Available online at www.sciencedirect.com
ScienceDirect
Pattern Recognition 40 (2007) 121–133
PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY
www.elsevier.com/locate/patcog

Using perceptual relation of regularity and anisotropy in the texture with independent component model for defect detection

O.G. Sezer[a,1], A. Ercil[b,*], A. Ertuzun[c]

[a] *Electrical Sciences and Computer Engineering, Brown University, Providence, RI 02912, USA*
[b] *Faculty of Engineering and Natural Sciences, Sabanci University, Istanbul, Turkey*
[c] *Electrical and Electronic Engineering, Bogazici University, Istanbul, Turkey*

Received 15 November 2005; received in revised form 2 December 2005; accepted 15 May 2006

Abstract

This paper addresses the raw textile defect detection problem using independent components approach with insights from human vision system. Human vision system is known to have specialized receptive fields that respond to certain type of input signals. Orientation-selective bar cells and grating cells are examples of receptive fields in the primary visual cortex that are selective to periodic- and aperiodic-patterns, respectively. Regularity and anisotropy are two high-level features of texture perception, and we can say that disruption in regularity and/or orientation field of the texture pattern causes structural defects. In our research, we observed that independent components extracted from texture images give bar or grating cell like results depending on the structure of the texture. For those textures having lower regularity and dominant local anisotropy (orientation or directionality), independent components look similar to bar cells whereas textures with high regularity and lower anisotropy have independent components acting like grating cells. Thus, we will expect different bar or grating cell like independent components to respond to defective and defect-free regions. With this motivation, statistical analysis of the structure of the texture by means of independent components and then extraction of the disturbance in the structure can be a promising approach to understand perception of local disorder of texture in human vision system. In this paper, we will show how to detect regions of structural defects in raw textile data that have certain regularity and local orientation characteristics with the application of independent component analysis (ICA), and we will present results on real textile images with detailed discussions.
© 2006 Pattern Recognition Society. Published by Elsevier Ltd. All rights reserved.

Keywords: Human vision; Texture defect detection; Independent component analysis; Receptive fields

1. Introduction

Texture defect detection problem played a significant role for quality control in the production phase of goods and attracted the attention of researchers for many years. Although there are some dedicated methods that work well for certain type of textures, they often fail when different type of texture is given. On the other hand, defect detection ability of a human observer is far superior to current algorithms and works even when the observer sees that texture type for the first time. To understand the process of perception of texture, Rao and Lohse [1] revealed in their classical experiment, that regularity and anisotropy play a significant role as high-level features for texture perception by humans. The anisotropy in this nomenclature can be defined as the directionality or the orientation field in the texture. Every periodic pattern has one or more characteristic directions defined by its periodicity vectors which identify directionality (anisotropy) of the texture. Regularity, on the other hand, is known as the opposite of randomness corresponding to a higher-level organization of the structure of the texture with a repetitive deterministic intensity pattern. From this motivation, disturbance of the regularity and/or the orientation in the structure of the texture can guide us in detecting the

* Corresponding author. Tel.: +90 2164839543; fax: +90 2164839550.
E-mail address: aytulercil@sabanciuniv.edu (A. Ercil).
[1] This research was done when the author was at the Electrical and Electronics Engineering Department of Bogazici University and Faculty of Engineering and Natural Sciences Sabanci University.

0031-3203/$30.00 © 2006 Pattern Recognition Society. Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2006.05.023

defects and there are successful algorithms in the literature that make use of this idea such as Ref. [2]. Here, we will try to relate the properties of human vision system with these high-level features by means of independent component analysis (ICA) so as to achieve defect detection in texture structure.

First attempts to understand human vision system and birth of experimental psychology later called "psychophysics" dealt with changes in the mental state due to given input to the brain (a black box) such as a light beam. These studies were later supported by neuro-physiological works and after Hubel and Wiesel's pioneering experiments [3,4] growing attention of scientists was focused on the properties of the neurons that act as receptive fields in the primary visual cortex. It was shown in these experiments that these receptive fields are localized in time and space, have band-pass characteristics both in spatial and temporal domains and are selective to certain orientations. This idea being as the pivoting point, Barlow [5] proposed that these receptive fields act like some redundancy reduction mechanism and produces factorial coding of the input data. Hence the factorial coding and oriented-edge selective receptive field idea merged together and Field [6] claimed that these receptive fields enable sparse representation of the input data. Thus, only a few features are needed to be active for representing an image and for a group of images a particular feature will rarely be active. This theory later was tested experimentally by Olshausen and Field [7,8] by using a network that maximizes the sparseness of the input data coming from patches of natural images. These works were also supported by Bell and Sejnowski [9] and Hurri's [10] papers, however they used ICA, which aims to search for factorial coding of the data by finding mutually independent components. It is shown in Ref. [8] that ICA and maximization of sparseness for input data actually are related. Afterward van Hateren and van der Schaaf [11] quantitatively compared the properties of independent component filters and receptive fields in primary visual cortex. They showed that the properties of the independent component filters obtained by ICA on a large set of natural images resemble properties of the receptive fields of simple cells in macaque monkeys' cortex, which indicates that expected statistics of the natural stimuli in the environment affects the characteristics of receptive fields. Although the independent component model lacks many aspects of simple cells such as contrast adaptation [12] and nonlinearities in orientation tuning [13], it has clear information theoretical conclusions that based on statistics of stimuli. Furthermore, it can be said that receptive fields work to decompose and reduce the information redundancy in the scene that fall onto retina for different specialized tasks such as edge detection or contrast adjustment. With the same incentive, we will try to apply ICA to extract statistical properties of texture images, rather than natural images, for detecting underlying defects by reducing redundancies and decomposing texture images into independent components, which may help us to represent them in a lower-dimensional space.

In previous works on ICA, researchers tried to come up with a common representation basis for getting general statistics of the environment. Hurri [10] showed in his work that independent components of natural and texture images appeared to be different from each other. The reason for this difference is due to different nature of the statistical characteristics of natural and texture images. Hurri in his experiments used various types of texture images from Brodatz image database. Since each type of texture has drastically different structural properties, representing different textures with common independent components was infeasible. In our work, we extract independent components for each texture separately for analyzing its own structure. We observed that depending on the texture type, independent components are similar to bar or grating cells defined in Refs. [14,15] which give clues about regularity and anisotropy of the texture. Briefly we can say that, grating cells are selective for periodic oriented patterns. This is why they detect exclusively oriented texture and do not respond to contours, edges. On the other hand, bar cells are selective to oriented edges or contours (i.e., bars) that are not part of a periodic pattern. Different from the simple and the complex cells, grating and bar cells are periodic- and aperiodic-pattern selective cells, respectively. Since the structural properties of the raw textile data is formed by combination of the periodicity and the orientation features and also a defect can be defined as disruption of the structure of the texture, the idea to relate these cells with defect detection problem becomes intuitive. Even though, it is known that bar and grating cell models exhibit nonlinear behavior, they can guide us in understanding the relation of the independent components with the structure of the texture in defect analysis.

