

<http://ansinet.com/itj>

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSI*net*

Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Textural Fabric Defect Detection using Adaptive Quantized Gray-level Co-occurrence Matrix and Support Vector Description Data

Bi Mingde, Sun Zhigang and Li Yesong

Key Laboratory of Ministry of Education for Image Processing and Intelligent Control,
Department of Control Science and Engineering, Huazhong University of Science and Technology,
1037 Luoyu Road, Hubei, 430074, Wuhan, China

Abstract: A new defect detection algorithm base on Support Vector Data Description (SVDD) is proposed. A fabric texture model is built on the gray-level histogram of textural fabric image. Two Gray-level Co-occurrence Matrix (GLCM) features are used to characterize the fabric texture. And an adaptive quantization scheme base on the texture mode is proposed to reduce the size of GLCM and reduce the computational complexity of feature extraction. Besides, two new features are proposed to characterize the continuous property of the fabric defects. The SVDD classifier is used as a detector for defect detection. Experimental results of real fabric defects are provided to validate the effectiveness and robustness of the proposed detection algorithm. And a prototyped detection system is built to evaluate the real-time performance of the detection algorithm.

Key words: Fabric defect, fabric texture model, gray-level co-occurrence matrix, supports vector description data, real time

INTRODUCTION

Fabric defect detection is one of the most important procedures effecting the manufacturing efficiency and quality in fabric industry. So far, most factories accomplish this study by human vision which makes the quality of fabric devoid of consistency and reliability. It is found that even a highly trained inspector can only detect about 70% of the defects at a speed of 15-20 m min⁻¹ (Sari-Sarraf and Goddard, 1999). As the development of image processing technology, many methods base on texture analysis were proposed for defect detection. Several statistical methods using texture features, such as fractal dimension (Conci and Proenca, 1998; Bu *et al.*, 2009), morphological feature (Mallick-Goswami and Datta, 2000; Chandra *et al.*, 2010) and co-occurrence matrix (Latif-Amet *et al.*, 2000; Lin, 2010) were proposed to discriminate defects from normal texture. Cohen *et al.* (1991) used the Gauss Markov Random Field (GMRF) to model the texture pattern of non-defective fabric image and the defect is detected by the rejection of the model using hypothesis testing theory. Because of the high degree of periodicity of the fabric texture, the spectral approaches are also used for fabric defect detection. Chan and Pang (2000) used Fourier transform for this task which turned out to be only suitable for global defect

detection because of its poor local resolution in the frequency domain. Defect detection methods with multiresolution decomposition using a bank of Gabor filters were proposed by Bodnarova *et al.* (2002) and Kumar and Pang (2002). As Gabor filter banks will lead to redundant features at different scales, the orthogonal wavelet transform (Nadhim, 2006) are also used for texture characterization (Loum *et al.*, 2007), classification (Raju *et al.*, 2008) and fabric defect detection (Yang *et al.*, 2002; Mingde and Zhigang, 2011).

The defect detection can be considered as a one-class classification problem. Ohanian and Dubes (1992) and Randen and Husoy (1999) provide some occasions that, in the viewpoint of texture classification, GLCM outperforms other features such as Gabor filters feature, fractal feature, MRF feature. However GLCM is still degraded for its high computational complexity. In this study, an adaptive quantization method is proposed base on the texture model of mixture of two Gauss distribution. The quantization method reduces the gray-level from 256 levels to several levels which greatly reduces the computation complexity of GLCM features. The SVDD classifier is used as a detector for defect detection. Several machine learning based texture classification methods have been proposed using neural network (Kumar, 2003; Jianli and Baoqi, 2007;

Corresponding Author: Sun Zhigang, Key Laboratory of Ministry of Education for Image Processing and Intelligent Control, Department of Control Science and Engineering, Huazhong University of Science and Technology, 1037 Luoyu Road, Hubei, 430074, Wuhan, China Tel: +86 027 87544420 Fax: +86 027 87544420

Chandra *et al.*, 2010; Nagarajan and Balasubramanie, 2008; Mahi and Izabatene, 2011), Support Vector Machine (SVM) (Kumar and Shen, 2002; Chu *et al.*, 2011) and SVDD (Bu *et al.*, 2009, 2010). The neural network and support vector based methods considered the defect detection as a two-class classification problem which required both defective and non-defective samples for training. However, the requirement of large quantities of defective samples is usually not desirable for online inspection in industrial occasions. Similar to SVM, the SVDD classifier is also a kernel function based classifier which does not suffer the problem of local minima. However, it is a one-class classifier which requires only the non-defective sample for training, thus it is adopted in our algorithm.

FABRIC TEXTURE MODEL

Figure 1a and b show two samples of non-defective texture of plain and twill fabrics, respectively. It can be see

that the plain and twill texture are, respectively, made up of one and two gray tones which is also indicated in their histograms in Fig. 1c and d by solid lines. The fabric texture model is built on the histogram of gray-level image of fabric texture. The histogram of the plain fabric texture tends to obey the Gauss distribution and the histogram of the twill fabric texture tends to obey a mixture of two Gauss distributions. So, the Probability Density Function (PDF) of gray-level in the plain fabric image can be modeled as:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{\sigma^2}} \tag{1}$$

where, μ and σ are the mean and standard deviation of the Gauss distribution, respectively. And the PDF of gray-level in the twill fabric image can be modeled as:

$$f(x) = \frac{P_1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(x-\mu_1)^2}{\sigma_1^2}} + \frac{P_2}{\sqrt{2\pi}\sigma_2} e^{-\frac{(x-\mu_2)^2}{\sigma_2^2}} \tag{2}$$

