an introduction of prior work on real Gabor function.

### 2.1 Multiscale and Multiorientation Gabor Filters

A Gabor function is a complex exponential modulated by a Gaussian function in the spatial domain, and is a shifted Gaussian in the frequency domain. In general, an  $n$ -dimensional Gabor function is expressed as<sup>24,25</sup>

$$h(\mathbf{p}) = f(\mathbf{p})m(\mathbf{p}), \quad (1)$$

where  $f(\mathbf{p})$  is a Gaussian function (aperture) given by

$$f(\mathbf{p}) = \frac{1}{[(2\pi)^n |C|]^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{p}-\mathbf{p}_0)^T C^{-1}(\mathbf{p}-\mathbf{p}_0)\right], \quad (2)$$

and  $m(\mathbf{p})$  is a complex modulating function

\*Preattentive texture discrimination is attributed to difference in first-order statistics of stimulus features such as the orientation, size, and brightness of constituent elements.<sup>19</sup>

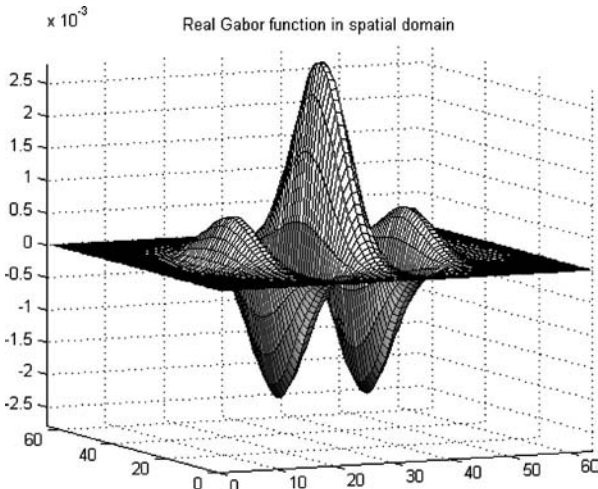


Fig. 1 Perspective view of real Gabor function in spatial domain.

$$m(\mathbf{p}) = \exp[j\boldsymbol{\omega}_0^T(\mathbf{p} - \mathbf{p}_0)] \tag{3}$$

where  $\mathbf{p}$ ,  $\mathbf{p}_0$ ,  $\boldsymbol{\omega}_0 \in R^n$ ,  $C$  is an  $n \times n$  positive definite covariance matrix, and  $|C| = \det C$ .

For the 2-D case (to be used henceforth), the horizontal and vertical spatial coordinates are represented by the vector  $\mathbf{p} = [x, y]^T$ . The shape of Gaussian function  $f(\mathbf{p})$  is controlled by the matrix  $C$ , and the vector  $\mathbf{p}_0 = [x_0, y_0]^T$  stands for the translation of its origin. If  $\sigma_x$  and  $\sigma_y$  are variances of the Gaussian function along the  $x$  and  $y$  axes, respectively, then

$$C = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix}, \tag{4}$$

where the constants  $\sigma_x$  and  $\sigma_y$  determine the scale and the width/aspect ratio, which is used to adjust the orientation sensitivity of the Gabor function. The vector  $\boldsymbol{\omega} = [u, v]^T$  represents two axial frequencies along two coordinates. The vector  $\boldsymbol{\omega}_0 = [u_0, v_0]^T$  represents the overall translation frequency of the Gabor function. In two dimensions, the Gabor filter is tuned to the orientation  $\theta = \tan^{-1}(v_0/u_0)$  from the  $u$  axis. In the frequency domain, the Gabor function acts as a 2-D bandpass filter represented as a shifted Gaussian, frequency-centered at  $\boldsymbol{\omega}_0 = [u_0, v_0]^T$ . The axis of modulation in the frequency domain is oriented at an angle of  $\tan^{-1}(v_0/u_0)$  from the  $u$  axis.

Figure 1 shows the perspective plot of the real component of a Gabor function. In the frequency domain, a Gabor function is a 2-D bandpass filter, represented as a shifted Gaussian function centered at  $(u_0, v_0)$ :

$$H(\boldsymbol{\omega}) = \exp[-\frac{1}{2}(\boldsymbol{\omega} - \boldsymbol{\omega}_0)^T C (\boldsymbol{\omega} - \boldsymbol{\omega}_0)]. \tag{5}$$

Equation (1) can be interpreted as the sum of two Gaussian functions that are cosine (real) and sine (imaginary) modulated. The impulse responses of these odd (real) and even (imaginary) Gabor functions are approximately Hilbert pairs. This approximation is more exact when their

amplitudes are close, and this can be ensured by choosing a Gabor filter with small half-peak bandwidth.<sup>24</sup>

### 2.2 Prior Work Using Real Gabor Functions

The real part of a Gabor function has been shown to be useful as a correlation filter for object detection.<sup>26-29</sup> Casasent and Smokelin<sup>26</sup> use a weighted combination of real Gabor functions to detect multiple classes of objects in clutter, with object distortions and contrast variations present. Their work employed initial real-Gabor-function parameters based on the nominal target characteristics, combined several real Gabor functions into a macro Gabor filter, and used a general neural network algorithm to refine parameters of the initial macro Gabor filter. A similar approach for object detection with reduced false alarms and higher probability of detection has been suggested in Ref. 27. In this approach a clutter Gabor function has been employed to locate candidate clutter regions and an imaginary Gabor function to detect object edges, in addition to a macro Gabor filter suggested in Ref. 26. In Refs. 28 and 29 the spatial modulation frequency of a real Gabor function is selected to produce one large positive lobe and two smaller negative lobes on either side (Fig. 1), since this selection yields a proven blob detector.<sup>26</sup> The prior work in Refs. 26-29 has been focused on target detection. However, texture segmentation using only real Gabor functions has been detailed in Ref. 5. Portilla et al.<sup>30</sup> use real Gabor functions for extraction of texture features for synthesis of texture by analysis.

### 3 Gabor Filters for Defect Segmentation

In the spatial domain, an image is classically described as a collection of pixels, and in the frequency domain, as a sum of sinusoids of infinite extent. A fabric image can be represented in either the frequency or the spatial domain. Both are relevant in a vision system entailing frequency sampling localized in space. The defect segmentation involves identification of regions with uniform textures in a given image. Appropriate measures of texture are needed in order to decide whether a given region has uniform texture. Defect segmentation in texture requires simultaneous measurements in both the spatial and the frequency domain. Filters with small bandwidths in the frequency domain are more desirable, because they allow us to make finer distinctions among different textures. On the other hand, accurate localization of texture boundaries requires filters that are localized in the spatial domain. However, the effective width of the filter in the spatial domain and its bandwidth in the frequency domain are inversely related. In this sense, Gabor filters achieve the maximum possible joint resolutions in the spatial and frequency domains.<sup>31</sup>

Escofet et al.<sup>13,14</sup> used a combination of real and imaginary Gabor functions to detect defects. In our work, we use only real Gabor functions and ignore imaginary ones. Real Gabor functions act as proven blob detectors,<sup>26</sup> while imaginary Gabor functions act as proven edge detectors.<sup>32</sup> As discussed in Sec. 1.2, Malik and Perona<sup>18</sup> have shown that odd-symmetric mechanisms (imaginary Gabor functions here) are not useful in texture discrimination. While analyzing defects against texture background, we have found that the contribution from imaginary Gabor functions is insignificant but they account for nearly 50% of the total



computation time. Thus, we approximate the impulse response of real and imaginary Gabor functions as Hilbert pairs as used by Shi.<sup>24</sup>

The analytical form of a 2-D real Gabor function in the spatial domain is given by

$$h(x,y)_{mn} = \frac{1}{2\pi|C|^{1/2}} \cos \omega_m^T(\mathbf{p}_n - \mathbf{p}_0) \times \exp \left[ -\frac{1}{2}(\mathbf{p}_n - \mathbf{p}_0)^T C^{-1}(\mathbf{p}_n - \mathbf{p}_0) \right], \quad (6)$$

where  $m$  is the index for scale and  $n$  is an index for orientation. The spatial modulation frequency  $\omega_m$  is only in one direction, since we use Gabor function to detect only height and width. The vector  $\mathbf{p}_0$  shifts the origin of the real Gabor function, so that the output for each input position is shifted by  $\mathbf{p}_0$ . The real Gabor functions for different orientations are obtained by the coordinate transformation  $\tilde{\mathbf{p}}_n = J_n \mathbf{p}_n$  and  $\tilde{\mathbf{p}}_0 = J_n \mathbf{p}_0$ , with<sup>25</sup>

$$J_n = \begin{bmatrix} \cos \theta_n & -\sin \theta_n \\ \sin \theta_n & \cos \theta_n \end{bmatrix}. \quad (7)$$

The angle  $\theta_n$  rotates the real Gabor function for any desired orientation. The parameters  $\omega_m$  and  $\theta_n$  represent the angular frequency and orientation for the  $mn$  channel. The parameters  $\sigma_x$  and  $\sigma_y$ , which define the matrix  $C$ , control the bandwidth of the function.

In this scheme, power-spectrum sampling of an input image at different scales and orientations is performed. The complete set of self-similar Gabor functions used to sample the input image is obtained by rotation (varying  $\theta_n$ ) and scaling (varying  $\omega_m$ ) of the basic Gabor function.