From image processing point of view, we can classify defects in raw textile data into three categories: intensity defects, geometric defects and mixture of both. Intensity defects (Refer to Fig. 1(a)) are more easily detected by a human observer than geometric errors. We observed that independent components similar to bar cells respond to intensity defects when there is an imperfection in the structure of the texture rather than responding to texture itself. That probably explains why intensity defects catch the attention of an observer, regardless of the texture type. For geometric defects (refer to Fig. 1(b)), studies of von der Heydt et al. [14] revealed special cells called grating cells in the early vision system that respond to both orientation and periodicity of the given pattern and Petkov and Kruizinga [15] simulated these findings for texture detection and noted that these cells just respond to textured areas in a given scene and respond weakly or do not respond at all to an input image which contains non-texture image attributes. Thus, grating cells and bar cells can be thought of as higher-level processor units in the visual system that respond to regularity and orientation of the texture's structure, and this can be used as a model for texture analysis. From this point, we expect geometric

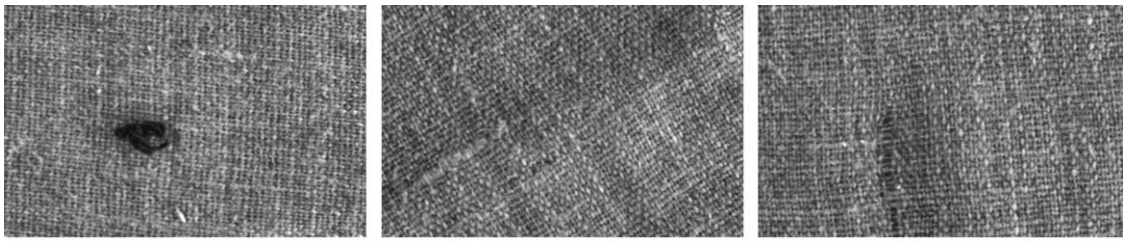

Fig. 1. Three texture images from the TILDA database ($c2r2$): (a) an intensity defect; (b) geometric defect; and (c) mixture of both.

defects, which can be described as disruption of periodicity and/or orientation of the pattern, to cause different response characteristics in the grating cells that are representing the defective region of the input image compared with their response characteristics to defect-free region. In this article, we will try to imitate this effect by extracting independent components that describe the statistical properties of the texture, which have similar form as grating cells that are described in literature. It is important to note that our aim here is not to validate or simulate the model set by von der Heydt et al. [14] but to give possible statistical intuitions that may help us to understand principle behind the possible evolution of the grating and bar cells and use this idea for texture analysis.

2. Independent component analysis

Linear transformation of the multivariate data is sought in many disciplines such as statistics, signal processing or neural networks for suitable representation of the original data. This representation enables one to have simple description of the data for both computationally and conceptually. ICA [16], a recently emerged method that became the center of attention in many research areas and applications, is also a linear transformation. It minimizes the statistical dependence in representing the multivariate data to capture the essential structure in it. In ICA rather than second order statistics, which is the case for principle component analysis (PCA), higher-order statistics of the multivariate data is used for estimation of the linear model.

Definition of the ICA can be stated as follows [17]:

Let s_i be independent non-gaussian sources and x_j be the observed data that we get from our data acquisition device. Assume x_j can be represented as linear mixture of independent components s_i's:

$$x_j = a_{j1}s_1 + a_{j2}s_2 + \cdots + a_{jn}s_n \quad \text{for all } j\text{'s.} \tag{1}$$

The above linear combination can be represented in vector–matrix notation as

$$\boldsymbol{x} = \boldsymbol{A}\boldsymbol{s}, \tag{2}$$

where $\boldsymbol{x}$ and $\boldsymbol{s}$ are column vectors of observed data x_j's and independent components s_i's, respectively, and $\boldsymbol{A}$ is the mixing matrix. Therefore, it can be said that ICA is a generative model which describes how the observed data $\boldsymbol{x}$ can be represented by superposition of sources s_j's or superposition of columns of mixing matrix $\boldsymbol{A}$.

In ICA model, $\boldsymbol{s}$ is also called as a latent variable or a hidden factor that underlies in random variable $\boldsymbol{x}$. In this model, vector $\boldsymbol{x}$ is the only a priori known quantity and both $\boldsymbol{A}$ and $\boldsymbol{s}$ are assumed to be unknown. Therefore $\boldsymbol{A}$ and $\boldsymbol{s}$ should somehow be estimated with the information that $\boldsymbol{s}$ is non-gaussian and entries of the $\boldsymbol{s}$ vector are statistically independent. Fortunately, ICA enables us to make use of the given assumptions in the model to estimate both $\boldsymbol{A}$ and $\boldsymbol{s}$. Once $\boldsymbol{A}$ is estimated its (pseudo) inverse $\boldsymbol{W}$ can be used to find source or latent variable vector

$$\boldsymbol{s} = \boldsymbol{W}\boldsymbol{x}. \tag{3}$$

The above equation is also known as sparse coding, which is closely related to ICA. Sparse coding enables one to represent data by having just a few active units out of a larger collection of model vectors. Therefore in sparse coding, data is represented by each of the components that are rarely active (i.e., zero most of the time) [18].

In this paper, our aim is to represent image patches (sub-windows) extracted from a textile image with a linear generative model shown in Eq. (2) where the stochastic sources (s_i) are as mutually independent as possible, which will be achieved by ICA. There are two standard preprocessing steps that are used in ICA. These steps are applied in order to make ICA estimation better conditioned and simpler; hence they do not change the ICA model given in Eq. (2). The first step is centering the data by subtracting the mean of the data from itself:

$$\boldsymbol{x} = \boldsymbol{x} - E\{\boldsymbol{x}\}. \tag{4}$$

Now we have zero-mean sources. Whitening the data is the second step in preprocessing. This means that we linearly transform the data $\boldsymbol{x}$ into $\boldsymbol{x}'$ so the components of $\boldsymbol{x}'$ are uncorrelated and their variances are equal to unity. In other words, the covariance matrix of the transformed data $\boldsymbol{x}'$ is equal to identity matrix and $\boldsymbol{x}'$ is called the whitened data [19]:

$$E\{\boldsymbol{x}'\boldsymbol{x}'^{\mathrm{T}}\} = \boldsymbol{I} \quad \text{where } \boldsymbol{x}' = \boldsymbol{V}\boldsymbol{x}. \tag{5}$$

The whitening transformation is always positive. For image data there are two commonly used analytically available solutions for whitening transformations [9]. The first one is the symmetric whitening matrix $\boldsymbol{V}_{\mathrm{ZCA}} = \boldsymbol{E}\{\boldsymbol{x}\boldsymbol{x}^{\mathrm{T}}\}^{-1/2}$ (ZCA: zero-phase component analysis) which is a local solution. It was argued in Ref. [20] that center-surround receptive fields of neurons in the retina and the lateral geniculate nucleus (LGN) carry out a similar activity to the input data as symmetric whitening. The other one is PCA solution. Here, the whitening matrix is $\boldsymbol{V}_{\mathrm{PCA}} = \boldsymbol{D}^{-1/2}\boldsymbol{E}^{\mathrm{T}}$ where $\boldsymbol{E}$ is the matrix containing eigenvectors and $\boldsymbol{D}$ has eigenvalues of the covariance matrix on its diagonal. In our work, we used second solution since PCA enables one to reduce the dimensionality of the image data, which lowers the run-time computational cost and memory consumption in the ICA estimation.