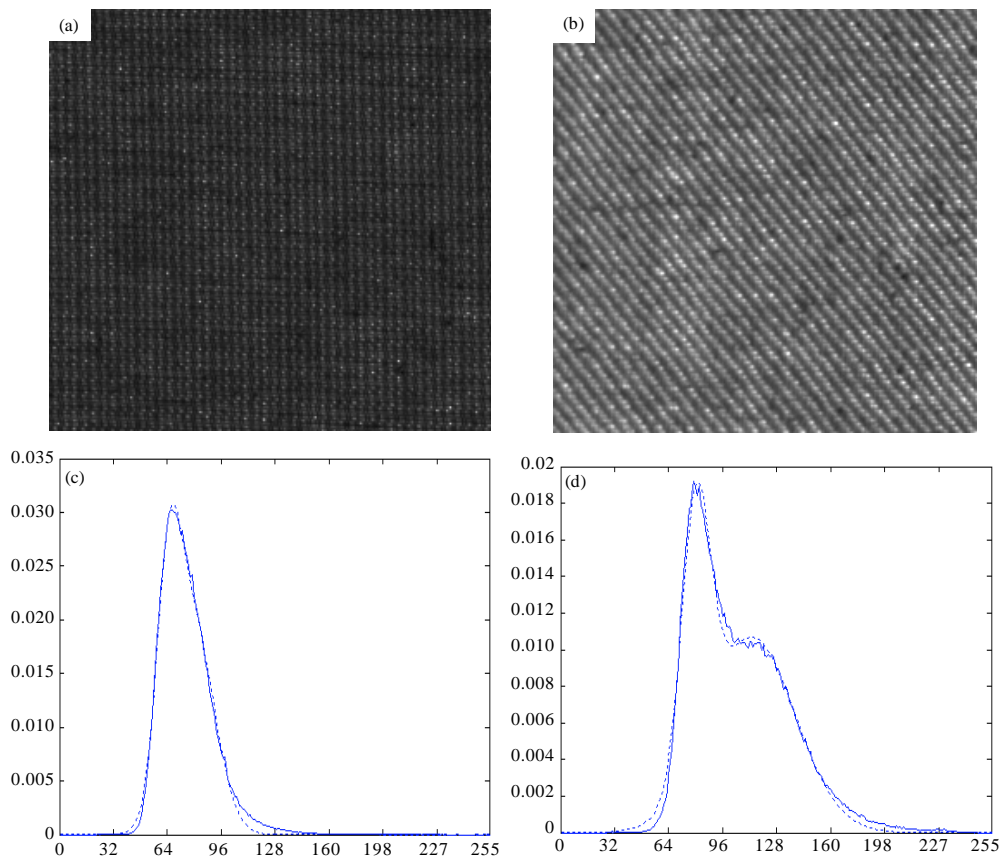


Fig. 1(a-d): Non-defective samples of (a) plain, (b) twill fabrics and (c, d) corresponding histogram and fitting results

where, $P_1+P_2 = 1$ and $\mu_1 \leq \mu_2$. P_1, μ_1, σ_1 and P_2, μ_2, σ_2 are the proportions, means and standard deviations of two Gaussians in the mixture, respectively.

In order to build a consistent model for both plain and twill fabrics, only Eq. 2 is used and Eq. 1 is considered as a particular case of Eq. 2 with $P_1 = 1, P_2 = 0, \mu_1 = \mu_2$ and $\sigma_1 = \sigma_2$. And parameters of mixed Gauss distribution in Eq. 2 can be estimated by curve fitting. The fitting results of both plain and twill fabric image histograms are illustrated in Fig. 1c and d with dash lines, respectively.

GRAY-LEVEL CO-OCCURRENCE MATRIX AND ADAPTIVE QUANTIZATION

Spatial gray-level co-occurrence estimates image properties related to second-order statistics. It turns out that GLCM has become one of the most well-known texture features and is widely used for texture characterization (Shang *et al.*, 2011; Sheng *et al.*, 2010). A co-occurrence matrix is a square matrix whose elements correspond to the relative frequency of occurrence of pairs of gray level values of pixels separated by a certain distance in a given direction. The $G \times G$ gray-level co-occurrence matrix P_d for a displacement vector $d = (dx, dy)$ is defined as follows. The entry (I, j) of P_d is the number of occurrences of the pair of gray levels I and j which are a distance d apart. Formally, it is given as:

$$P_d(I, j) = |\{(x_1, y_1), (x_2, y_2) : I(x_1, y_1) = I, I(x_2, y_2) = j\}| \tag{3}$$

where, I denotes an image of size $U \times V$ with G gray values $(x_1, y_1), (x_2, y_2), U \times V, (x_2, y_2) = (x_1+dx, y_1+dy)$ and $|\cdot|$ is the cardinality of a set. The co-occurrence matrix reveals certain properties about the spatial distribution of the gray levels in the texture image.

Generally, for a gray-level image whose pixel is represented by an 8-bit integer (i.e., $G = 256$), its GLCM is

a matrix of size 256×256 . Extracting features from such a large matrix is quite computational expensive. An intuitive way to reduce the size of GLCM is to reduce the gray level G of original image by equal quantization (Latif-Amet *et al.*, 2000). However, during the equal quantization if the gray level of the defect is close to the normal texture, it is probably quantized to the same level as the normal texture which makes it hard to distinguish. Figure 2a and b illustrate a defective fabric image and its equal quantization result with 16 levels, respectively. The quantized image is histogram equalized for visual convenience. It can be seen that the defect in Fig. 2b becomes ambiguous and hard to detect.

In this study, an adaptive quantization method is proposed based on the fabric texture model elaborated in earlier. As shown in Fig. 1, the histogram of gray-level of normal texture obeys a mixture Gauss distribution which contains two Gaussians. Due to the randomization of texture intensity, to which Gaussian a single pixel belongs and the distance between its gray-level and the center of its belonging Gaussian which is considered as Gaussian Central Distance (GCD), is of more importance than in which exact gray-level the pixel is. The pixel in the image is quantized by examining its GCD. Prior to the calculation of GCD, we should decide which of the two Gaussians the pixel belongs to. We consider that if $g < \mu_1$, then it belongs to the first Gaussian and if $g > \mu_2$ then it belongs to the second Gaussian. For the gray-level $\mu_1 < g < \mu_2$, the following probability criterion is used to determine its belonging Gaussian:

$$P(\omega_i | g) = \frac{P(g | \omega_i)P(\omega_i)}{\sum_{i=1}^2 P(g | \omega_i)P(\omega_i)} \tag{4}$$

where, $\omega_i, i = 1$ and 2 , denote first and second Gaussian, respectively, $P(\omega_i)$ is the prior probability of each

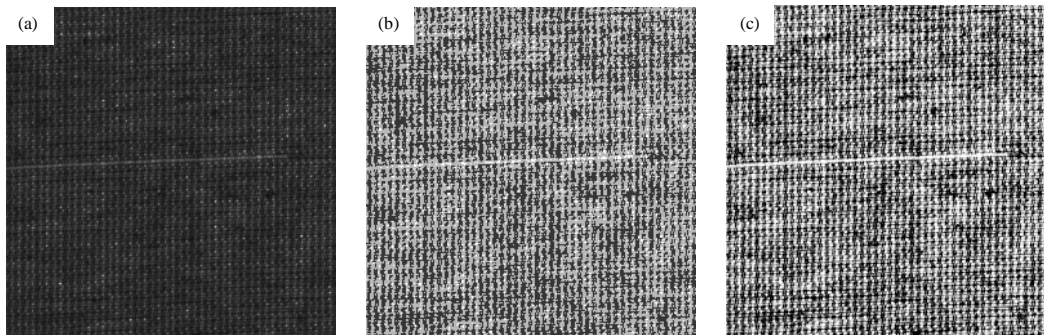


Fig. 2(a-c): Comparison of quantization methods

Gaussian, i.e., $P(\omega_1) = P_1$ and $P(\omega_2) = P_2$ in Eq. 2. If $P(\omega_1|g) > P(\omega_2|g)$, then g is considered to belong to first Gaussian, otherwise it belongs to the second Gaussian. This classification method is known as Bayes classification which minimizes the misclassification rate. Formally the adaptive quantization method consists of following steps:

- Step 1:** Divide both Gaussians into several non-overlapping intervals separately formulated as $[0, \mu_1 - (0.5 + N)\lambda\sigma_1]$, $[\mu_1 + (0.5 + n)\lambda\sigma_1, \mu_1 + (0.5 + n + 1)\lambda\sigma_1]$ and $[\mu_2 + (0.5 + N)\lambda\sigma_2, 255]$, where $n = -(N + 1), -N, \dots, N - 1$. λ is a constant determining the width of the interval and each Gaussian is divided into $2N + 3$ intervals
- Step 2:** For each gray-level g from 0 to 255, determine which Gaussian it belongs to and which interval of the Gaussian it belongs to, then vote for that interval. The intervals which are never voted are discarded and the completely voted intervals are preserved. Here, the complete voted interval refers to the interval that all the gray-levels within it only vote for itself
- Step 3:** For the rest intervals which are not completely voted. If any of them overlaps with any other non-complete voted intervals, then merger them together to form a new interval. If the merged interval overlaps with any complete voted interval, then truncate the overlapping areas of the merged interval to make sure that all the intervals are non-overlapping. Finally, index all the intervals with zero-based numbers and make sure that larger gray-levels correspond to larger index. Each interval forms a new quantized level.