Sixteen Gabor filters that sample the input image in the Fourier domain in a log-polar scheme at four orientations and four scales are shown in Fig. 2.<sup>30</sup> The circles in this figure represent the bandwidth of corresponding Gabor filters at half-peak magnitude. As detailed in Sec. 3.1, four spatial frequencies ( $f_{\max}, f_{\max}/2, f_{\max}/4, f_{\max}/8$ ) shown in Fig. 2 are distributed in octaves, each of which is further rotated in steps of 45 deg (0, 45, 90, 135 deg). Thus a bank of real Gabor functions corresponding to the 16 channels shown in Fig. 2 is used to capture features from the input image. As a compromise between computational load and performance, we have limited the total number of channels to 16. Moreover, there is psychophysical evidence that the human visual system uses a similar number of channels.<sup>1</sup> For an input image  $i(x,y)$  and an  $N \times N$  real Gabor function given by Eq. (6), the filtered image  $I_{mn}(x,y)$  is obtained as

$$I_{mn}(x,y) = h(x,y)_{mn} * i(x,y) = \sum_{k=1}^N \sum_{l=1}^N h_{mn}(k,l) i(x-k,y-l). \quad (8)$$

The above operation requires half the computational time required by that of Ref. 13, in which complex Gabor functions were used, for the calculation of feature vectors.

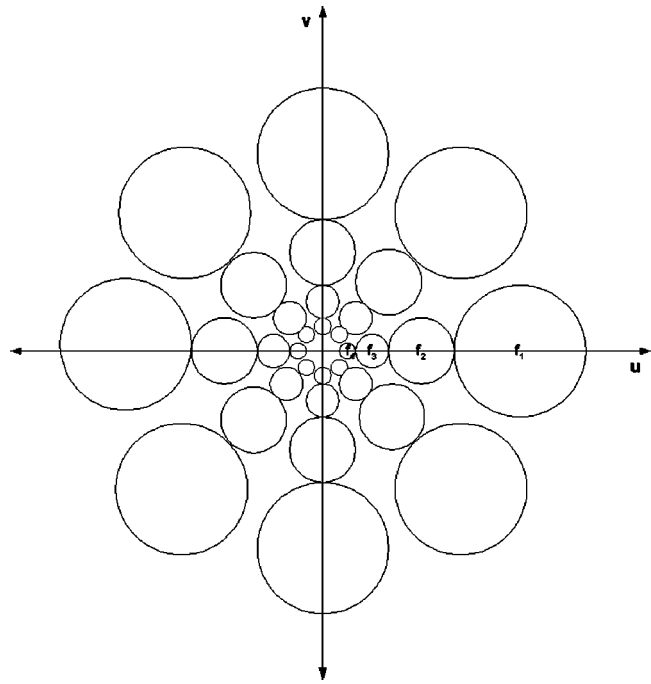


Fig. 2 Frequency-domain representation of 16 Gabor filters on log-polar scale.

An appropriate filter design with small convolution masks allows an efficient implementation of real Gabor functions in the spatial domain. The size of this real-Gabor-function mask is an important parameter. Reliable measurements of texture features call for large mask sizes. On the other hand, large mask size significantly increases the computational load, which is undesirable for on-line inspection. This is because the total number of real operations (additions and multiplications) for each of the sixteen channels is proportional to  $N^2$ , where  $N \times N$  is the mask size. Without any significant degradation in performance we have been able to use  $7 \times 7$  filter masks instead of  $9 \times 9$  as used in Ref. 13. This has resulted in about 40% saving of computational load per frame. Similarly, with some marginal and acceptable degradation in performance, we have been able to use  $5 \times 5$  filter masks, which results in about 70% saving (as compared with  $9 \times 9$  masks) of computational load per frame. This is also supported by the findings by Randen and Husøy<sup>33</sup> that for most textures (in their texture segmentation experiment) a  $5 \times 5$  mask is adequate and only a few texture pairs have required larger mask sizes.

The performance of this algorithm as a function of mask size for various defects is presented in Sec. 6. Every image pixel of fabric under inspection is convolved with the real Gabor function mask (6) to obtain the filtered pixel value. This operation provides us with a set of 16 images from each of the 16 channels, which is used as a feature vector for defect segmentation.

### 3.1 Selection of Parameters

In order to discriminate defects against the textured background, it is necessary to select a set of channel filters that will accomplish the task. Gabor filters act as bandpass filters, and by an appropriate choice of their parameters, they

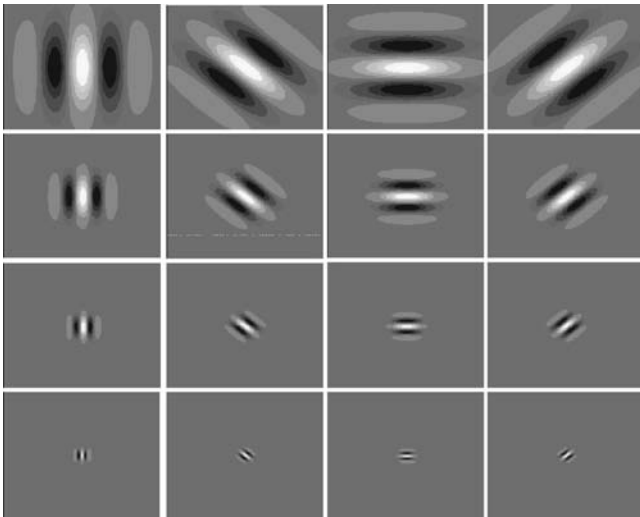


Fig. 3 Sixteen real Gabor functions in spatial domain.

can be tuned to discriminate local fabric defects. The spatial modulation frequency of real Gabor functions is selected so as to produce one large positive lobe and two smaller negative lobes on either side (Fig. 1), since this yields a proven blob detector.<sup>5,26</sup> In Ref. 26 real Gabor functions were shaped to produce blob detectors for object detection, while our work is similar to Ref. 5, in which multiscale and multiorientation real Gabor functions (Fig. 3) are used to capture texture features. We have considered the model with circular symmetry ( $\sigma_x = \sigma_y = \sigma$ ) and a spatial bandwidth proportional to spatial frequency. For Gabor filter defined by Eq. (1), the half-peak-magnitude axial ( $B_a$ ) and orientation ( $B_\theta$ ) bandwidths as shown in Fig. 4 are defined as<sup>4,5</sup>

$$B_a = \log_2 \left[ \frac{\omega_m \sigma + (2 \ln 2)}{\omega_m \sigma - (2 \ln 2)} \right], \quad B_\theta = 2 \tan^{-1} \left[ \frac{(2 \ln 2)}{\omega_m \sigma} \right]. \quad (9)$$

Several experiments have shown that the frequency bandwidth of cells in the visual cortex is about one octave.<sup>34</sup> This justifies the choice for fixing the axial bandwidth as one octave. From Eq. (9) this can be achieved by

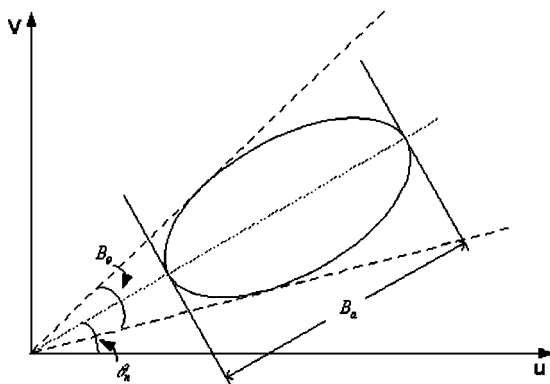


Fig. 4 Frequency-domain parameters of a Gabor filter.

$$\sigma = \frac{3(2 \ln 2)^{1/2}}{\omega_m}. \quad (10)$$

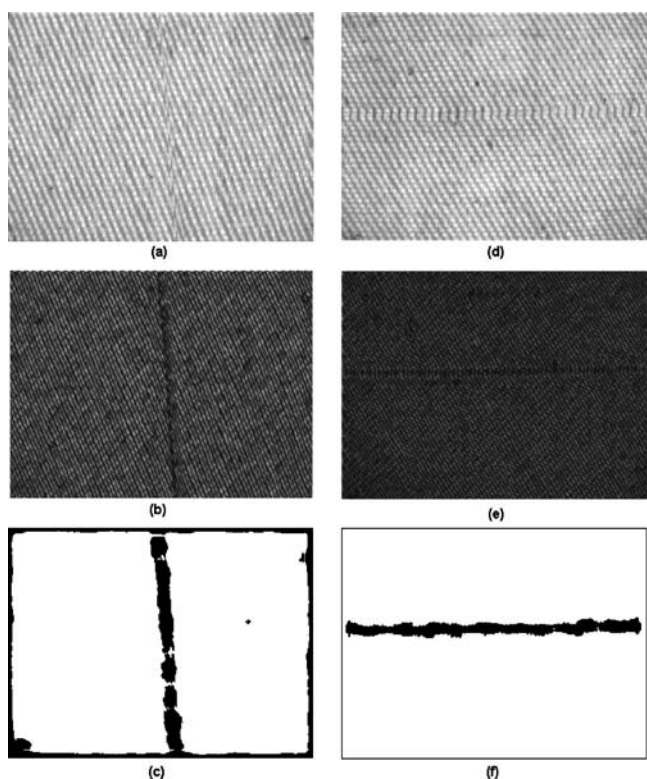
In this way, the radial and angular bandwidths are constant on the log-polar scale, and are equal to one octave and 36.87 deg, respectively.