After the preprocessing steps described above, the goal of the ICA is to find an orthogonal transformation matrix $\boldsymbol{W}$ such that statistical dependencies between the estimated sources are minimized.

$$\hat{\boldsymbol{s}} = \boldsymbol{W}\boldsymbol{x}' = \boldsymbol{W}\boldsymbol{V}_{\mathrm{PCA}}\boldsymbol{x} = \boldsymbol{W}\boldsymbol{D}_n^{-1/2}\boldsymbol{E}_n^{\mathrm{T}}\boldsymbol{x}, \tag{6}$$

where $\boldsymbol{D}_n$ denotes the diagonal matrix containing the n largest eigenvalues of the covariance matrix of $\boldsymbol{x}$ and $\boldsymbol{E}_n$ denotes the corresponding matrix whose columns are the eigenvectors corresponding to the n largest eigenvalues. The next step, therefore, is to apply one of the ICA algorithms available in the literature. We used symmetric fixed-point algorithm [21] with $f(x) = \tanh(x)$ nonlinearity in order to obtain a fast solution using a simple algorithm. The algorithm starts from a random orthogonal matrix $\boldsymbol{W}$ and in each iteration, rows of it, $(\boldsymbol{w}^{\mathrm{T}})$, is updated by

$$\boldsymbol{w}_i := E\{\boldsymbol{x}' f(\boldsymbol{w}_i^{\mathrm{T}}\boldsymbol{x}')\} - E\{f'(\boldsymbol{w}_i\boldsymbol{x}')\}\boldsymbol{w}_i \tag{7}$$

followed by ortho-normalization of the matrix through $\boldsymbol{W} := (\boldsymbol{W}\boldsymbol{W}^{\mathrm{T}})^{-1/2}\boldsymbol{W}$. And finally, after convergence is achieved the estimated basis is constructed as

$$\boldsymbol{A} = \boldsymbol{E}_n\boldsymbol{D}_n^{1/2}\boldsymbol{W}^{\mathrm{T}}, \tag{8}$$

where $\boldsymbol{A}$ is the mixing matrix in the ICA model. In this paper the columns of de-mixing matrix, $\boldsymbol{W}$, are called as independent component filters. After this mathematical description of how to obtain independent component filters of texture images, methodology that we follow in the algorithm will be explained next.

3. Texture defect detection

3.1. Overview

Texture defect detection is a broad research field. There are two important variables in texture analysis: the underlying characteristics of the texture and the type of the defects. Most often, methods that perform well in one type of texture or defect fail in others. Therefore, before developing an algorithm these variants should be well defined.

Texture defects are classified into three categories as it was stated before; geometrical defects, intensity defects, and a mixture of both (refer to Fig. 1). A geometrical defect is a distortion, which does not significantly change gray level histogram at that point but rather changes the existing spatial correlation between the pixels. Intensity defects, on the other hand, cause noticeable changes in the gray-level intensity values [22].

We will mainly concentrate on defect detection in raw textile images which is a difficult problem due to two reasons: (i) textile fabric images are characterized by complex textures and (ii) defect types occupy a wide spectrum (i.e., training algorithm by certain type of defects does not guarantee detection of other types of defects). Therefore simple thresholding techniques will not satisfy the required performance and methods that use higher-order statistics are needed [23].

3.2. Methodology

Defect detection system described in this paper can be divided into two parts as learning phase (offline) and application phase (online). All images to be used in the experiments are taken from the TILDA [24] database and the properties of the database are given in the *Implementation* section. In the learning phase, independent component filters are obtained from defect-free textile fabric image of the TILDA database for each textile type separately. The independent component filters will be used in the application phase to extract feature vectors of each overlapping sub-windows (or patches) of the incoming test image. We used the coefficients of the independent component filters (i.e., sources) as the feature vector. The feature vector of each patch can be computed from Eq. (3) once the matrix $\boldsymbol{A}$ or its pseudo-inverse $\boldsymbol{W}$ is constructed. The observation vector, $\boldsymbol{x}$, in Eq. (3) is the gray-scale values of the relevant sub-window that is centered and scaled by its variance.

The feature vectors of defect-free region have Laplacian distribution with zero mean and unit variance due to the ICA architecture and the preprocessing step [25]. This is because only defect-free textile images are used in the learning phase of the algorithm. Hence, for defect-free regions these feature vectors are rarely active and close to the zero vector. On the other hand, defective regions have different structure that is not included in the learning phase while we extract independent component filters. Therefore, feature vectors of defective regions deviate largely from the mean feature vector of the defect-free textile image. This enables us to detect these defective regions by just using the deviation from the mean with some automatic threshold value which depends on the spread of the data.

Summary of detection of defect:

(1) Compute Euclidean distance between the feature vector of each sub-window and vector s_{mean} (the mean of the

feature vectors in the learning phase):

$$\text{distance}_i = [(\boldsymbol{s}_{\text{mean}} - \boldsymbol{s}_i)^{\text{T}}(\boldsymbol{s}_{\text{mean}} - \boldsymbol{s}_i)]^{1/2}, \quad (9)$$

where $\boldsymbol{s}_i$ is the feature vector of the ith sub-window of incoming test image.

(2) Classify a sub-window as defective if distance exceeds some threshold value α:

$$\text{sub-window}_i = \begin{cases} \text{defective}, & \text{distance} > \alpha, \\ \text{non-defective}, & \text{otherwise.} \end{cases}$$

The automatic threshold value is determined by the following formula:

$$\alpha = D_m + \eta \text{IQR}, \quad (10)$$

where D_m is the median value of the Euclidean distances obtained by Eq. (9), IQR is the inter quartile range width and η is a constant determined experimentally. The inter-quartile range is found by calculating the difference in value between the upper quartile and the lower quartile of Euclidean distances, that is:

$$\text{IQR} = \text{upper quartile value} - \text{lower quartile value}. \quad (11)$$

The inter-quartile range is often used to give an idea of how widely the data values are spread out. Inter-quartile range is more robust than the statistical measures such as the range or the standard deviation of the data in case of existence of outliers in the data which makes IQR a better tool for automatic threshold generation as given in Eq. (10) [26].