Figure 3 gives an example of adaptive quantization procedure. Figure 3a and b illustrate the division results of two Gaussians of a mixture distribution with $\lambda = 1$, $N = 2$. Each Gaussian is divided into 7 intervals. The final quantization result is illustrated in Fig. 3c. The intervals $[0, 54)$, $[54, 64)$, $[64, 74)$ and $[74, 84)$ of the first Gaussian and $[99, 125)$, $[126, 151)$, $[151, 176)$ and $[176, 255]$ of the second Gaussian are completely voted intervals. Thus each of them directly forms a quantized level in Fig. 3c. The intervals $[84, 94)$ of the first Gaussian and $[74, 99)$ of the second Gaussian are non-completely voted intervals and they overlap with each other. So they are merged into $[74, 99]$. As $[74, 84)$ is a completely voted interval, the merged interval is truncated into $[84, 94)$ which forms a new quantized level in Fig. 3c. Finally, 256 gray-levels are quantized to 9 levels.

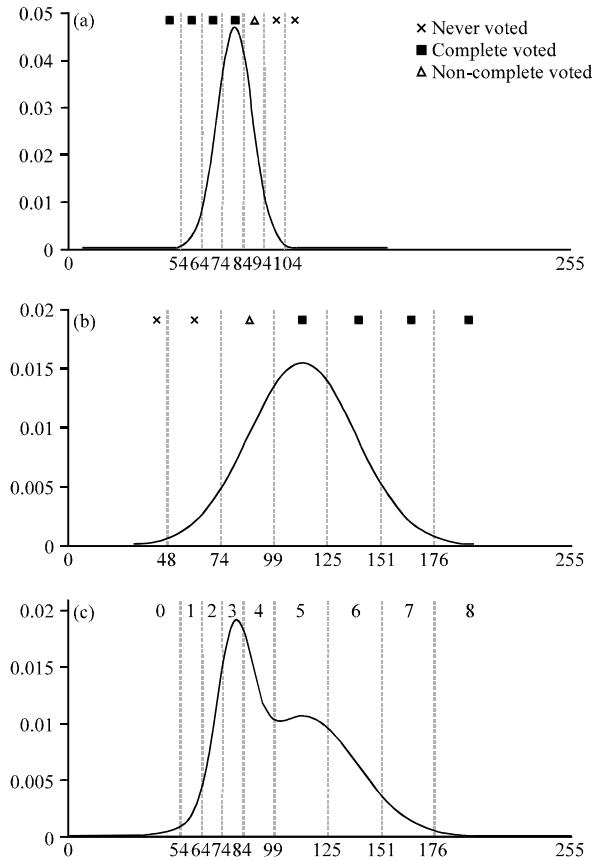


Fig. 3(a-c): Example of adaptive quantization procedure

The pixel is quantized according to its GCD which is measured by the standard deviation of the belonging Gaussian rather than its gray-level. Pixels belonging to the same Gaussian with similar GCD rather than similar gray-levels tend to be quantized to the same level. Particularly, quantized level 0 and $L - 1$ are two special levels, where L denotes the total number of quantized levels. It can be seen that quantized level 0 is out of the confidence interval of the first Gaussian which means, in the sense of hypothesis testing, the pixels within this quantized level do not belong to the normal texture. As the pixels in this level are much darker than the normal texture, they are called darkness exceptional pixels. Similarly the pixels in quantized level $L - 1$ are called lightness exceptional pixels. The advantage of adaptive quantization method is that it can preserve most useful information of the original image while reduce the 256 gray-levels to several quantized levels. Besides, as the quantization is based on GCD of the texture model, the defective region does not obey the texture model and thus contains more lightness or darkness exceptional pixels, so the quantization method can also emphasize the presence

of the defects. Figure 2c shows the result of adaptive quantization of Fig. 2a with only 9 quantized levels which is even better than Fig. 2b with 16 levels.

SUPPORT VECTOR DESCRIPTION DATA (SVDD)

Support vector data description is a powerful kernel method that has been commonly used for novelty detection (Tax and Duin, 2004). It provides a solution to the one-class classification problem without any negative sample in training. By mapping the data into a higher dimensional space, the objective of SVDD is to find, in this space, a spherically shaped boundary around the training dataset such that the sphere can enclose as many samples as possible while having minimum volume. The sphere is characterized by its center c and radius $R > 0$. Let $\{x_i\}$, $i = 1, 2, \dots, M$, be a set of training examples, $x_i \in R^d$, d being the dimension of the input space and M is the number of training samples. The minimization of the sphere volume is achieved by minimizing its square radius R^2 . To allow for the presence of outliers, slack variables ξ_i 's are introduced so that the problem of constructing an optimal separating hypersphere is converted to the following optimization problem:

$$\begin{aligned} \min R^2 + \frac{1}{\sqrt{M}} \sum_{i=1}^M \xi_i \\ \text{s.t. } \|\Phi(x_i) - c\|^2 \leq R^2 + \xi_i, \xi_i \geq 0, i = 1, \dots, M \end{aligned} \tag{5}$$

where, ξ_i accounts for possible errors, v is a user-provided parameter specifying an upper bound on the fraction of outliers and controls the trade-off between the hypersphere volume and the errors and Φ is a map function which maps input data into higher dimensional space. The corresponding dual problem is:

$$\begin{aligned} \max \sum_i \alpha_i K(x_i, x_i) - \sum_{i,j} \alpha_i \alpha_j K(x_i, x_j) \\ \text{s.t. } \sum_i \alpha_i = 1, 0 \leq \alpha_i \leq \frac{1}{\sqrt{M}}, i = 1, \dots, M \end{aligned} \tag{6}$$

where, $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_M\}$ is called lagrange multiplier vector, $K(-,-)$ is called kernel function such that $K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$ and a most widely used kernel function is the Radial Basic Function (RBF):

$$K(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{2\sigma^2}} \tag{7}$$

The optimization problem in Eq. 6 can be solved using standard quadratic programming methods and an optimal solution for α can be obtained. All x_i corresponding to non-zero α_i are called Support Vectors

(SV) which usually occupy a small quantity of the training data. Then the optimal solution for c is:

$$c = \sum_{i \in SV} \alpha_i \Phi(x_i)$$

On testing, for a new sample x , it is subjected to the map function Φ and if the distance from the mapped data to the center of the optimal hypersphere is smaller or equal than the radius of the hypersphere, then it is accepted, otherwise it is rejected which can be formulated as:

$$\begin{aligned} y(x) &= \text{sgn}(R^2 - \|\Phi(x) - c\|^2) \\ &= \text{sgn}\left[R^2 - \sum_{x_i, x_j \in SV} \alpha_i \alpha_j K(x_i, x_j) + 2 \sum_{x_i \in SV} \alpha_i K(x_i, x) - K(x, x)\right] \end{aligned} \tag{8}$$

Because optimal hypersphere is found by solving a quadratic programming problem, the SVDD do not suffer the problem of local minimum which means that SVDD training can always find a global minimum, thus has a good generalization capability.