The spatial-frequency plane of the acquired fabric image is divided into four different orientations (0, 45, 90, and 135 deg). A common method of decomposing the frequency band, motivated by a human-vision model,<sup>34</sup> has been to use an octave-band (dyadic) decomposition.<sup>5,30</sup> Therefore, we divide the radial axis of the spatial-frequency plane into four equal octave bands (centered at  $f_1, f_2, f_3,$  and  $f_4$ ). In a bandwidth of one octave, the spatial frequency increases by a factor of two. The highest central frequency for each direction is located at the Nyquist frequency to avoid ringing and noise.<sup>35</sup> The resultant filter bank performs log-polar sampling of acquired fabric image.

The width of the thinnest yarn of this fabric, expressed in terms of the number of pixels, determines the maximum frequency of interest. Let  $f_1$  be this maximum frequency, also denoted by  $f_{\max}$ . This choice of radial frequency guarantees that the passband of the filter with highest radial frequency (i.e.,  $f_{\max}$ ) falls inside the image array. Thus we select the next radial frequency of the Gabor filter,  $f_2$ , at the next lower octave channel (one octave away), i.e.,  $f_2 = f_{\max}/2$ . Similarly,  $f_m = f_{\max} 2^{1-m}$  ( $m = 1, 2, 3, 4$ ). For a particular fabric, if it is found that the thinnest yarn occupies 12 pixels, then we choose  $f_{\max} = 1/12$  cycle/pixel. With a maximum frequency of this order, we expect to detect few defects whose sizes are approximately that of one yarn. The larger defects can be located with filters of lower frequencies. That is, the greater the extent of the defect, the lower is the filter frequency needed to detect it. The contour locations of 16 Gabor filters along with their center frequencies are shown in Fig. 2.

### 3.2 Supervised Segmentation of Defects

If *a priori* knowledge regarding the orientation and size of local fabric defects is made available to the process, it can be regarded as supervised defect segmentation. In such cases, segmentation can be performed using just one appropriately tuned Gabor filter instead of the bank of Gabor filters discussed in last subsection. Figure 5 clearly depicts successful supervised defect segmentation using a Gabor filter. From the visual examination of the fabric sample in Fig. 5(a) it can be observed that the defect is approximately one yarn wide and is located at about 90 deg in the spatial plane. Since one yarn in this image occupies 24 pixels (approximately), a Gabor filter located at  $f = 1/24$  cycle/pixel is chosen. A  $15 \times 15$  (*ad hoc*) Gabor filter mask with  $f = 1/24, \theta = 90$  deg, and half-peak bandwidth of one octave [Eq. (10)] was found to be appropriate for attenuation of background and accentuation of defects. Further, a  $9 \times 9$  median filter was used to suppress the speckle-like noise from the filtered image, and the resulting image was thresholded to obtain a binary image of the segmented defect as shown in Fig. 5(c). The thresholding value is obtained from Eq. (18), as detailed in Sec. 5.2. The median filtering attenuates irrelevant spectral features that do not contribute to



**Fig. 5** Supervised defect segmentation: (a) and (d), test samples; (b) and (e), corresponding Gabor-filtered images; (c) and (f), binarized filtered images with segmented defects.

an efficient segmentation using thresholding. Figure 5(d) shows another fabric sample, in which the observed defect is approximately 2 yarns wide and is oriented at 90 deg in the spatial plane. A similar processing of this image yields the segmented defect shown in Fig. 5(f). For the fabric sample images shown in Fig. 5, defect segmentation can also be achieved with a convolution mask smaller than  $15 \times 15$  (ad hoc), but with some degradation in performance: the segmented defect is not so clear as in Fig. 5.

A supervised approach will have limitations as compared to the flexibility of unsupervised approach. However, in many industrial inspection applications it may be assumed that the orientation and resolution of defects are fixed. Supervised defect segmentation can be economically implemented on general-purpose hardware for inspection of defects of known sizes in a known direction. However, unsupervised defect segmentation is a more critical task and is more suitable for on-line detection of local defects in the textile industry. This problem is discussed in the next section.

#### 4 Online Defect Detection

Any online fabric defect detection system must be capable of integrating defects captured at different orientations and resolution levels of the Gabor filter. The desired output is a binary image of local defects in the fabric. The desired procedure must be robust, automatic, and applicable to fabric with different structures. Vibration-free images of the fabric under test are acquired using backlighting.

The block diagram of this procedure is shown in Fig. 6 and discussed here. The procedure starts with an acquired image of fabric with defects  $i(x,y)$ . The acquired images exhibit artifacts of brightness gradient due to nonhomogeneous lighting. These artifacts are corrected by subtracting a reference image from the acquired images. This reference image of plain white paper is acquired under the same illumination condition. The Gabor functions described in Sec. 3.1 are applied to this corrected image and the 16 filtered images F1 to F16 are obtained. As detailed in the next subsection, a nonlinear operator on these images generates corresponding blob descriptors T1 to T16. A similar set of operations on a defect-free fabric sample (reference) is used to obtain blob descriptors (R1 to R16) for a reference fabric sample. As shown in Fig. 6, the mean and standard deviations from each of the sixteen blob descriptors (R1 to R16) are computed at the beginning and stored before the fabric inspection. This set of means and standard deviations is utilized to generate a feature difference array (TDD1, . . . , TDD16). Section 5 explains the operation of the sensitivity control, which is used to monitor noise and generates 16 images (S1 to S16). As shown in Fig. 6 and detailed in Sec. 5.1, these images are subjected to image fusion, and a set of four images (N1 to N4) corresponding to each of the four scales ( $m$ ) is obtained. These images are in turn combined (Sec. 5.1) to a single image output  $N$  so as to further reduce false alarms. This image  $N$  is subjected to calibration, which is detailed in Sec. 5.2, and the result is a final image of segmented defects (if any) in the fabric sample under inspection.

#### 4.1 Extraction of Texture Blob Descriptors

For reliable defect segmentation, it is necessary to have a set of feature vectors that can characterize the texture. These texture features form the basis for defect segmentation. Now our objective is to find a transform function that can enhance the changes in each of the 16 images F1 to F16, which may correspond to a defect, in such a way that a thresholding operation can segment the defect from the textured background. Talukdar and Casasent<sup>36</sup> have developed a linear maximum representation and discrimination feature (MRDF) that has been shown to outperform standard linear feature extraction techniques such as the Fisher linear discriminant, Fukunaga-Koontz (FK) transform, and Karhunen-Loève (KL) transform for segmentation of image data. However, linear transforms are only optimal when the data are Gaussian and symmetrically distributed about the mean. They are not necessarily the best for image data that are not characterized by Gaussian probability density functions. Therefore, nonlinear transforms are necessary. Several nonlinear transforms, such as independent-component analysis and nonlinear principal-component analysis, are iterative (and may fail to converge to the globally optimal solution) and have limitations on the rank of decision surfaces. These shortcomings of nonlinear transforms have been addressed by Talukdar and Casasent<sup>36,37</sup> with the use of a nonlinear MRDF. They have derived a closed-form solution that automatically finds the best nonlinear transform, which is a polynomial mapping of the input, without any increased computational complexity as compared with linear approaches. Applications of the nonlinear MRDF have been successfully demonstrated for estimation of the