The algorithm we have used has two basic parameters to be determined: (i) η factor in Eq. (10) for each textile type and (ii) the size of sub-windows. The η value for best performance can be determined by trial and error for each textile type separately. It is also important to decide on appropriate sub-window size for better defect detection performance, since feature vectors are calculated within local overlapping sub-windows of size $N \times N$. The choice of sub-window size depends on two factors: (i) how localized the defects are (i.e., size of the defects); and (ii) for a non-defective sample how representative of the texture is the data in a window of such size [27]. For determining the location of the defect, the window size should be as small as possible. However, if we choose a very small window size, we will lose the texture of the textile fabric and this will result in independent components, which do not represent the texture in that textile fabric image. Here we implement a new idea for determining the sub-window size by using statistical properties of the textile images. The original idea comes from watermarking where Mihcak et al. [28] tried to hide information in video applications and wanted to protect information against attacks such as rotation, translation, zooming or bending. They observed that applying watermarking to images using randomly selected overlapping windows enables one to have robustness in the statistics. In other words, if windows are randomly selected, statistics of the attacked and the original versions of the image will be rotation and translation invariant. Mihcak et al. also observed that as the window size increases, statistical difference between the attacked and the original versions of the image decreases and converges to a steady state value. This idea helps us to find the optimal window size in the analysis of the texture.

In ICA, statistical information about the multivariate data, which is the structure of the texture in our case, is acquired. We observed that when many randomly selected sub-windows are used, statistical information about the texture becomes invariant to rotation and translation. Furthermore, it can be seen from Fig. 2 that as the size of sub-windows gets larger, statistical difference between the original image and its translated and rotated versions decreases. Here, statistical difference is measured by finding mean-square error (MSE) between randomly selecting sub-windows from the original image and sub-windows in the same location of the rotated and translated image. The MSE is calculated for each sub-window and average of these MSE is taken and this process is repeated for different sub-window sizes. Positions of sub-windows will depend on the random seed used in the algorithm; in order to make algorithm seed independent we also averaged average-MSEs obtained from different seeds to give final result shown in Fig. 2(a) and (b). Since defective ($e1$–$e4$ classes) and non-defective ($e0$ class) images in TILDA database are obtained by a fix-positioned camera, scale invariance will not be considered. There are two competing forces in determining the optimal window size: If the window size is too small, the surface texture cannot be modeled accurately, however, if the window size is too large, then the defect may not significantly affect the feature vector. It can be observed from Fig. 2 that for window width that is greater than 30, statistical difference does not change significantly. For both translated and rotated versions the derivative of the average-MSE curve reduces below 0.1 between 30 and 40 sub-window sizes. In view of the fact that our images are multiples of 32 and 32 lies in the region where statistical difference starts to become stable, we have selected 32×32 pixel size sub-windows for our computations.

4. Implementation

The idea of using ICA as a tool for texture defect detection is tested using two different sets of images. The first one is an in-house dataset obtained by using a CCD camera and a roll of raw textile fabric that contains different defects faced in the textile industry and the other is the TILDA database. In this paper we will just give the results for the TILDA dataset, which became a common benchmark for defect detection studies for enabling comparison of the algorithms. Details of the in-house dataset performance can be found in Ref. [29].

TILDA data set has been created by the workgroup on texture analysis of Deutsche Forschungsgemeinschaft Germany (DFG) [24]. This database contains both figurative

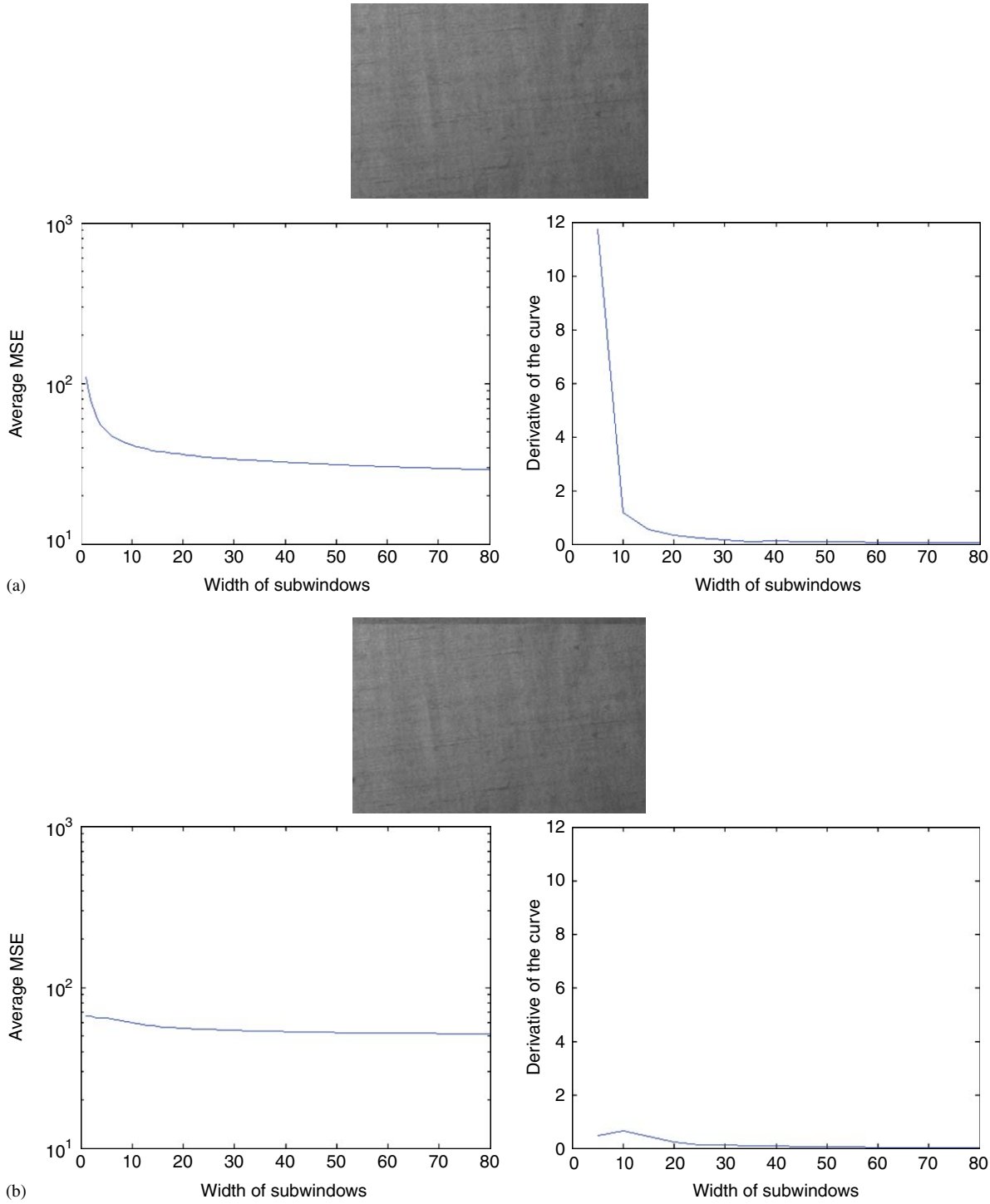

Fig. 2. A defect-free ($e0$ class) image from TILDA database is 5° rotated (a) and translated by 20 pixel rows (b). Their corresponding semi-log graphs showing average-MSE vs. width of sub-windows (left) and derivative of the average-MSE curve vs. width of sub-windows (right).

and raw textile images with or without defects. In the database, images are taken at arbitrary rotation and have 768×512 pixel size. We resize the images by a factor of 0.5 by using bicubic interpolation. Since our aim is to analyze texture defects rather than defects in the shapes or figures we exclude figurative textile images from our studies. Moreover, we believe that detection of the imperfections in the figurative images need to be explained by a higher-level visual process rather than receptive fields idea using independent components. There are four different types of raw textile images; two of them are at $c1$ and the other two at $c2$ textile groups. For each textile type there are four different defect classes ($e1$–$e4$) and one defect-free class ($e0$). All classes contain 50 images, for defective classes these

50 images contain both different types of defects and same defect taken at different rotation angles.