DEFECT DETECTION ALGORITHM

Detection of detects is considered as a one-class classification problem. The defect detection algorithm is divided into two phases: Learning phase and classification phase. In the learning phase the fabric image is quantized and a set of GLCM features, as well as some new features are extracted from the quantized image to characterize the fabric texture. The extracted features are used as training data for SVDD training to generate a SVDD classifier. In classification phase the same features are extracted from the quantized testing image and subjected to the SVDD classifier to determine whether it is defective or not.

Feature extraction: Several features extracted from GLCM are used in the proposed detection algorithm. Generally, for GLCM, the displacement parameter (dx, dy), by which two highly related pixels are departed, is a good selection to characterize the fabric texture. A natural selection of displacement parameters for texture characterization is (0, 1) (1, 1) (1, 0) and (1, -1) which are also used for fabric texture characterization by Latif-Amet *et al.* (2000) and Lin (2010), since the neighboring pixels are considered highly related. These four displacement parameters correspond to 0, 45, 90 and 135°, respectively, where most defects are present. Chan and Pang (2000) find out from the frequency spectrum of the fabric texture image that texture periodicities are existing in the warp (0°) and fill (90°) direction in the fabric texture which can be calculated as

the reciprocal of the first harmonic frequency f_0 and f_{90} along warp and fill direction, respectively. Because of the highly periodicity of the fabric texture, two pixels departed by a texture periodicity are also considered highly related. Therefore, two additional displacement parameters are used in our algorithm: $(0, T_0)$ $(T_{90}, 0)$, where T_0 and T_{90} denote the texture periodicities at 0 and 90°, respectively, that is $T_0 = 1/f_0$, $T_{90} = 1/f_{90}$. f_0 and f_{90} can be obtained from the 1-D Fourier spectrum of fabric texture at 0 and 90°, respectively. Figure 4 shows a Fourier spectrum of a real fabric texture at 0°, the fabric image is made zero mean in advance to suppress the Direct Current (DC) component. Because T_0 and T_{90} are usually floating-points while displacement parameters must be integers for the computation of GLCM, the first-order linear interpolation is used to estimate the values of $I(x+dx, y+dy)$ where dx, dy are float-points. Haralick *et al.* (1973) proposed 14 features from GLCM for texture classification, in this study only two of them, namely contrast CON and inverse difference moment IDM, are used:

$$CON = \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} (i-j)^2 p(i, j) \quad (9)$$

$$IDM = \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \frac{1}{1+(i-j)^2} p(i, j) \quad (10)$$

where, L is the column (or row) number of GLCM (i.e., the total quantized levels), $p(I, j)$ refers to the normalized entry of the co-occurrence matrices. That is $p(I, j) = P_d(I, j)/R$, where $P_d(I, j)$ is the GLCM with displacement parameter d and R is the total number of pixel pairs (I, j) .

In addition, we propose 2 extra features for each displacement parameter (dx, dy) based on following conceptions. As discussed in earlier the pixel in the original image is quantized by its GCD and the pixel with larger GCD means more unlikely it belongs to the normal texture and its intensity tend to be much darker or lighter than the normal texture. It is found that defects, within their boundaries, tend to have more dark pixels (e.g., oil-stain, dirty-yarn, etc.) or light pixels (e.g., miss-pick, thin-place, etc.) than the normal texture. In turn large quantities of dark pixels or light pixels within a local region may indicate a defect in that region. Figure 5 shows a data fragment of quantized image of Fig. 2a with $L = 8$. All the lightness exceptional pixels whose quantized level is larger than a threshold UL which corresponds to the upper limit of 70% confidence interval of the second Gaussian, are marked by rectangles. There are mainly three reasons for appearance of lightness exceptional pixels: variation of normal texture, the noise and the real

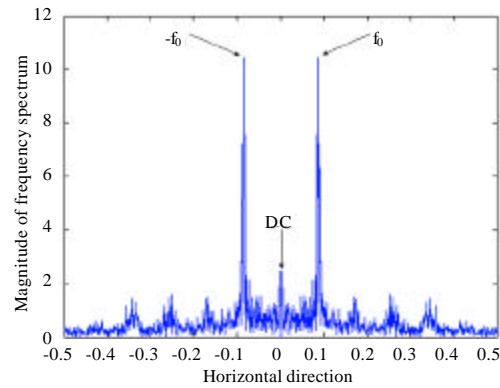


Fig. 4: Magnitude of frequency spectrum of fabric image along horizontal direction

3	3	3	4	5	4	3	2
4	4	4	6	7	5	4	3
4	4	4	5	5	4	5	4
5	4	4	4	5	6	5	4
4	2	3	3	4	5	3	3
3	2	4	4	5	4	3	4
4	5	6	6	7	7	6	6
5	6	6	6	6	7	6	5
3	4	3	4	4	4	4	3
3	3	2	4	5	4	3	3
3	4	4	5	5	4	4	4
4	4	4	4	5	4	3	4
7	5	3	4	5	5	3	3
6	3	3	3	5	5	3	3
5	3	4	5	6	5	4	3
5	4	5	5	5	5	4	3

Fig. 5: Data fragment of quantized image of Fig. 2a

defects. Because of the random property of variation of normal texture and the noise, lightness exceptional pixels and darkness exceptional pixels tend to scattered within the fabric image. However, the real defects tend to be continuous and constitute a small portion of field. Thus it is obvious that the connected lightness exceptional pixels in the center of Fig. 5 indicate a defect corresponding to the misspick in Fig. 2a and other scattered lightness exceptional pixels are caused by the variation of normal texture or the noise. A feature is proposed to characterize this property of defect.