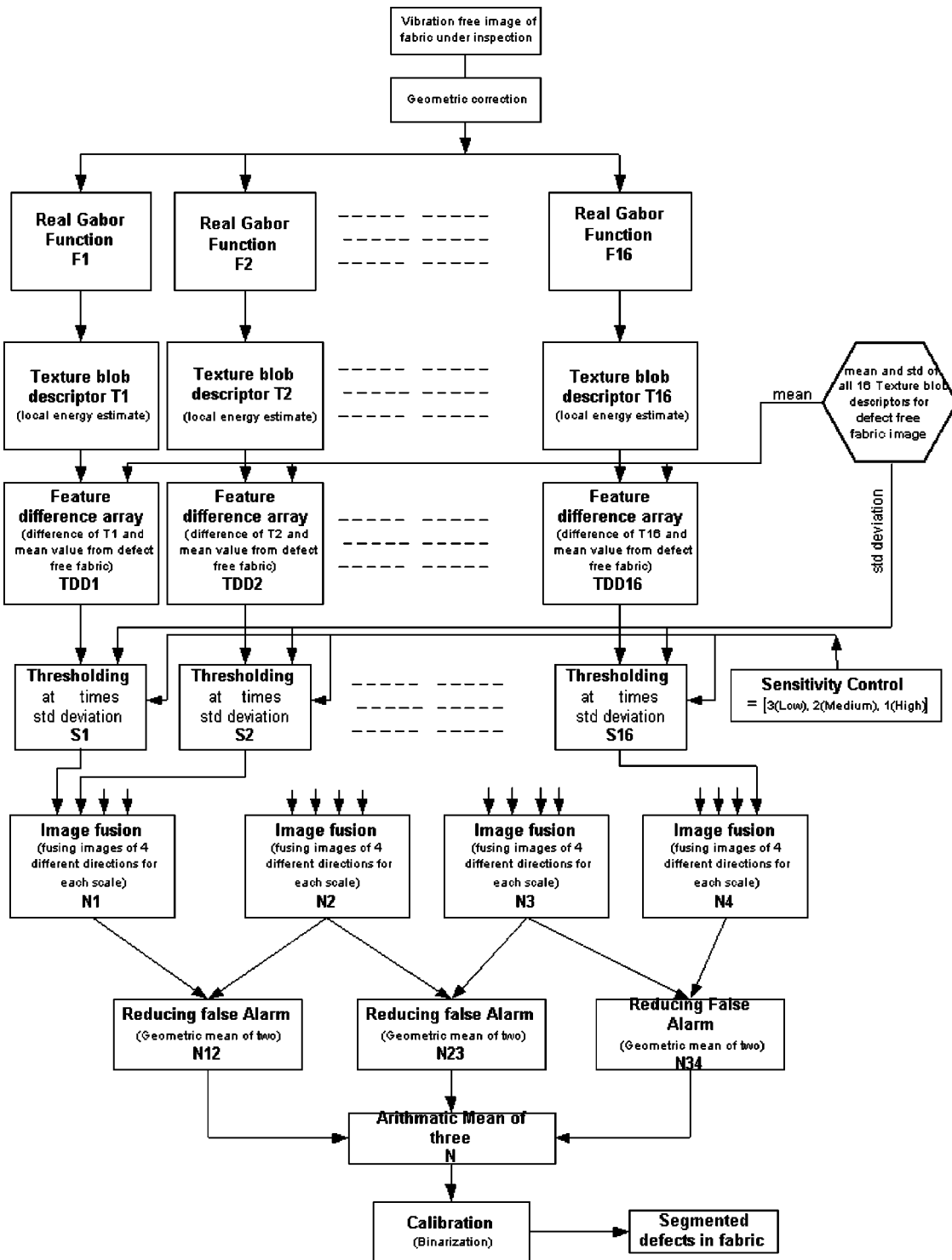


Fig. 6 Schematic diagram of on-line defect segmentation setup.

pose of an unknown image<sup>38</sup> and x-ray inspection of pistachio-nut images.<sup>36</sup> For our defect detection problem, only discrimination is necessary (not representation), and therefore a *discriminatory* nonlinear MRDF can potentially be used to enhance the performance of the 16 filtered images (F1 to F16). Primarily because of the need to minimize computational requirements in a real-time environment, we have not investigated nonlinear MRDF for this work.

In our work, we assume that the energy distribution in the frequency domain identifies a texture or a defect. Based on this assumption, we have used a local energy function to calculate texture blob descriptors. The objective of the local energy function is to estimate the energy in the sixteen filtered images in the local region. We prefer a nonlinear local energy function, as it is computationally efficient. Some of the commonly used nonlinear functions<sup>33</sup> are the



magnitude  $|x|$ , the square  $|x|^2$ , and the rectified sigmoid  $|\tanh \beta x|$ . Unser and Eden<sup>39</sup> have proposed and analyzed several combinations of the first and the second nonlinearity for texture segmentation. They have concluded that the nonlinear function that squares along with logarithmic normalization is the best combination. However, they did not test the rectified sigmoid, which is also an important nonlinear function similar to the sigmoidal activation function commonly used in artificial neural networks. In contrast to the rectified sigmoid, the magnitude and squaring functions do not require any tuning parameters. However, in the current application, this may be a disadvantage, as the parameter  $\beta$  might be tuned. Furthermore, the nonlinearity of the sigmoid is supported by human visual models for texture discrimination, as discussed in Sec. 1.2. Therefore, the local energy function proposed by Jain and Furrokhnia<sup>5</sup> is appropriate for our application. T1 is obtained as

$$T1(x,y) = |f[F1(x,y)]|, \quad f(t) = \tanh \beta t. \quad (11)$$

Similarly, a new set of 16 images, described here as texture blob descriptors (T1 to T16) are obtained. The parameter  $\beta$ , which gives the saturation characteristics of this function, depends on the dynamic range of gray levels in the acquired images.<sup>33</sup> Empirically, this is fixed at 0.3 to give a fast-saturating, thresholdlike function.

The application of the local energy function transforms the sinusoidal modulations in the filtered image to square modulations. Therefore, this operation can be interpreted as blob detection.<sup>5</sup> Since the nonlinear function used for each of the 16 filtered images F1 to F16 is odd-symmetric, the texture blob descriptors T1 to T16 are accompanied by both dark and light blobs.

Individual blobs in T1 to T16 are identified and are assigned to defect or defect-free texture. Texture descriptors for the reference (defect-free) fabric sample (R1 to R16) are obtained in a similar manner. For each of these 16 texture descriptors corresponding to the defect-free sample, we obtain the mean and standard deviation. This set of means and standard deviations is used for further characterization of each pixel from the texture blob descriptors (T1 to T16).

## 5 Statistical Defect Segmentation

Texture features characterized by the enhanced local gray-level statistical distribution (11) are asymptotically uniform for defect-free fabric (in sufficiently large image areas). Given a prototype of the fabric under test, defect segmentation requires identification of a proper distance among such distributions. Classical approaches based on estimation of some statistical moments (e.g., mean value, standard deviation) or other statistical parameters<sup>40</sup> allow very quick characterization of image pixels. On the other hand, methods based on higher-order statistics (e.g., co-occurrence matrix, run-length metrics, statistical feature matrices) provide more information but are highly demanding in both computational and memory requirements. Primarily to avoid a computational bottleneck in the online fabric defect detection system, we have processed the texture blob descriptors T1 to T16 by using first-order statistical analysis.

The set of texture blob descriptors T1 to T16 forms the basis for defect segmentation. From these descriptors, a

comparison for each pixel of a defective fabric with the corresponding pixel of defect-free fabric is made. If the difference is small, the probability of a pixel corresponding to the defect-free sample is high. If the difference is large, it is highly probable that this pixel corresponds to a defect. For each of the texture blob descriptors, the texture descriptor difference (magnitude) TDD can be written as

$$TDD(x,y) = |T(x,y) - m_{\text{defect-free}}|, \quad (12)$$

where  $m_{\text{defect-free}}$  is the mean of the corresponding texture blob descriptors for defect-free fabric (R1 to R16). Next, the standard thresholding operation to reduce the noise is performed. For each of the pixels in TDD, we find corresponding pixels in  $S$ :

$$S(x,y) = \begin{cases} TDD(x,y) & \text{if } TDD(x,y) \geq \tau \text{ sd,} \\ 0 & \text{otherwise.} \end{cases} \quad (13)$$

The thresholding is thus proportional to the standard deviation (sd). This standard deviation is calculated for defect-free fabric from each of its sixteen texture blob descriptors, R1 to R16. The magnitude of the coefficient  $\tau$  depends on the sensitivity as fixed by the user.

The motivation behind the introduction of *sensitivity control* in our algorithm is twofold. When we increase the size of fabric in the image frame in an attempt to increase the performance, the mean gray-level variation in the resulting image tends to be uniform. Now the thresholding limit has to be much smaller (highly sensitive) to discriminate the defective pixels. We have found that a thresholding limit equal to one standard deviation increases the sensitivity and is most suitable for large-area images with uniform mean gray level. Second, when the fabric size in the image frame is small so that gray-level variations are considerably nonuniform, a thresholding limit of twice the standard deviation (medium sensitivity) is appropriate. On the other hand, when the number of yarn impurities in the fabric is high, the sensitivity has to be kept low ( $\tau=3$ ) to discriminate the defects against the noisy background. Thus the sensitivity control largely depends on the fabric texture and image acquisition conditions, and must be adjusted accordingly.

Now we have obtained 16 threshold feature difference images (S1 to S16), and our next task is to preserve the pixels in each of these images that correspond to a defect.

### 5.1 Image Fusion

Evaluating the reliability of texture difference pixels from different channels is crucial when the texture difference images reveal inconsistencies. A number of fusion algorithms have been developed<sup>29,41-43</sup> and used to reduce the false-alarm rate while maintaining high probability of detection. One approach that has been successfully used for target detection involves deliberate generation of unwanted output (clutter) followed by its subtraction from the detection output. Examples of this approach have been detailed in Refs. 41 and 42: the hit-miss transform, involving the intersection fusion of the outputs from hit and miss filters,<sup>41</sup> and the morphological wavelet transform, involving the subtraction of clutter-map output from analog clutter-reduction output.<sup>42</sup> Casasent and Smokelin<sup>27</sup> use imaginary



and clutter Gabor-filter outputs to reduce false-alarm rates in detection output generated by using only real Gabor functions. Casasent and Ye<sup>43</sup> have performed qualitative and quantitative analysis of binary and analog fusion algorithms. A binary fusion algorithm uses the logical AND of outputs obtained from several detection outputs. However, such an algorithm does not weight the difference in confidence levels between different detection outputs, and so analog-and-hierarchical fusion algorithms have been shown to produce better results, i.e., lower false-alarm rates. As detailed in Ref. 43, an analog-and-hierarchical fusion algorithm uses a mapping function to convert different detection outputs into a common range. This step is motivated by the fuzzification function used in fuzzy logic, and is also used in a fusion algorithm based on Bernoulli's rule of combination,<sup>44</sup> though with a different membership function.