In the learning phase, independent components are extracted from defect-free samples of each textile type. The first step in the procedure is to make each defect-free image zero mean. After that observed data matrix $\boldsymbol{X}$ is formed by placing gray-level intensity values of the subwindows selected randomly from every image in the set of defect-free textiles into columns of $\boldsymbol{X}$. A total of 10 000 sub-windows are used to construct $\boldsymbol{X}$, which means $\frac{10\,000}{50} = 200$ sub-windows are selected randomly from each defect-free image. Since each sub-window has 32×32 pixel size converting it into a column vector gives a vector of size 1024, hence the size of $\boldsymbol{X}$ becomes $1024 \times 10\,000$. The next step is to reduce the dimension of $\boldsymbol{X}$ and whiten the observed data by means of PCA. In our algorithm, we select the 16 largest dimensions to remain; in order to eliminate over-learning, which will disturb our calculations. Also in this way, we have preserved more than 90% of the energy in the images.

After dimension reduction, estimation of ICA becomes easier. Number of iterations is another parameter, which should be considered in the computation of the independent components. The fix-point algorithm that we use in analysis is known for its fast convergence time and thus small number of iterations should be sufficient. Since learning phase in done offline we use a high number of iterations (1000 iterations) for achieving best results experimentally. FastICA algorithm [21] with $\tanh(x)$ is used to estimate independent components. Once the matrix $\boldsymbol{A}$ and its (pseudo) inverse $\boldsymbol{W}$ are constructed, the feature vectors can be computed using

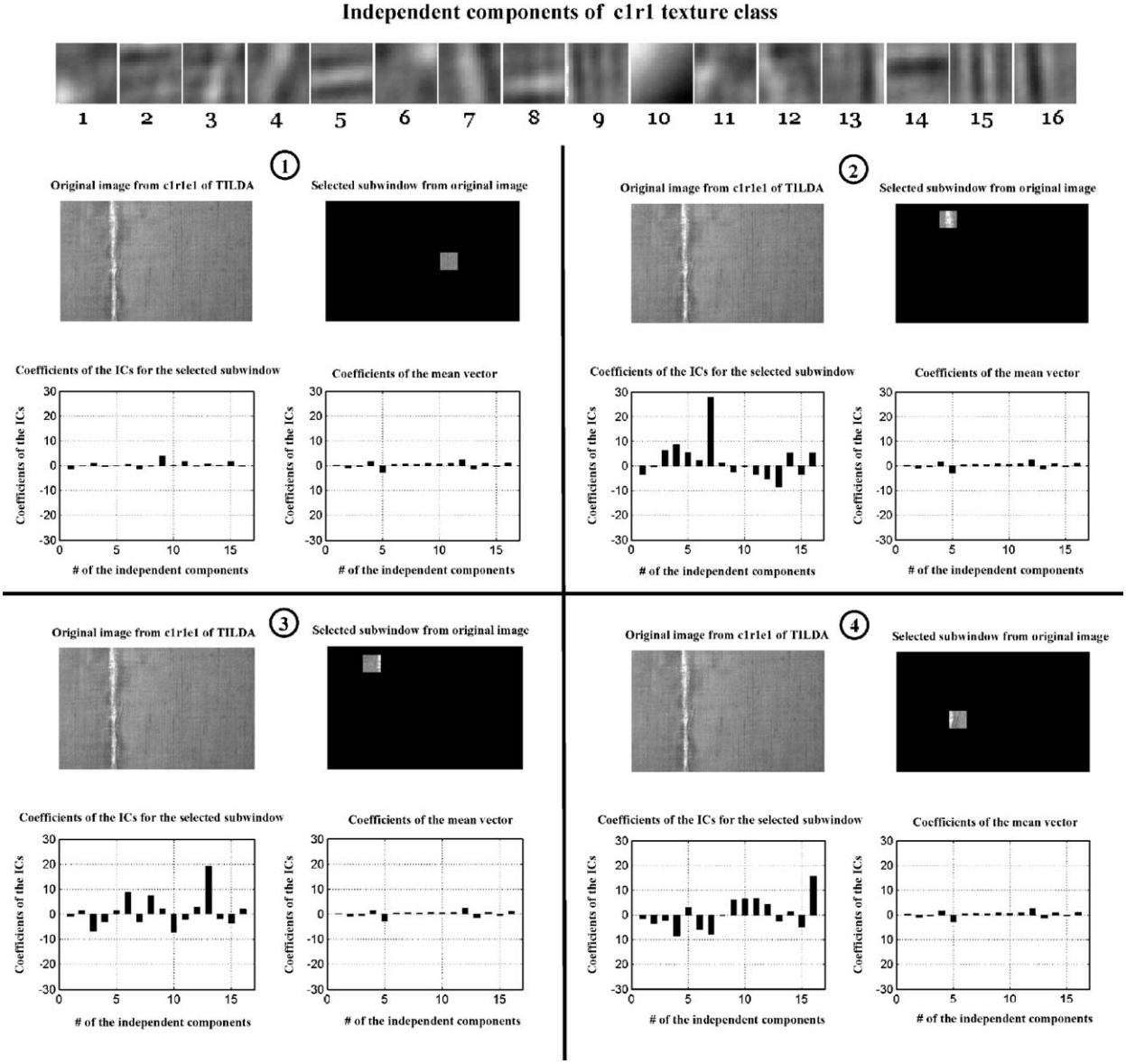

Fig. 3. Demonstration of extracting features as coefficients of the independent components obtained from $c1r1$ texture.

Eq. (3). Here $\boldsymbol{x}$ is a column vector of the sub-window coming from the test image and $\boldsymbol{s}$ is the corresponding feature vector of the sub-window (i.e., coefficients of independent components). This step is repeated for every sub-window in the test image. It is important to note that, since we normalize the observed data to zero mean and unit variance before obtaining the independent components, the same process is repeated for $\boldsymbol{x}$ vector prior to feature vector $\boldsymbol{s}$ is extracted. Sub-windows from a test image are obtained sequentially with the half-subwindow-size overlap in both vertical and horizontal directions. In this fashion except those in the border of the image, every point in the image appears in four different and neighboring sub-windows and that reduces the probability of missing a defect.

Defect detection, in application phase (online) is done by classifying a sub-window according to its Euclidean distance to the mean feature vector. Indication of defect resides in the distance information between the test and the mean feature vectors. If the distance exceeds a certain threshold value, that sub-window is labeled as defective. The threshold value depends on η as shown in Eq. (10), and for different textile types it is expected to find different η values for best performance. In this paper, performance of the defect detection system is demonstrated with examples rather than reporting detection and false alarm rates since in the TILDA database the exact position and the extent of the defects are not given.