Given a displacement parameter (dx, dy) , let Q be the quantized image, $u_k = x+k \cdot dx, v_k = y+k \cdot dy$, if for all $k = 0, 1, \dots, C-1, Q(u_k, v_k)$ are larger than UL and for $k = -1$ and $C, Q(u_k, v_k)$ are not larger than UL , then points in position (u_k, v_k) ($k = 0, 1, \dots, C-1$) are considered as a Lightness Exceptional Run (LER). C denotes the length of the LER. The feature Long Lightness Exceptional Run Emphasis (LLERE) is defined as:

$$LLERE = \sum_{s \in S_l} [C_s^2 \cdot \sum_{k=1}^{C_s-1} (s(k)-UL)] \quad (11)$$

where, S_l denotes a set containing all the LERs whose length is larger than 1 in quantized image matrix. C_s and $s(k)$ denote the length and the k -th element of the LER s . The feature LLERE is similar to the feature long run emphasis of Gray Level Run Length Matrix (GLRLM) (Galloway, 1975) which has been widely used for texture characterization and classification. They are both built on the statistic of consecutive pixels with the same attribution. However, in Eq. 11 the LERs whose length is smaller than or equal to 1 which are probably caused by the variation of normal texture or the noise, are not involved in the computation so that LLERE can emphasize the presence of real defects. l is set to the 95th percentile of the order statistics obtained from non-defective image samples. Similarly, another feature called Long Darkness Exceptional Run Emphasis (LDERE) which is the counterpart of LLERE is also used as a feature:

$$LDERE = \sum_{z \in Z_l} [C_z^2 \cdot \sum_{k=0}^{C_z-1} (DL-z(k))] \quad (12)$$

where, Z_l denotes a set containing all the Darkness Exceptional Runs (DER) whose length is larger than 1 in quantized image matrix. The definition of DER is similar to LER except that the values of its elements are smaller than a threshold DL which corresponds to the lower limit of 70% confidence interval of the first Gaussian. C_z and $z(k)$ denote the length and the k th element of the DER z . Compared to the features extracted from GLCM, feature LLERE and LDERE also characterize the relationship of pixels separated by a certain distance in a given direction but they put more emphasis on the continuousness of exceptional pixels along that direction which makes them suitable to detect tiny directional defects. In summary 6 displacement parameters are used and 4 features are extracted for each displacement parameter. In all 24 features are extracted to form a feature vector.

Learning phase: Images of non-defective texture are used in the learning phase. All these images are divided into non-overlapping subregions. The feature vectors V_m , $m = 1, 2, \dots, M$ are extracted from subregions using the feature extraction method proposed earlier, where M denotes the total number of non-defective subregions. Then the feature vectors are normalized as:

$$NV_m(r) = (V_m(r) - \eta_r) / \beta_r \quad (13)$$

Where:

$$\eta_r = \min_{m=1,2,\dots,M} \{V_m(r)\} \quad (14)$$

$$\beta_r = \max_{m=1,2,\dots,M} \{(V_m(r) - \eta_r)\} \quad (15)$$

$r = 0, 1, 2, \dots$ is the feature index of the feature vector, NV_m denotes the normalized feature vector, η_r and β_r are called offset coefficient and scale factor of the normalization, respectively. The objective of the normalization is to set the values of all the features in the feature vector within interval $[0, 1]$ and make sure that all the features in the feature vector have nearly the same weight. Then these feature vectors are subjected to the SVDD training. During the training, there are two parameters should be decided: the trade-off parameter v in Eq. 6 and the RBF width parameter σ in Eq. 7. A large v allows more outliers of the hypersphere in the training dataset which corresponds to larger rejection rate and larger fraction of support vectors. Tax and Duin (2004) find out that the fraction of support vectors relates to the false alarm rate, so the parameter v can be decided by the expected false alarm rate. To choose the optimal value of parameters σ , the cross validation method is used in the SVDD training.

Classification phase: All the fabric images under inspection are divided into non-overlapping regions of the same size as in learning phase. From each subregion, a feature vector E is extracted using the feature extraction method proposed earlier. Then the feature vector E is normalized using η_r and β_r which have been computed in the learning phase:

$$NE(r) = (E(r) - \eta_r) / \beta_r \quad (16)$$

Different from the normalization in learning stage, the values of normalized features $NE(r)$ can be less than zero or greater than one. The normalized feature vector NE is then subjected to the SVDD classifier and the final classification result can be acquired by Eq. 8. If its output is 1, then the subregion is considered as non-defective otherwise it is considered as defective.

RESULTS AND DISCUSSION

Three datasets containing one plain and two twill fabrics of different texture background are used to evaluate the performance of the fabric detection algorithm. All the three fabrics and defects on them are produced in factory practice. All of the images are acquired by line scan CCD camera with a spatial resolution of 0.2 mm/pixel against a backlighting illumination and digitalized into 256×256 pixels with a gray level of 256. Detailed information of the three datasets is presented in Table 1. Each image is divided into non-overlapping subregions of size 64×64 pixels and each subregion is considered as one sample for defect detection.

In the training phase, for each dataset, 1000 non-defective samples are used for SVDD training. The training parameter v is set to 0.01 and σ is decided by 10-fold cross validation. Figure 6a and b illustrate the cross validation accuracy and the proportion of support vectors with various values of σ^2 , where $\sigma^2 = 2^{-3}, 2^4, \dots, 2^{15}$ are selected as candidates which is suggested by Hsu *et al.* (2003). It can be seen from Fig. 6a that both too small and too larger values of σ^2 result in low cross validation accuracy. Large value of σ^2 will create a simple decision boundary with small proportion of support vectors which is unable to separate the defective and non-defective samples and leads to high miss rate. While small value of σ^2 will create an excessive complex decision boundary with large proportion of support vectors which leads to over-fitting and high false alarm rate. $\sigma^2 = 2^7, 2^6$ and 2^4 with highest cross validation accuracy are selected for the three dataset, respectively and their corresponding support vector proportion are 1.3, 2.3 and 1.3%, respectively which only occupy a very small portion of total training vectors. It can be seen from Fig. 6a that the values of cross validation accuracy nearby the optimal value of σ^2 are nearly stationary, so the cross validation method for finding optimal σ^2 is robust.

Figure 7 illustrates the adaptive quantization and detection results of several typical defective samples of three datasets with quantization parameter $\lambda = 1, N = 2$. Dataset 1, 2 and 3 are quantized to 8, 9 and 10 levels,

respectively. We can see that after the quantization all the defects are still clear and intact but the number of gray-levels is decreased from 256 to 8, 9 and 10, respectively which greatly reduced the computational load of GLCM and its features.

Quantization parameter selection: As mentioned earlier, there are two important parameters affecting the procedure of quantization. One is the interval width λ , the other one is N which relates to the total number of quantized levels. The interval $[\mu_i - (0.5+N)\lambda, \mu_i + (0.5+N)\lambda]$ constitutes a confidence interval of the i -th Gaussian. Gray-levels out of this interval are considered either dark exceptional or light exceptional. Large value of N corresponds to more computational load, while small value of N may loses some detailed information of the defects and makes them hard to detect. In order to find the optimal parameters, several pairs of (λ, N) are used to the three datasets. The detection results of the three datasets of different parameter pairs of (λ, N) are presented in Table 2-4, respectively. λ is set inversely proportional to N so that the confidence intervals of all parameter pairs are nearly the same. As the false alarm relates to the trade-off parameter v in Eq. 6, it does not have large variation for different parameter pairs. For $N = 0$ to 2, the miss detection rate decreases intensively which means the detection performance is getting much better, while for $N = 3$ to 6 the miss detection rate does not have great improvement and it changes inversely to the false alarm rate which means the detection performance does not greatly improved but adding more computational load. As the increase of value of N , more detailed information of fabric texture is preserved. For defects miss-yarn of Dataset 1 and dirty-yarn of Dataset 2 only small value of N is sufficient to defect most of them, because they