The main function of the image fusion module in our work is to attenuate background pixels and accentuate defective pixels from four directions. Bernoulli's rule of combination, which is often used in image fusion,<sup>12,44</sup> is extended here for integrating images from four directions. For each of the four images at every scale  $m$ , the following mapping function is used to convert pixel values into a common output range from 0 to 1:

$$O_{mn}(x,y) = \frac{S_{mn}(x,y) - \min[S_{mn}(x,y)]}{\max[S_{mn}(x,y)] - \min[S_{mn}(x,y)]}, \quad (14)$$

where the image input  $S_{mn}(x,y)$  is the same as  $S(x,y)$  from Eq. (13), but has been subscripted with indices for scale  $m$  and orientation  $n$  as in Eq. (6). Thus for Eq. (14) the feature difference images S1 to S4 are denoted by  $S_{11}(x,y)$  to  $S_{14}(x,y)$  to indicate that the images have originated from real Gabor functions at scale  $m=1$  and orientation  $n = 1,2,3,4$ , (0, 45, 90, and 135 deg).

Next, a fused output  $N_m(x,y)$  is generated for every scale  $m$ , by fusion of the normalized images (14) from four directions:

$$\begin{aligned} N_m(x,y) = & \sum_{n=1}^4 O_{mn}(x,y) - [O_{m1}(x,y)O_{m2}(x,y) \\ & + O_{m2}(x,y)O_{m3}(x,y) + O_{m3}(x,y)O_{m4}(x,y) \\ & + O_{m4}(x,y)O_{m1}(x,y) + O_{m4}(x,y)O_{m2}(x,y) \\ & + O_{m3}(x,y)O_{m1}(x,y)]. \end{aligned} \quad (15)$$

Thus for every scale  $m=1,2,3,4$ , we obtain one fused image output  $N_m(x,y)$  from the four images  $S_{m1}(x,y)$ ,  $S_{m2}(x,y)$ ,  $S_{m3}(x,y)$ , and  $S_{m4}(x,y)$  using Eqs. (14) and (15). As detailed in Ref. 12, the fused outputs tend to follow one of the inputs closely if the other inputs possess low values. On the other hand, an input with very high values tends to dominate the outputs, regardless of their values. Thus the pixels from the defects captured in any of the four orientations will dominate in the final fused image for each scale. Thus the fusion suppresses the noise and combines sixteen images (S1 to S16) into four images (N1 to N4).

Due attention to reducing false alarms should be given when the information gathered from the four resolution lev-

els (N1 to N4) is combined. It is reasonable to assume that a defect will appear on at least two adjacent resolution levels; otherwise it is highly unlikely that it is a defect. As reported by Escofet et al.,<sup>13</sup> this consideration has been found to reduce the false-alarm rate while preserving most of the defective areas. This is ensured by computing geometric means of every adjacent level. For example, N12 is computed as

$$N12(x,y) = [N1(x,y)N2(x,y)]^{1/2}. \quad (16)$$

Next, a set of three images N12, N23, and N34 are obtained. An arithmetic mean will combine defects captured by them. This image  $N(x,y)$  in essence contains contribution from all sixteen texture descriptors (T1 to T16):

$$N(x,y) = \frac{1}{3}[N12(x,y) + N23(x,y) + N34(x,y)]. \quad (17)$$

## 5.2 Calibration

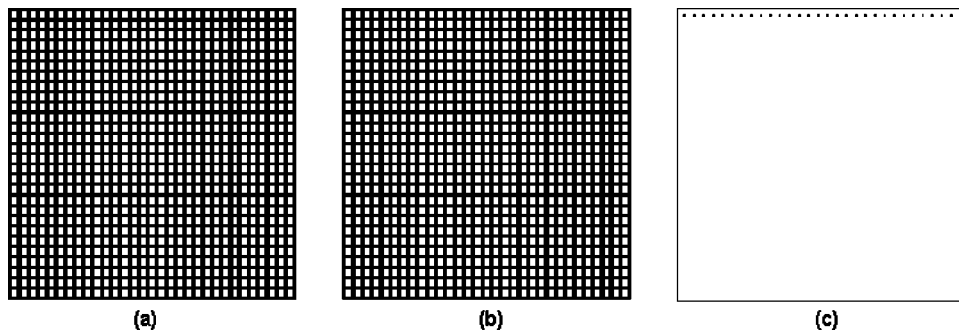
Finally, this last image is subjected to thresholding. A thresholding value is selected such that values below it are considered as belonging to regular texture in the fabric, and values above as belonging to defects. This value is suitably obtained by calibration of the system at the beginning of the operation. For calibration, a fabric sample without any defects and yarn impurities is used. With the use of this reference image,  $N_{\text{defect-free}}(x,y)$  is obtained from Eq. (17). The threshold value  $\eta_{\text{th}}$  is obtained by

$$\eta_{\text{th}} = \max_{x,y \in W} \{N_{\text{defect-free}}(x,y)\}, \quad (18)$$

where  $W$  is a window centered at the image. Thus the threshold value  $\eta_{\text{th}}$  is the maximum amplitude of gray levels, within the window  $W$ , in the image  $N_{\text{defect-free}}(x,y)$  obtained from the reference image. The window size is chosen to avoid the effects of border distortion.<sup>22</sup> It is obtained by removing 10 pixels (ad hoc) from each side of the image  $N_{\text{defect-free}}(x,y)$ . This choice depends on the mask size of the real Gabor functions; for a  $7 \times 7$  mask, at least seven pixels from the border have to be ignored. The magnitude of  $\eta_{\text{th}}$  is such that isolated noisy pixels in  $N(x,y)$  are completely isolated in the output binary image. Binarization based on this threshold limit helps to suppress the noise, although this operation also suppresses some of the defects captured at different orientations and frequencies. In prior work,<sup>13</sup> the threshold value was based on the mean and standard deviation of the final image [ $N_{\text{defect-free}}(x,y)$  here]. As stated in Ref. 13, and from our experiments, this threshold value generates large noise in the output and requires an opening operation with a convolution mask (typically  $3 \times 3$ ) to eliminate the noise. With the use of the threshold value suggested in Eq. (18), the opening operation is not needed and this results in reduction of computational load, which is critical for real-time implementation of this algorithm.

## 6 Results

We have tested this defect segmentation algorithm on both synthetic and real test fabric images. The reason for testing this algorithm on synthetic images was to ensure that the



**Fig. 7** Synthetic test fabrics for evaluation: (a) with defect, (b) without defect, (c) with segmented defect.

algorithm is able to discriminate difficult fabric defects, which humans can discriminate preattentively. The image acquisition subsystem developed at the Industrial Automation Laboratory at The University of Hong Kong has been used to capture gray-level images of defective test fabrics. All images used here were acquired using backlighting. Samples of the most commonly occurring fabric defects (mixed filling, waste, mispicks, kinky filling, misreed, etc.) were gathered from a loom, and their gray-level images were used to evaluate the algorithm described. Some of these results are reproduced here.

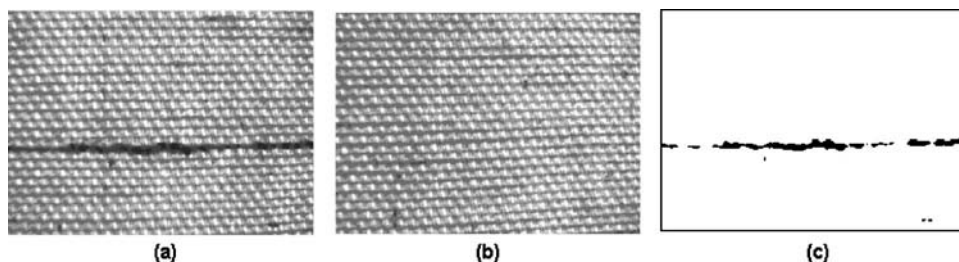
Figures 7(a) and 7(b) show synthetic binary images of the test and the reference fabric, respectively. The second row of the test image has two more pixels than the other rows. This simulated defect of thick yarn can be seen segmented in the final image [Fig. 7(c)]. Figures 8(a) and 8(b) show real fabric images, with defects segmented in Fig. 8(c). The border effect in the segmented image is found to be reasonably localized, and the border distortion can be ignored.

Accurate segmentation of defects is limited by poor yarn quality. Impurities that are naturally present in fabric yarns tend to obscure more subtle defects. This effect can be seen in Figs. 9 and 10. The fabric image in Fig. 9 has defects along with large yarn impurities. The defects are segmented at three different sensitivities: low, high, and medium. As seen from results in Fig. 10, low sensitivity helps to suppress the yarn impurities but at the expense of losing some pixels from the defect. However, when segmenting defects from fabric with large impurities, the sensitivity has to be reduced to avoid yarn impurities appearing as defects in output.