5. Results and discussions

In order to give intuition about how the algorithm works, we will first show how we obtain the coefficients of

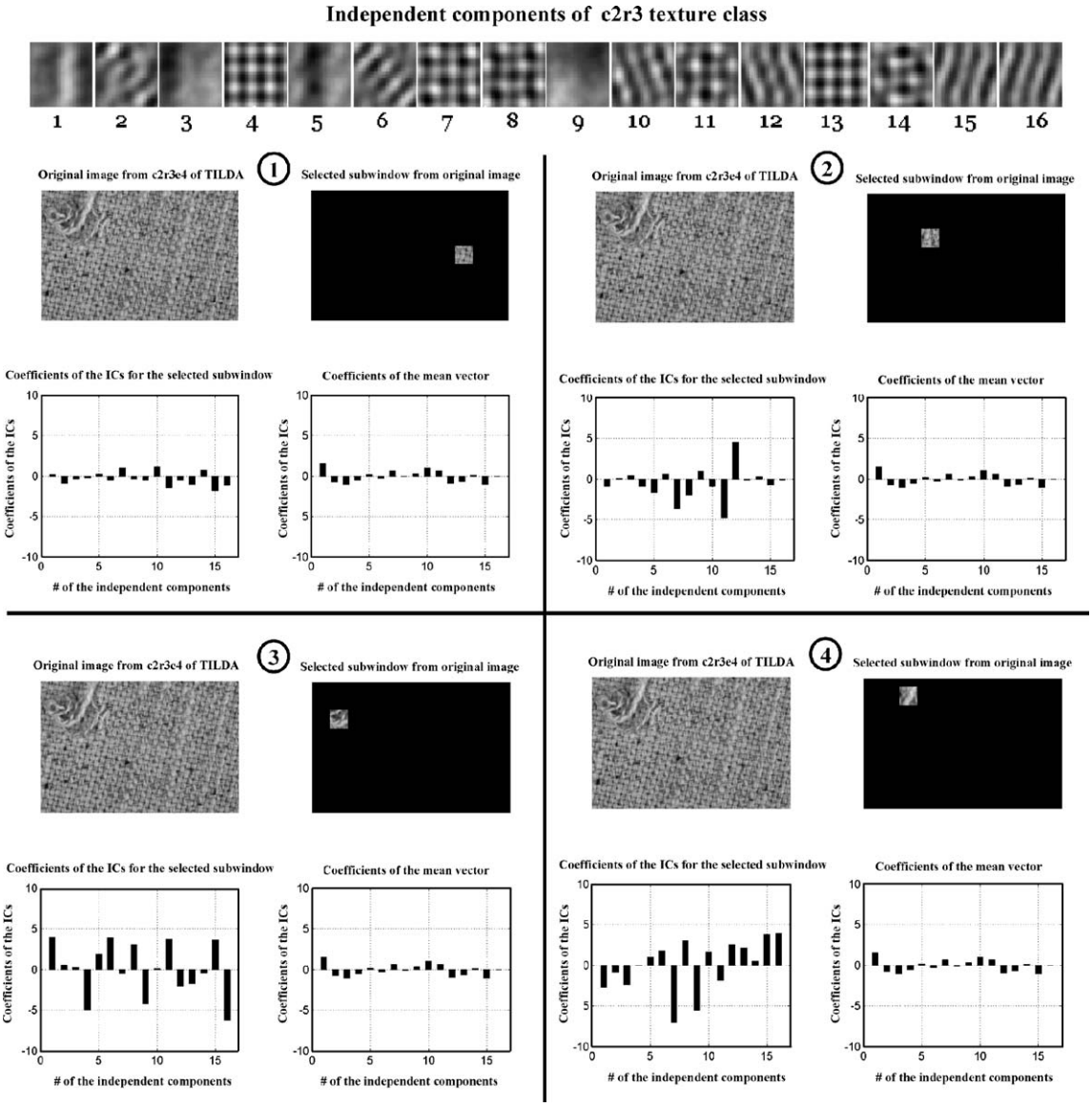

Fig. 4. Demonstration of extracting features as coefficients of the independent components obtained from $c2r3$ texture.

independent components of the selected sub-windows. In Figs. 3 and 4 the idea is demonstrated by using two different textile types; one has low regularity and dominant orientation field ($c1r1$) and the other with higher regularity than orientation field ($c2r3$). At the top of each figure, the corresponding independent components extracted from that texture are given. These figures illustrate results for selection of four different sub-windows, one being from a defect-free region, and the rest from different locations of the defect. Note that in the second case of the Fig. 3, seventh-independent component which corresponds to a centered bar has the highest coefficient. In fact this result matches with the selected sub-window in the second case. Similar matches exist for the third and fourth cases.

If we refer to Fig. 4, we observe that coefficients of independent components of the defect-free sub-window, shown in first case, are close to zero and 11th, 15th and 16th-independent components have relatively high values, which is in accordance with the structure of the texture. On the other hand, in those sub-windows corresponding to selected regions of the defect have different response patterns. Moreover, in the second case, we observe that 12th independent component has the largest coefficients which signals a change in orientation and periodicity whereas in the third and fourth cases an irregular response pattern which contains high coefficient values among the independent components that represent differing orientations and periodicity or those that do not have any periodicity information at all.

In Figs. 5–12 for each textile group first independent components that are extracted from non-defective images are presented, then for each group, defect detection results are given from defective image classes, $e1$–$e4$. These images are selected in order to have a good representation of the defect types in each class. It is important to note that the independent component filters or equivalently called independent components of each textile group differ with respect to each other according to the structure of the textile. Hence, ICA enables us to analyze the textile with its own structural elements that are represented by the extracted components.

In order to understand the reason why $c1r1$ and $c1r3$ have independent components that look like orientation-selective bar cells (center-on surround-off or center-off surround-on) whereas $c2r2$ and $c2r3$ have independent components that resemble grating cells we should refer to the structural composition of the textile groups. If we refer to the independent components of $c1r1$, we observe that they are similar to bar cells because $c1r1$ has low-regularity pattern with quite distinct orientation, which is what we expect to have. Similarly, although $c1r3$ has higher regularity, orientation of the texture is still dominant which results in mostly bar cell like independent components with a few of them looking similar to definition of grating cell. On the other hand, regularity of $c2r2$ and $c2r3$ is higher due to their grid-like structures. Hence regularity is dominant feature for these textile groups and that generates independent components, which are mostly similar to grating cells. Refer to Ref. [30] for detail explanation of regularity and local orientations of textures. Of course, these independent components do not encompass all the possible orientation of the texture pattern. Since we have used dimension reduction, only a limited number of dominant orientations appeared in the independent components. Nevertheless, we are able to detect defects even with this limited number of orientations.