Table 1: Information of experimental datasets

Attribute	Dataset 1	Dataset 2	Dataset 3
fabric type	plain	twill	twill
T_0	5.5	9.4	6.8
T_{90}	9.7	11.6	14.1
Non-defective	1936	964	1548
Defective	155 (miss-pick)	65 (dirty-yarn)	342 (miss-pick)
	464 (miss-yarn)	42 (hole)	250 (thin-place)

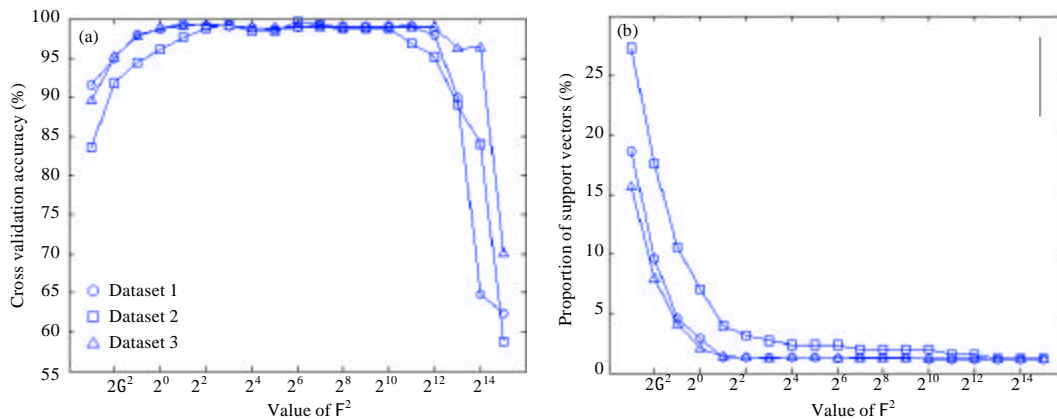


Fig. 6(a-b): The cross validation accuracy and the proportion of support vectors with various values of σ^2

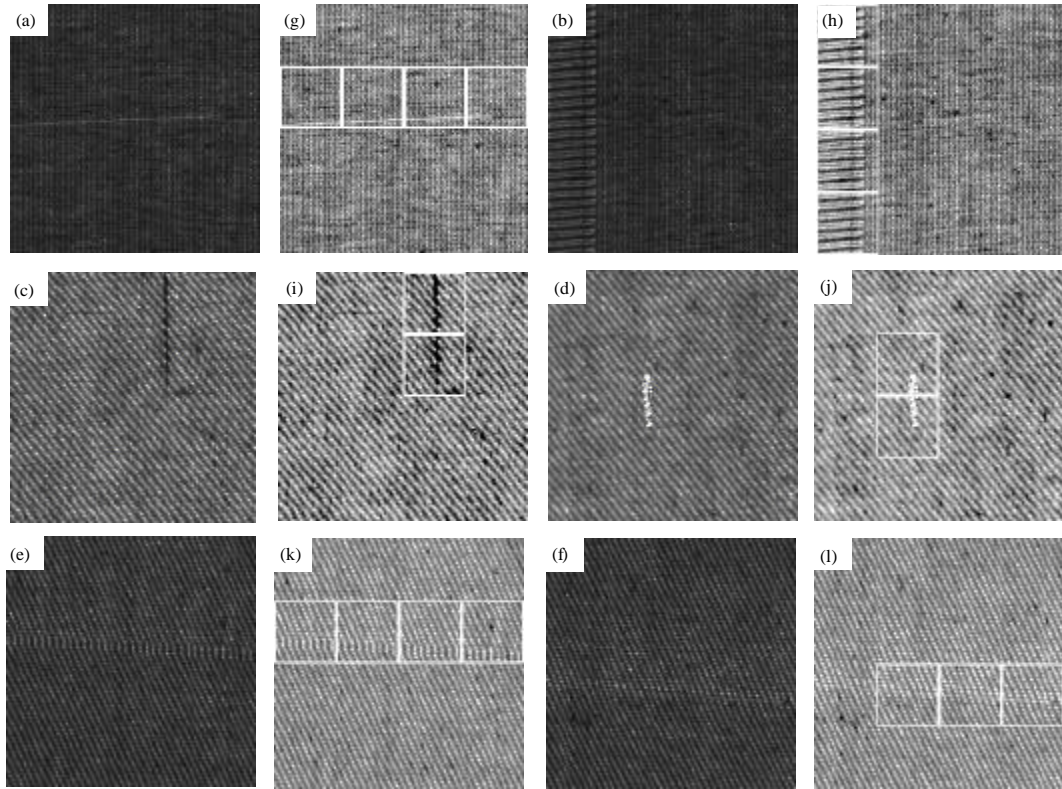


Fig. 7(a-l):(a, b) Miss-pick and miss-yarn of Dataset 1, (c, d) dirty-yarn and hole of Dataset 2, (e, f) miss-pick and thin-place of Dataset 3 and (g-l) respectively; corresponding quantization images and detection result

Table 2: Detection results of Dataset 1 with different parameter pairs (λ , N)

N	λ	L	False alarm	Miss detection of miss-pick	Miss detection of miss-yarn
0	5.00	3	18 (0.92)	103 (66.45)	18 (3.88)
1	1.67	6	20 (1.03)	13 (8.39)	2 (0.43)
2	1.00	8	20 (1.03)	3 (1.93)	0 (0)
3	0.71	10	22 (1.14)	3 (1.93)	0 (0)
4	0.56	13	22 (1.14)	3 (1.93)	0 (0)
5	0.45	16	29 (1.29)	2 (1.29)	0 (0)
6	0.38	19	23 (1.19)	3 (1.93)	0 (0)

Values in brackets are percentage

Table 3: Detection results of Dataset 2 with different parameter pairs (λ , N)

N	λ	L	False alarm	Miss detection of dirty-yarn	Miss detection of hole
0	5.00	3	10 (1.05)	5 (8.33)	6 (14.2)
1	1.67	7	12 (1.24)	1 (1.53)	2 (4.76)
2	1.00	9	12 (1.24)	0 (0)	1 (2.38)
3	0.71	12	12 (1.24)	0 (0)	1 (2.38)
4	0.56	15	14 (1.45)	0 (0)	0 (0)
5	0.45	18	13 (1.34)	0 (0)	0 (0)
6	0.38	20	13 (1.34)	0 (0)	0 (0)

Values in brackets are percentage

contain lots of dark or light exceptional pixels whose gray-level are out of the confidence interval. Generally, N = 2 and 3 are optimal parameters which can be used to detect most of the defects.