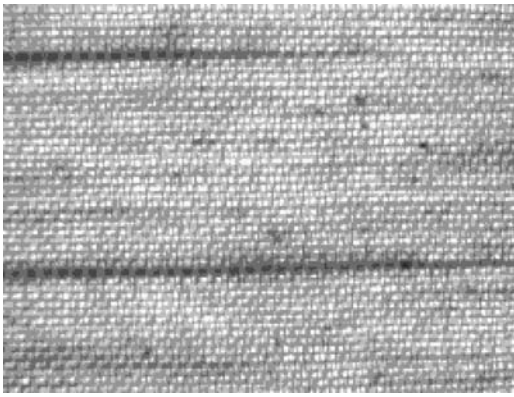
The image acquisition subsystem was adjusted to acquire large images while fabric is in motion with a velocity

of 30 cm/s. The acquired images were digitized in  $385 \times 287$  pixels, with 8-bit resolution (256 gray levels). Nine images of fabric with defects were chosen to have large characteristic variability in composition and structure. It is assumed that these sample images are representative of fabric defects in the textile industry. All these images cover 10-cm width and 7.5-cm height of actual fabric. Figures 11, 12, and 13 illustrate the defect segmentation achieved with the proposed algorithm. Due to the increase in the area of fabric per frame, we have increased the sensitivity ( $\tau = 2$ ). Figure 13(b) shows a fabric sample with defects that are visible only with difficulty. The appearing defects only alter the spatial arrangement of neighboring pixels and not the mean gray level. This change is registered by real Gabor functions and enhanced by the local energy function, and segmentation is successfully achieved as shown in Fig. 13(e).

The lack of appropriate quantitative measures for the goodness of segmentation makes it very difficult to evaluate and compare different defect detection methods. However, a simple criterion that has been used in many texture segmentation algorithms<sup>5,15,33</sup> is the percentage of misclassified pixels. In defect segmentation problem, it is defective pixels that are of interest, and therefore the percentage of all defective pixels misclassified is what is reported here (Table 1). From this table some general observations can be made. With the increase in sensitivity from low to high, the percentage of misclassified pixels increases for every image sample (for a  $7 \times 7$  mask), except for the sample of Fig. 11(b). Another effect on performance due to the variation in mask size can be observed. At the medium sensitivity, as the mask size is increased from  $5 \times 5$  to  $9 \times 9$ , for every image sample there is a decrease in percentage of misclas-

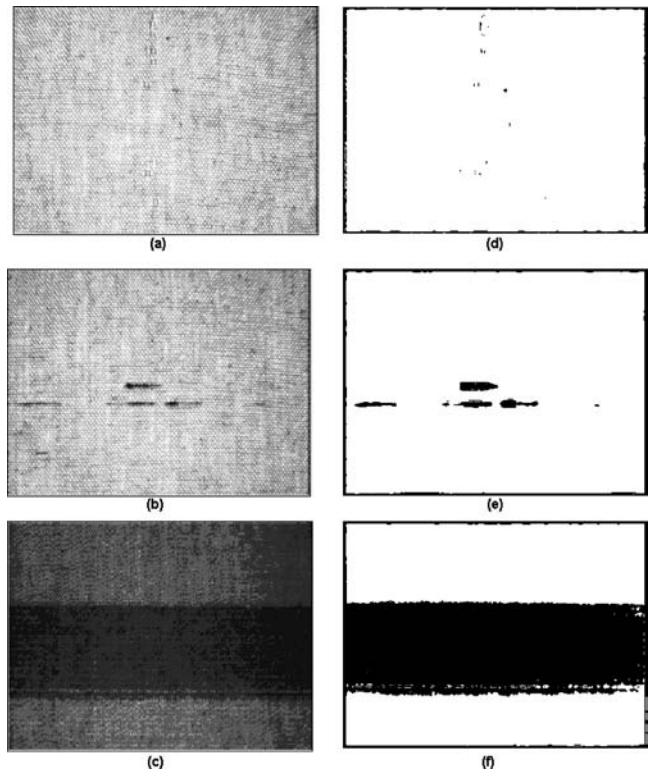


**Fig. 8** Fabric sample for test: (a) with defect, (b) without defects, (c) with segmented defect.

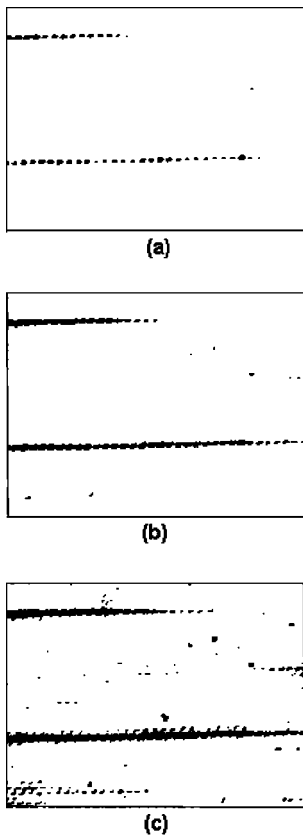


**Fig. 9** Fabric sample with defects and yarn impurities.

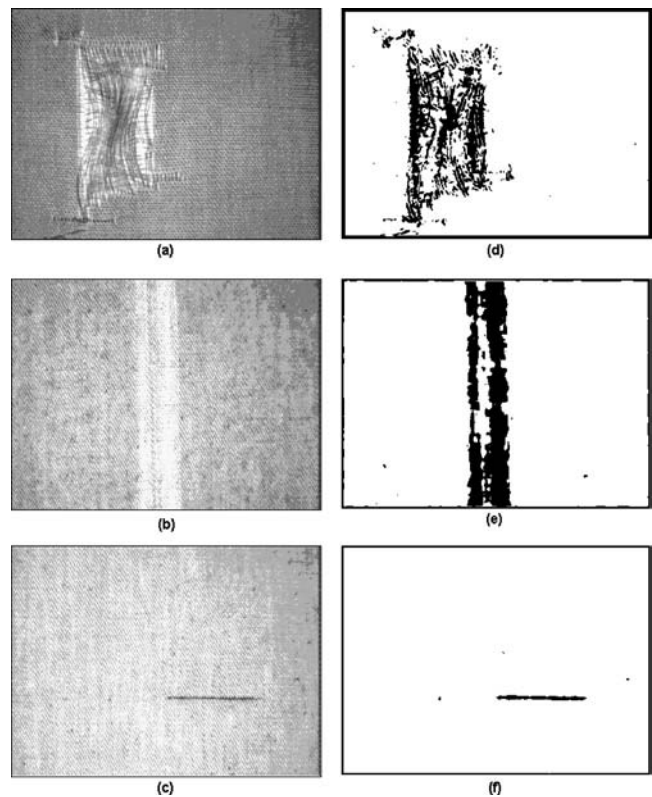
sified pixels, with the exception of the samples in Figs. 11(b) and 12(b). The image in Fig. 13(b) registered the largest increase in performance (9.26 times) with the increase in mask size from  $5 \times 5$  to  $9 \times 9$  at the medium sensitivity. For the images shown in Figs. 12, 13, and 14



**Fig. 11** Various defective-fabric test samples (a),(b),(c), and binarized segmented defects (d),(e),(f).



**Fig. 10** Segmented defects at (a) low, (b) medium, and (c) high sensitivity.



**Fig. 12** Various defective-fabric test samples (a),(b),(c), and segmented defects (d),(e),(f).

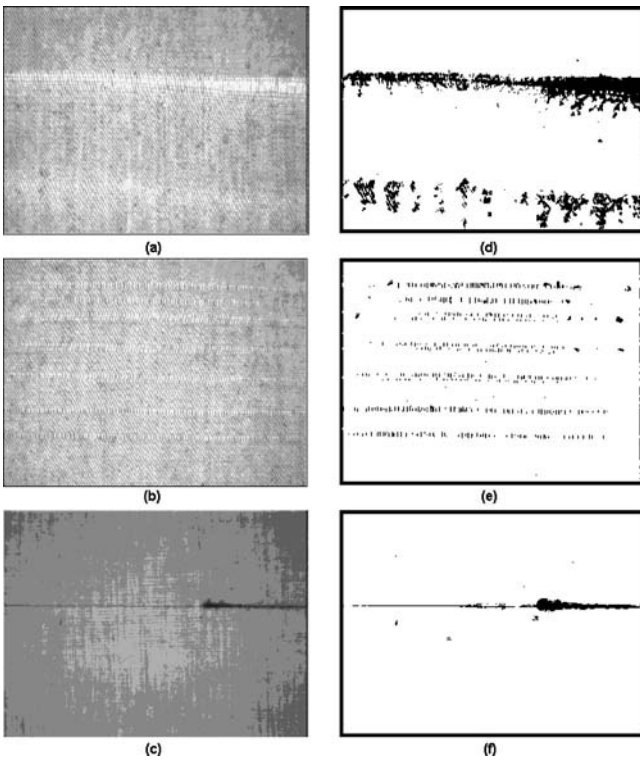


Fig. 13 Various defective-fabric test samples (a),(b),(c), and binarized segmented defects (d),(e),(f).

the percentage of misclassified pixels is below 2%, except for one sample in Fig. 12(a). The conclusion that can be drawn from Table 1 is that there is an overall reduction in percentage of pixels misclassified as defects with increase

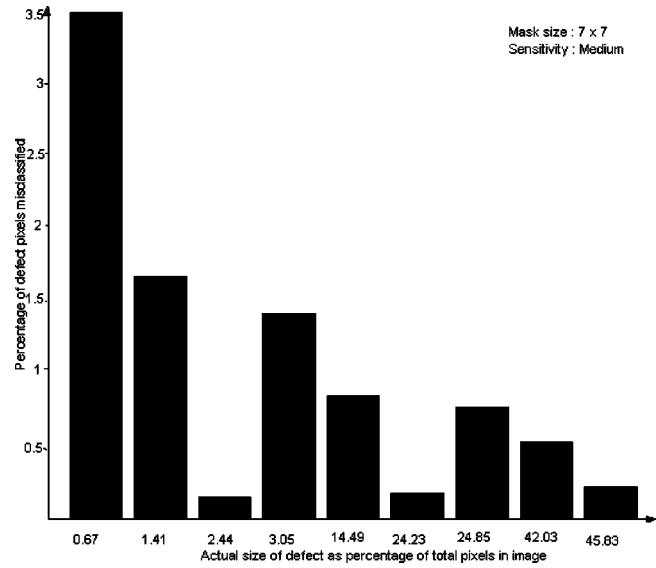


Fig. 14 Performance as a function of defect size.

in mask size and with decrease in sensitivity, as intuitively expected.