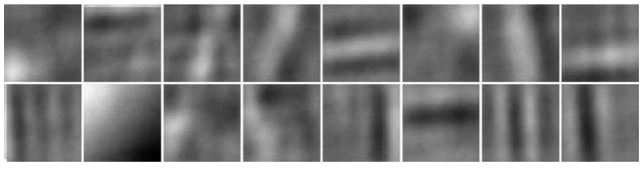

Fig. 5. Independent components of $c1r1$ texture class from non-defective images.

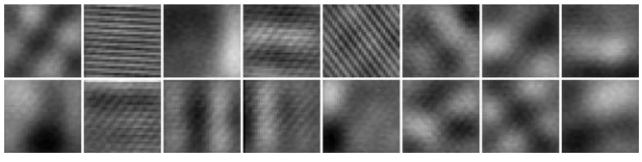

Fig. 7. Independent components of $c1r3$ texture class from non-defective images.

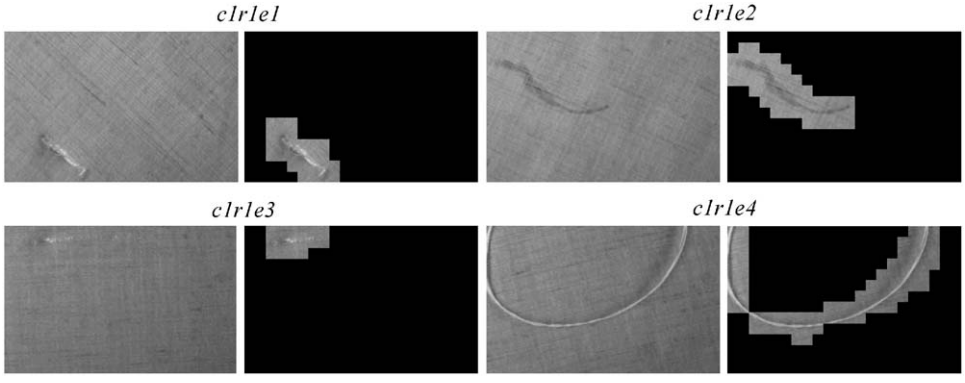

Fig. 6. Detection results for TILDA images drawn from $c1r1$ texture class, $\eta = 2.5$.

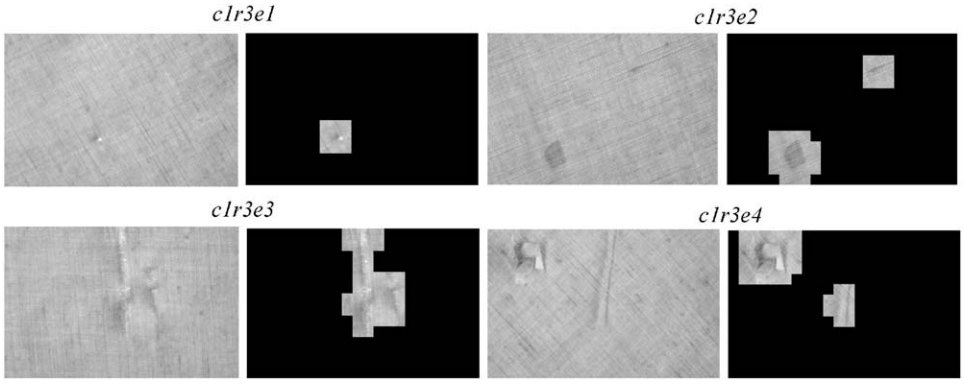

Fig. 8. Detection results for TILDA images drawn from $c1r3$ texture class, $\eta = 2.5$.

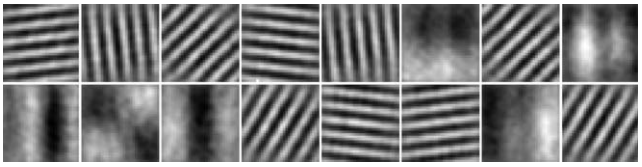

Fig. 9. Independent components of $c2r2$ texture class from non-defective images.

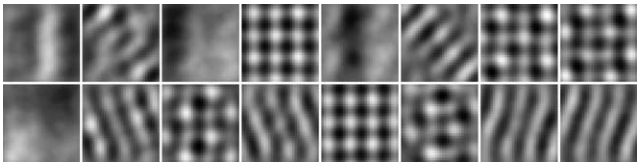

Fig. 11. Independent components of $c2r3$ texture class from non-defective images.

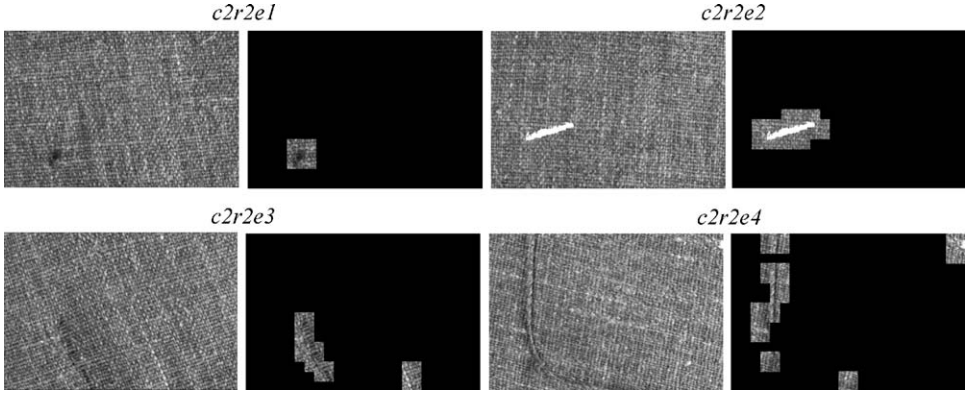

Fig. 10. Detection results for TILDA images drawn from $c2r2$ texture class, $\eta = 2$.

Quantitative results of the algorithm are given in Table 1. The results of the algorithm are given for defective classes ($e1$–$e4$) and defect-free class ($e0$) of each textile type. The false alarm rate is evaluated only for non-defective textures since the location of the defects are not labeled for the other images. The $e0$ column gives the false alarm rate for a given η as the number of sub-windows identified as defective versus ($N_{\text{FA-S}}$) total number of sub-windows ($N_{\text{T-S}}$). The number of sub-windows used to analyze a single image is 345, hence total number of sub-windows used in the analysis of $e0$ class is $345 \times 50 = 17\,250$. For defective classes $e1$–$e4$, results are given as the number of images in that class when defect is not detected (N_{Miss}) versus total number of images in that class (N_{T}). Results for $c2r3e3$ are not given due to high number of misdetection, the reason for which will be explained below.