Characteristic of features: The proposed algorithm use all 24 features described, to detect different kinds of defects. In order to know the specific characteristic of each feature, we also investigate on the discriminative

Table 4: Detection results of Dataset 3 with different parameter pairs (λ , N)

N	λ	L	False alarm	Miss detection of miss-pick	Miss detection of thin-place
0	5.00	3	13 (0.84)	312 (91.23)	152 (60.8)
1	1.67	8	19 (1.23)	41 (11.99)	29 (11.6)
2	1.00	10	18 (1.16)	11 (3.21)	20 (8)
3	0.71	13	21 (1.36)	11 (3.21)	17 (6.8)
4	0.56	15	21 (1.36)	12 (3.51)	18 (7.2)
5	0.45	18	19 (1.23)	14 (4.09)	20 (8)
6	0.38	21	20 (1.29)	15 (4.38)	18 (7.2)

Values in brackets are percentage

Table 5: Discriminative features of each kind of defects in Fig. 7

Defects	Fig. No.	(dx, dy)	Discriminative features	U_n	σ_n	U_d	J
Miss-pick	7a	(0,1)	LLERE	0.16	0.19	11.85	60.96
		(0,1)	CON	0.44	0.20	-0.44	4.51
		(0,1)	IDM	0.61	0.21	1.37	3.59
Miss-yarn	7b	(0,1)	LDERE	0.16	0.19	97.47	507.30
		(0,1)	LDERE	0.14	0.15	21.02	139.22
		(0,1)	CON	0.44	0.20	-2.34	14.14
		(0,1)	IDM	0.61	0.21	4.58	18.76
Dirty-yarn	7c	(1,0)	LDERE	0.21	0.21	8.28	38.47
		(1,0)	CON	0.49	0.29	1.13	2.16
		(1,0)	IDM	0.42	0.28	-0.18	2.12
Hole	7d	(1,0)	LDERE	0.36	0.23	1.97	7.03
		(1,0)	IDM	0.42	0.28	0.68	0.93
Miss-pick	7e	($T_{90},0$)	LDERE	0.25	0.17	0.96	4.19
		($T_{90},0$)	CON	0.49	0.23	3.14	11.77
		($T_{90},0$)	IDM	0.47	0.21	-0.29	3.57
Thin-place	7f	($0,T_0$)	LLERE	0.16	0.11	0.82	5.92

features for each kind of defects. Here, the discriminative features refer to the normalized features, among all 24 features which have large distances between the defective samples and non-defective samples. A criterion function is used to characterize this distance which is formulated as:

$$J = \frac{|U_n - U_d|}{\sigma_n} \quad (17)$$

where, U_n and σ_n are the mean value and standard deviation of the normalized feature extracted from the non-defective subregions in the learning phase (Eq. 13), U_d is the mean value of the normalized feature extracted from defective subregions in classification phase (Eq. 16). The higher magnitude of J indicates larger distance between the feature of the defect and normal texture which means better detection performance. Table 5 shows the discriminative features and their values of J for each kind of defect in Fig. 7, only the displacement parameters and features which have large J are presented, others are omitted. Generally, the value of J in Table 5 is consistent with the detection rate in Tables 2-4, that is larger value of J corresponds to higher detection rate. It also suggests that the features LLERE and LDERE are more effective than CON and IDM to detect tiny directional defects such as miss-pick of Dataset 1, dirty-yarn of Dataset 2 and thin-place of Dataset 3, because CON and IDM characterize the global texture pattern and not quite sensitive to the local textural change caused by tiny

defects. However, LLERE and LDERE emphasize the continuousness of lightness exceptional pixels and darkness exceptional pixels, respectively, so they are suitable to characterize tiny directional defects. All of features LLERE, LDERE, CON, IDM with displacement parameter (0,1) have large value of J for defect miss-yarn, because the texture pattern of miss-yarn is quite different from normal texture and containing lots of dark and light exceptional pixels in horizontal orientation. Particularly, for the defect thin-place in Fig. 7e which is not even obvious to the human inspector and thus very difficult detect, its defective pixels (lightness exceptional pixels) are arranged periodically in horizontal direction, so feature LLERE with displacement parameter (0, T_0) can characterize it.

Real-time performance: In order to evaluate the real-time performance of our detection algorithm, a prototyped defect detection system is built in our laboratory. The architecture of the detection system is illustrated in Fig. 8. Line-scan camera Dalsa SP-14 with pixel resolution of 2048 is used to capture the image of fabrics moving on a conveyor belt. The localization resolution is 0.2 mm/pixel, so each camera can cover 0.4 cm transversal and four cameras are used to cover 1.6 m transversal. An encoder is implemented to synchronize the scan rate of the cameras with the movement speed of the fabrics. The image data of the camera are transferred to the image acquisition and processing card via camera link interface. The processing unit is a Digital Signal Processor (DSP) TI TMS320C6713 operating at 300 MHz and the proposed

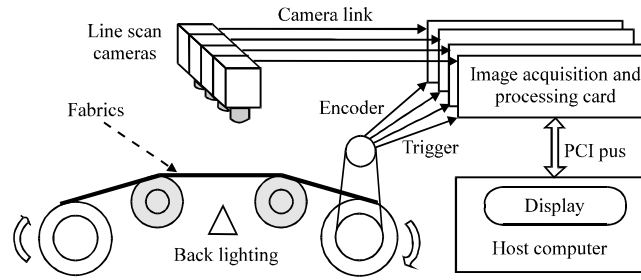


Fig. 8: The architecture of prototyped defect detection system

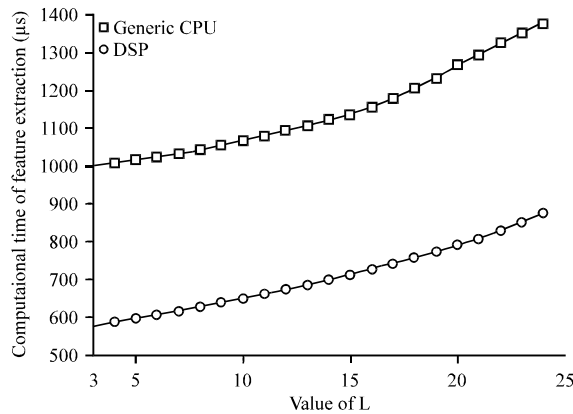


Fig. 9: Real time performance of the detection algorithm

detection algorithm is implemented in it. All the detection results are uploaded to a host computer for display via PCI bus.

Compared to the generic CPU in host computer, DSP has better real-time performance benefiting from its specialized architecture such as hardware multiplier and instruction pipeline. Here, we only focus on the computational time in the classification phase and the time-consuming procedures in the learning phase such as parameter evaluation of the texture model, SVDD training and cross validation are not taken into consideration, because the learning phase is finished before the real-time inspection, thus usually not time-constrained. The classification phase consists of two parts: feature extraction and decision. The computational time of feature extraction relates to the total number of quantized levels. Feature extraction time for subregion of size 64×64 with different values of L is presented in Fig. 9. For comparison both computational time implemented in DSP and generic CPU P4 3.0 G are presented. The computational time of non-quantized feature extraction with 256 gray-levels is 45844 μ s in DSP and 29635 μ s in generic CPU. We can find out that the adaptive quantization greatly reduces the computational complexity of feature extraction and

improve the real-time performance of the algorithm. The decision involves classification of SVDD classifier which is implemented by software LIBSVM (Chang and Lin, 2001). According to Eq. 8, only the support vectors which usually occupy a small quantity of the training data (Fig. 6b), are involved in the computation of classification. Moreover, LIBSVM uses a look-up table cache for the computation of kernel function in Eq. 7, so that the decision time is greatly reduced and only occupies a small portion in the detection algorithm. It can be seen from Fig. 9 that as L increases, the computational time increases nonlinearly and according to Table 2-4, L increases with the increase of N . However, when N is greater than 3, the detection rate does not increase obviously, so the parameter $N = 2$ or 3 is a good selection compromising between the detection rate and the real-time performance. The detection speed can achieve as fast as 40 m min^{-1} .