Another observation may be made in Fig. 14, where the percentage of pixels occupied by fabric defects in the acquired image is plotted against percentage of misclassified pixels from the defect. We observe the general increase in false alarms with increase in size of defects in the frame.

The criterion used in Table 1 (percentage of misclassified pixels) does not accurately reflect the ability of the algorithm to segment the defect. A low percentage of mis-

Table 1 Percentage of total defective pixels misclassified.

Mask size	Percentage misclassified								
	Fig. 11(a)			Fig. 11(b)			Fig. 11(c)		
	L	M	H	L	M	H	L	M	H
5	5.98	42.11	39.66	0.23	0.10	0.14	0.02	0.03	0.04
7	0.61	34.92	37.83	0.33	0.14	0.28	0.00	0.22	0.05
9	2.30	14.40	30.02	0.42	1.63	0.70	0.00	0.03	0.02
Mask size	Fig. 12(a)			Fig. 12(b)			Fig. 12(c)		
	L	M	H	L	M	H	L	M	H
	5	0.01	0.43	2.80	0.13	0.29	0.21	1.89	3.92
7	0.04	0.17	2.57	0.06	0.84	0.25	1.60	1.67	2.69
9	0.03	0.08	0.75	0.13	0.33	0.04	1.02	1.81	3.34
Mask size	Fig. 13(a)			Fig. 13(b)			Fig. 13(c)		
	L	M	H	L	M	H	L	M	H
	5	0.68	1.28	2.14	0.10	1.76	1.79	2.58	5.30
7	0.08	0.53	3.42	0.11	0.76	9.34	1.07	1.41	3.13
9	0.96	0.56	3.81	0.01	0.19	4.87	0.40	1.58	3.63



**Table 2** Performance analysis for defect sample in Fig. 11(a). P1: defective pixels identified; P2: actual defective pixels (obtained by visual examination); P3: noise (pixels appearing as defects in defect-free region). Total pixels in image (window): 97455. Sensitivity: L (low), M (medium), H (high).

Class	Total pixels								
	5×5			7×7			9×9		
	L	M	H	L	M	H	L	M	H
P1	44	100	92	61	124	104	66	104	93
P2	653	653	653	653	653	653	653	653	653
P3	39	275	259	4	228	247	15	94	196

classified pixels does not necessarily mean good defect segmentation, unless it is accompanied by a large number of pixels showing a defect in the defective region. Therefore, some of the results from Table 1 are elaborated in Tables 2, 3, and 4 to show complete statistics. We find a general increase in defective pixels (P1) with increase in sensitivity. However, this is accompanied by an increase in pixels appearing outside the defective region (P3), i.e., noise. With increase in mask size, output defective pixels (P1) also increase. But this increase is very small and requires about 50% (70%) more computation when the mask size is increased from  $5 \times 5$  to  $7 \times 7$  ( $9 \times 9$ ). Based on these experiments, we preferred to use  $7 \times 7$  masks without any significant change in output, and  $5 \times 5$  masks with marginal compromise for defects of smaller sizes.

In the textile industry, the majority of weaving defects occur either in the direction of motion (warp direction) or perpendicular to it (pick direction). Air-jet looms are most popular, and their predominant defects are end-outs (missing or broken warp yarns), slubs (excess yarn), and mispicks (missing or broken pick yarns). All these defects have been successfully segmented as illustrated in the previous figures. This algorithm has been evaluated with some of the less commonly occurring defects, which are caused by machine malfunction, such as holes [Fig. 11(a)] and oil spots [Fig. 11(b)]. It can also potentially be used for opaque textured materials (timber, plastic etc.) illuminated with front lighting.

The results achieved prove that this algorithm is robust, scalable, and computationally efficient and offers a high detection rate. One of its limitations is that it requires calibration whenever the fabric type (texture) or image acqui-

sition conditions are varied. This calibration is proportional to the width of the thinnest yarn in the fabric, which is a function of frequency-domain parameters. An extension of this technique that allows automatic calibration would therefore be highly desirable.

## 7 Discussion

The choice of elements of the proposed algorithm is mainly guided by two factors, computational complexity and performance. In a real-time environment, success of any algorithm depends on computational complexity, and therefore the computational requirements are stringent. The feature vectors should be combined in such a way as to reduce the probability of false alarm while maintaining high probability of detection.

Some elements of the proposed algorithm are based on a review of prior work by Escofet et al.<sup>13</sup> While they successfully segmented fabric defects using complex Gabor filters, their approach had certain shortcomings. First, the contribution from the imaginary Gabor function was very small, and it could not justify the additional 50% of computations required for calculating its feature vector. Second, the computational requirements for the multiresolution pyramid (low-pass residual images) was very high, and therefore a local energy estimate proved to be a better recourse in terms of computation and performance. Third,  $9 \times 9$  masks used were not optimal, and we can justify the choice of  $7 \times 7$  (or even  $5 \times 5$ ) masks through quantification of performance. Fourth, the previous work<sup>13</sup> did not allow for yarn impurities inherently present in fabric or for the physical

**Table 3** Performance analysis for defect sample in Fig. 12(a). P1: defective pixels identified; P2: actual defective pixels (obtained by visual examination); P3: noise (pixels appearing as defects in defect-free region). Total pixels in image (window): 97455. Sensitivity: L (low), M (medium), H (high).

Class	Total pixels								
	5×5			7×7			9×9		
	L	M	H	L	M	H	L	M	H
P1	5890	9780	11629	5442	9675	12811	5362	8541	12649
P2	23609	23609	23609	23609	23609	23609	23609	23609	23609
P3	2	101	660	10	39	607	7	18	177

**Table 4** Performance analysis for defect sample in Fig. 13(a). P1: defective pixels identified; P2: actual defective pixels (obtained by visual examination); P3: noise (pixels appearing as defects in defect-free region). Total pixels in image (window): 97455. Sensitivity: L (low), M (medium), H (high).

Class	Total pixels								
	5×5			7×7			9×9		
	L	M	H	L	M	H	L	M	H
P1	4898	16987	19660	3822	15934	22548	2167	10001	17952
P2	40957	40957	40957	40957	40957	40957	40957	40957	40957
P3	28	526	876	32	216	1402	21	56	1684

size of the fabric in the image used for processing.<sup>†</sup> Lastly, the two methods suggested by the authors for binarization generate unacceptably large noise, and the morphological operations suggested to remove this noise are computationally expensive. The threshold value computed by Eq. (18) is simple and highly successful in removing isolated noisy pixels, as shown from the results in this paper.

## 8 Conclusions

In this paper, a multichannel filtering approach for the detection of local fabric defects has been demonstrated. Image fusion has been successfully utilized in combining features from different channels. We have shown that the performance of the algorithm is significantly improved by varying the sensitivity in the presence of yarn impurities and low spatial sampling rate. Furthermore, considerable computational saving has been achieved, which is attributed to the use of real Gabor functions, smaller filter masks, and a local energy function.

The filtering and feature extraction operations (15) in this algorithm account for most of the required computations. However, these operations can be performed in parallel. Therefore, on-line implementation of this algorithm should utilize a high-performance DSP processor, such as the TMS320C80. Recently, real Gabor functions have been implemented using cellular neural networks (CNNs).<sup>24,45</sup> The advantage of CNNs is that they can be implemented using analog VLSI alongside photosensors (CMOS or CCD arrays) integrated with camera hardware. Thus Gabor-filtered outputs can be read off the chip directly. This approach will drastically relieve the computational bottleneck and make the use of DSP processors redundant.

The proposed algorithm is an alternative to currently accepted methods that do not take advantage of the spatial arrangement of gray levels in neighboring pixels, and instead rely on differences in their mean gray level. The algorithm has been tested on real fabric with success, and results are shown in this paper. The algorithm can potentially be used for defect segmentation in any textured material such as timber, paper, plastic, or carpet.

<sup>†</sup>The fabric size of 256×256 pixels, claimed in Ref. 13 is not consistent with the results shown, as the aspect ratio of the images are not 1:1. Information regarding the physical size of the fabric in the image and mode of acquisition (front or back lighting) is also missing.

## References

1. J. G. Daugman, "Spatial visual channels in the Fourier plane," *Vision Res.* **24**, 891–910 (1984).
2. F. W. Campbell and J. G. Robson, "Application of Fourier analysis to visibility of gratings," *J. Physiol. (London)* **197**, 551–566 (1968).
3. R. L. De Valois, D. G. Albrecht, and L. G. Thorell, "Spatial-frequency selectivity of cells in macaque visual cortex," *Vision Res.* **22**, 545–559 (1982).
4. A. C. Bovik, M. Clark, and W. S. Geisler, "Multichannel texture analysis using localized spatial filters," *IEEE Trans. Pattern Anal. Mach. Intell.* **12**, 55–73 (1990).
5. A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using Gabor filters," *Pattern Recogn.* **24**, 1167–1169 (1991).
6. I.-S. Tsai, Ch.-H. Lin, and J.-J. Lin, "Applying an artificial neural network to pattern recognition," *Text. Res. J.* **65**, 123–130 (1995).
7. E. J. Wood, "Applying Fourier and associated transforms to pattern characterization in textiles," *Text. Res. J.* **60**, 212–220 (1990).
8. M. Unser and F. Ade, "Feature extraction and decision procedure for automated inspection of textured materials," *Pattern Recogn. Lett.* **2**, 181–191 (1984).
9. X. F. Zhang and R. R. Bresee, "Fabric defect detection and classification using image analysis," *Text. Res. J.* **65**(1), 1–9 (1995).
10. F. S. Cohen, Z. Fau, and S. Attali, "Automated inspection of textile fabrics using textured models," *IEEE Trans. Pattern Anal. Mach. Intell.* **13**, 803–808 (1991).
11. C. H. Chan and G. Pang, "Fabric defect detection by Fourier analysis," *IEEE Trans. Ind. Appl.* **36**, 1267–1276 (2000).
12. H. Sari-Sarraf and J. S. Goddard, "Vision systems for on-loom fabric inspection," *IEEE Trans. Ind. Appl.* **35**, 1252–1259 (1999).
13. J. Escofet, R. Navarro, M. S. Millan, and J. Pladelloreans, "Detection of local defects in textiles webs using Gabor filters," *Opt. Eng.* **37**(8), 2297–2307 (1998).
14. J. Escofet, R. Navarro, M. S. Millan, and J. Pladelloreans, "Detection of local defects in textiles webs using Gabor filters," *Proc. SPIE* **2785**, 163–170 (1996).
15. A. K. Jain and K. Karu, "Learning texture discrimination masks," *IEEE Trans. Pattern Anal. Mach. Intell.* **18**, 195–205 (1996).
16. J. Beck, A. Sutter, and R. Ivry, "Spatial frequency channels and perceptual grouping in texture segmentation," *Comput. Vis. Graph. Image Process.* **37**, 299–325 (1987).
17. T. Caelli, "Three processing characteristics of visual texture segmentation," *Spatial Vis.* **1**, 19–30 (1985).
18. J. Malik and P. Perona, "Pretentive texture discrimination with early vision mechanisms," *J. Opt. Soc. Am. A* **7**, 923–932 (1994).
19. B. Julesz, "Textons, the elements of texture perception and their interactions," *Nature (London)* **290**, 91–97 (1981).
20. A. Sutter, J. Beck, and N. Graham, "Contrast and spatial variables in texture segregation: testing a simple spatial-frequency channels model," *Percept. Psychophys.* **46**, 312–332 (1989).
21. N. Graham, J. Beck, and A. Sutter, "Two nonlinearities in texture segregation," *Invest. Ophthalmol. Visual Sci.* **30**, 161 (1989).
22. R. Shapley and C. Enroth-Cugell, "Visual adaptation and retinal gain controls," *Prog. Retinal Res.* **4**, pp. 263–347 (1984).
23. P. Brodatz, *Textures: A Photographic Album for Artists and Designers*, Dover, New York (1966).
24. B. E. Shi, "Gabor-type filtering with cellular neural networks," in *Proc. IEEE International Symp. on Circuits and Systems, ISCAS'96*, Vol. 3, pp. 558–561 (1996).
25. D. Weber and D. Casasent, "Quadratic Gabor filter for object detection," in *Proc. 1997 South African Symp. on Communications and Signal Processing*, pp. 69–74 (1997).
26. D. Casasent and J. S. Smokelin, "Neural net design of macro Gabor

- wavelet filters for distortion-invariant object detection in clutter," *Opt. Eng.* **33**(7), 2264–2271 (1994).
27. D. Casasent and J. S. Smokelin, "Real, imaginary, and clutter Gabor filter fusion for detection with reduced false alarms," *Opt. Eng.* **33**(7), 2255–2363 (1994).
  28. D. Weber and D. Casasent, "Fusion and optimized Gabor filter design for object detection," *Proc. SPIE* **2588**, 662–675 (1995).
  29. D. Casasent, "Gabor wavelet filters and fusion for distortion-invariant multi-class object detection," *Proc. SPIE* **2491**, 430–440 (1995).
  30. J. Portilla, R. Navarro, O. Nestares, and A. Taberero, "Texture synthesis-by-analysis method based on a multiscale early-vision model," *Opt. Eng.* **35**(8), 2403–2417 (1998).
  31. J. Daugman, "Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters," *J. Opt. Soc. Am. A* **2**(7), 1160–1169 (1985).
  32. R. Malhotra, K. R. Namuduru, and N. Ranganathan, "Gabor filter-based edge detection," *Pattern Recogn.* **25**, 1479–1494 (1992).
  33. T. Randen and J. H. Husøy, "Filtering for texture classification: a comparative study," *IEEE Trans. Pattern Anal. Mach. Intell.* **21**, 291–310 (1999).
  34. D. A. Pollen and S. F. Ronner, "Visual cortical neurons as localized spatial frequency filters," *IEEE Trans. Syst. Man Cybern.* **13**, 907–916 (1983).
  35. R. Rodriguez-Sánchez, J. A. Garcia, J. Fdez-Valdivia, and X. R. Fdez-Vidal, "The RGTF representational model: a system for the automatically learned partitioning of 'visual patterns' in digital images," *IEEE Trans. Pattern Anal. Mach. Intell.* **21**, 1044–1073 (1999).
  36. A. Talukder and D. Casasent, "General methodology for simultaneous representation and discrimination of multiple object classes," *Opt. Eng.* **37**(3), 904–917 (1998).
  37. A. Talukder and D. Casasent, "Nonlinear features for classification and pose estimation of machined parts from single views," *Proc. SPIE* **3522**, 16–27 (1998).
  38. A. Talukder and D. Casasent, "Pose estimation and transformation of faces," *Proc. SPIE* **3522**, 84–95 (1998).
  39. M. Unser and M. Eden, "Nonlinear operators for improving texture segmentation based on features extracted by spatial filtering," *IEEE Trans. Syst. Man Cybern.* **20**, 804–815 (1990).
  40. S. Oe, "Texture segmentation method by using two-dimensional AR model and Kullback information," *Pattern Recogn.* **26**(2), 237–244 (1993).
  41. D. Casasent, A. Ye, J. S. Smokelin, and R. Schafer, "Optical correlation filter fusion for object detection," *Opt. Eng.* **33**(6), 1757–1767 (1994).
  42. A. Ye and D. Casasent, "Morphological and wavelet transforms for object detection and image processing," *Appl. Opt.* **33**(35), 8226–8239 (1994).
  43. D. Casasent and A. Ye, "Detection filters and algorithm fusion for ATR," *IEEE Trans. Image Process.* **6**, 114–125 (1997).
  44. M. Abdulhafour, J. Goddard, and M. A. Abidi, "Non-deterministic approaches in data fusion—a review," *Proc. SPIE* **1393**, 596–610 (1990).

45. K. R. Crouse and L. O. Chua, "Methods for image processing and pattern formation in cellular neural networks: a tutorial," *IEEE Trans. Circuits Syst., I: Fundam. Theory Appl.* **42**, 583–601 (1995).



**Ajay Kumar** received a BE degree in electronics engineering from Ravishankar University, India, in 1990, and an ME degree in electronics and communication engineering from Delhi University, India, in 1992. He worked at I.I.T. (Kanpur) as a junior research fellow and at I.I.T. (Delhi) as a senior scientific officer before joining Indian Railways. He joined the Indian Railway Service of Signal Engineers (IRSSE) in 1993 and worked as an assistant signal and telecom engineer. He worked as a project engineer at I.I.T. (Kanpur) during 1996–1997, and as an assistant professor at NIST, Berhampur, India, from September 1997 to September 1998. He joined the University of Hong Kong in November 1998 as a research associate, where he is now working for his PhD in the Department of Electrical and Electronics Engineering. His research interests include pattern recognition with emphasis on defect detection using wavelets, filter banks, general texture analysis, and neural networks. Ajay Kumar is a student member of the IEEE and an executive council member of the HKU IEEE student branch.



**Grantham Pang** obtained his PhD degree from the University of Cambridge in 1986. He was with the Department of Electrical and Computer Engineering, University of Waterloo, Canada, from 1986 to 1996 and joined the Department of Electrical and Electronic Engineering at the University of Hong Kong in 1996. Since 1988, he has published more than 100 technical papers and has authored or coauthored five books. His research interests include machine vision for defect detection, optical communications, expert systems for control-system design, intelligent control, and intelligent transportation systems. Dr. Pang was the organizing chair of the 1996 IEEE Symposium on Computer-Aided Control System Design. He is an editor of the *International Journal of Intelligent Control and Systems, Machine Intelligence and Robotic Control*, and *Control and Computers*. He is a Chartered Electrical Engineer and a member of the IEE, HKIE, and IEEE.