CONCLUSIONS

A new approach of textural fabric defect detection algorithm using SVDD has been demonstrated. A fabric texture model of mixed Gaussian distribution was built on the gray-level histogram of fabric image. Two GLCM features and two novel features were used to characterize the fabric texture pattern and emphasize the presence of the defects. An adaptive quantization method base on the texture model was proposed to reduce the size of GLCM, so that the computational complexity of feature extraction was tremendously reduced. The specific property of each feature was also discussed. User can remove unnecessary ones to further improve the real-time performance. The SVDD classifier was used as a detector and achieved good detection results for three datasets in experiment. A prototyped defect detection system, with high performance DSP as its processing unit, was built to evaluate the real-time performance of the proposed algorithm, experiment indicated that the detection speed

can reach as fast as 40 m min^{-1} , thus the proposed algorithm is suitable for on-line inspection in industrial occasions.

ACKNOWLEDGMENT

This study was supported by Open fund of Image Processing and Intelligent Control Key Laboratory of Education Ministry of China.

REFERENCES

- Bodnarova, A., M. Bennamoun and S. Latham, 2002. Optimal Gabor filters for textile flaw detection. *Pattern Recog.*, 35: 2973-2991.
- Bu, H.G., J. Wang and X.B. Huang, 2009. Fabric defects detection based on multiple fractal features and support vector data description. *Eng. Applied Artif. Intell.*, 22: 224-235.
- Bu, H.G., X.B. Huang, J. Wang and X. Chen, 2010. Detection of fabric defects by auto-regressive spectral analysis and support vector data description. *Text. Res. J.*, 80: 579-589.
- Chan, C.H. and G. Pang, 2000. Fabric defect detection by Fourier analysis. *IEEE Trans. Ind. Appl.*, 36: 1267-1276.
- Chandra, J.K., P.K. Banerjee and A.K. Dattab, 2010. Neural network trained morphological processing for the detection of defects in woven fabric. *J. Text. Inst.*, 101: 699-706.
- Chang, C.C. and C.J. Lin, 2001. LIBSVM: A library for support vector machines. <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>
- Chu, H., Z. Xie, X. Xu, L. Zhou and Q. Liu, 2011. Inspection and recognition of generalized surface defect for precise optical elements. *Inform. Technol. J.*, 10: 1395-1401.
- Cohen, F.S., Z. Fan and S. Attali, 1991. Automated inspection of textile fabrics using textural models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13: 803-808.
- Conci, A. and C. B. Proenca, 1998. A fractal image analysis system for fabric inspection based on box-counting method. *Comput. Netw. ISDN Syst.*, 30: 1887-1895.
- Galloway, M.M., 1975. Texture analysis using gray level run lengths. *Comput. Graph. Image Process.*, 4: 172-179.
- Haralick, R.M., K. Shanmugam and I.H. Dinstein, 1973. Textural features for image classification. *IEEE Trans. Syst. Man Cybernet.*, 3: 610-621.
- Hsu, C.W., C.C. Chang and C.J. Lin, 2003. A practical guide to support vector classification. Technical report, Department of Computer Science and Information Engineering, National Taiwan University, <http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf>
- Jianli, L. and Z. Baoqi, 2007. Identification of fabric defects based on discrete wavelet transform and back-propagation neural network. *J. Text. Inst.*, 98: 355-362.
- Kumar, A. and G. Pang, 2002. Defect detection in textured materials using Gabor filters. *IEEE Trans. Ind. Appl.*, 38: 425-440.
- Kumar, A. and H.C. Shen, 2002. Texture inspection for defects using neural networks and support vector machines. *Proc. ICIP*, 3: 353-356.
- Kumar, A., 2003. Neural network based detection of local textile defects. *Pattern Recognit.*, 36: 1645-1659.
- Latif-Amet, L., A. Ertuzun and A. Ercil, 2000. An efficient method for texture defect detection: Subband domain co-occurrence matrices. *Image Vision Comput.*, 18: 543-553.
- Lin, J.J., 2010. Pattern recognition of Fabric defects using case-based reasoning. *Text. Res. J.*, 80: 794-802.
- Loum, G., C.T. Haba, J. Lemoine and P. Provent, 2007. Texture characterisation and classification using full wavelet decomposition. *J. Applied Sci.*, 7: 1566-1573.
- Mahi, H. and H.F. Izabatene, 2011. Segmentation of satellite imagery using RBF neural network and genetic algorithm. *Asian J. Applied Sci.*, 4: 186-194.
- Mallik-Goswami, B. and A.K. Datta, 2000. Detecting defects in fabric with laser-based morphological image processing. *Text. Res. J.*, 70: 758-762.
- Mingde, B. and S. Zhigang, 2011. Fabric defect detection using undecimated wavelet transform. *Inform. Technol. J.*, 10: 1701-1708.
- Nadhim, A.M.M., 2006. Introduction to multitone multiresolution wavelet analysis. *Inform. Technol. J.*, 5: 494-502.
- Nagarajan, B. and P. Balasubramanie, 2008. Neural classifier for object classification with cluttered background using spectral texture based features. *J. Artif. Intell.*, 1: 61-69.
- Ohanian, P.P. and R.C. Dubes, 1992. Performance evaluation for four classes of textural features. *Pattern Recog.*, 25: 819-833.
- Raju, U.S.N., B.E. Reddy, V.V. Kumar and B. Sujatha, 2008. Texture classification based on extraction of skeleton primitives using wavelets. *Inform. Technol. J.*, 7: 883-889.

- Randen, T. and J. Husoy, 1999. Filtering for texture classification: A comparative study. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21: 291-310.
- Sari-Sarraf, H. and J.S. Goddard, 1999. Vision systems for on-loom fabric inspection. *IEEE Trans. Ind. Appl.*, 35: 1252-1259.
- Shang, S., X. Kong and X. You, 2011. Copy paper brand source identification using commodity scanners. *Inform. Technol. J.*, 10: 2112-2118.
- Sheng, X., P. Qi-Cong and H. Shan, 2010. An invariant descriptor design method based on MSER. *Inform. Technol. J.*, 9: 1345-1352.
- Tax, D.M.J. and R.P.W. Duin, 2004. Support vector data description. *Mach. Learn.*, 54: 45-66.
- Yang, X.Z., G. Pang and N. Yung, 2002. Discriminative fabric defect detection using adaptive wavelets. *Opt. Eng.*, 41: 3116-3126.