In general, the algorithm has higher performance for the first two textile types, i.e. $c1r1$ and $c1r3$. The structure of these texture types enables us to detect imperfections much easier, since they have relatively low regularity; both intensity and geometric defects cause light or dark blobs in the texture surface, which enable better performance. On the other hand, $c2r2$ and $c2r3$ textile groups have somewhat different characteristics, which cause reduction of the performance. For both $c2r2$ and $c2r3$, there are acceptable imperfections in their structure that are not regarded as defects. However, the algorithm treats these imperfections as defective regions and that causes an increase in false alarms as shown in Table 1 ($e0$ column of $c2r2$ and $c2r3$). Examining Fig. 13 shows the cases for these two textile types when disruptions in regularity are not accepted as defect. Especially in $c2r3$, the imperfections in regularity,

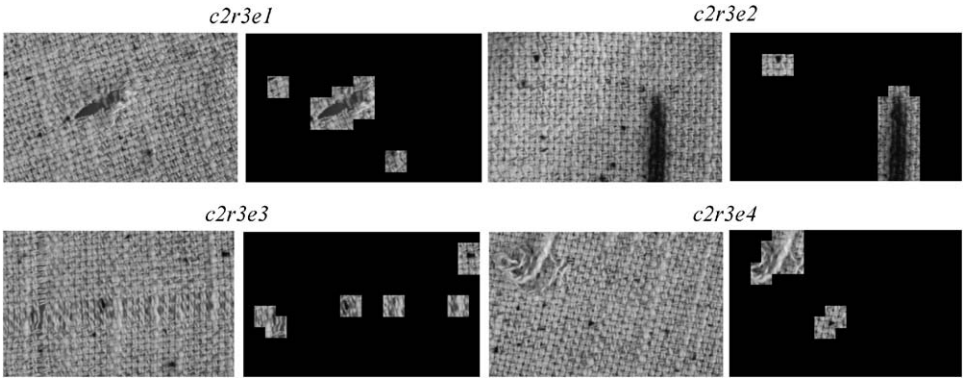

Fig. 12. Detection results for TILDA images drawn from $c2r3$ texture class, $\eta = 2$.

Table 1
Quantitative performance of the algorithm

	$e0$ ($N_{\text{FA-S}} : N_{\text{T-S}}$)	$e1$ ($N_{\text{Miss}} : N_{\text{T}}$)	$e2$ ($N_{\text{Miss}} : N_{\text{T}}$)	$e3$ ($N_{\text{Miss}} : N_{\text{T}}$)	$e4$ ($N_{\text{Miss}} : N_{\text{T}}$)
$c1r1$ ($\eta = 2.5$)	67:17 250	0:50	0:50	2:50	1:50
$c1r3$ ($\eta = 2.5$)	77:17 250	0:50	2:50	0:50	2:50
$c2r2$ ($\eta = 2.0$)	201:17 250	3:50	0:50	7:50	2:50
$c2r3$ ($\eta = 2.0$)	339:17 250	2:50	26:49	–	14:50

which are small holes between the stripes, reduce the performance. Although defects in $c2r3$ can be tracked from the coefficients of the independent component (such as defects in $c2r3e3$ class), the coefficients of these small holes suppress the detection of actual defects in the decision part of the algorithm. One way to eliminate this problem and increase the overall performance is to use the prior information given about the structure of the texture to build the decision algorithm. In other words, using a method such as support vector machine (SVM) to train the decision mechanism for defective and non-defective cases can possibly solve the problem.

There are two cases when the performance of the algorithm to locate the defect reduces: (i) when the size of the defects compared with size of the sub-windows is small and (ii) when defect causes minor perturbations in the intensity value. Fig. 14 shows two cases from $c2r3e4$ for case (i) and (ii) when detection in not possible.

Locating the entire defect is another possible issue. The algorithm has the flexibility to be tuned for particular appearance or geometry of defects by changing the η value. Fig. 15(a) and (b) bottom right images show the effect of reducing the η value which gives better performance for locating the defect in these cases. Of course an increase in the number of false alarms could arise for the rest of the images in that textile group if that η value is used. Rather than trial and error, methods like cross-validation can be used to determine η automatically.

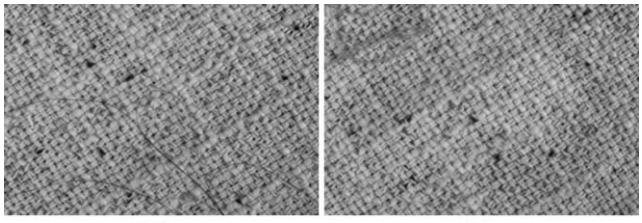

Fig. 14. Limitations of the algorithm due to: (a) small defect size (note the thin threads in the lower part of the image) and (b) minor intensity perturbations (defect is in the upper-left corner of the image).

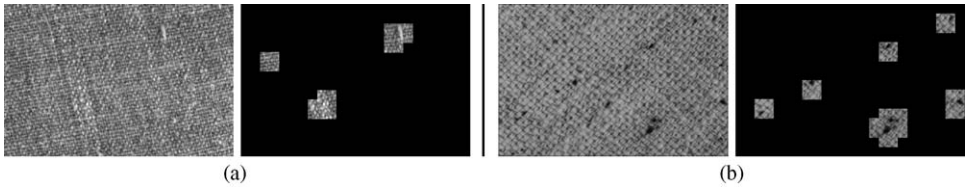

Fig. 13. Acceptable imperfections in the defect-free class of (a) $c2r2$, and (b) $c2r3$.

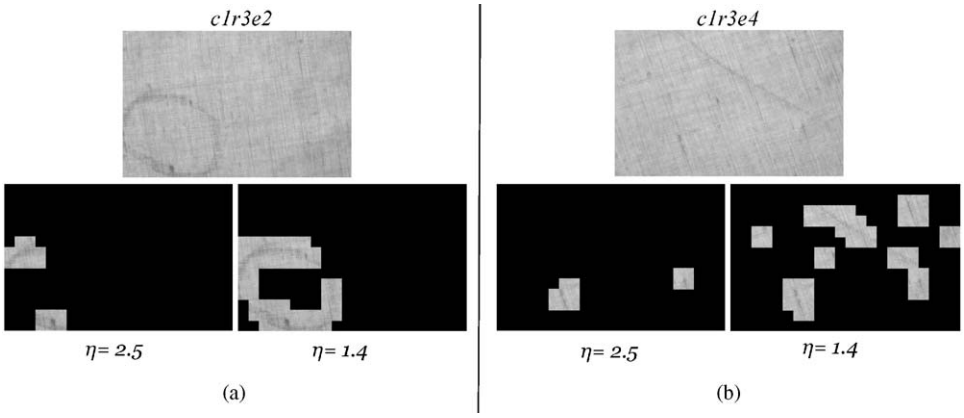

Fig. 15. Changing threshold value enables better detection performance.

Table 2
Computational complexity of the online stage of the algorithm

	Additions per image ($\times 10^6$)	Multiplications per image ($\times 10^6$)
ICA (online)	5.64	5.65

Computational complexity for online stage is given in Table 2 in terms of number of additions and number of multiplications for an image from the TILDA database which is scaled down to 256×384 pixels size (originally corresponds to $5 \times 7.5\,\text{cm}^2$ of the textile fabric). Two factors affect the computational computation: (i) size of the sub-windows and (ii) sub-window overlap factor. In the table, size of sub-windows is 32×32 pixels whereas overlap factor is 0.5, which corresponds to translation of half of the sub-window size. If non-overlapping sub-windows scheme is used (zero overlap factor) that will reduce complexity by a factor of 3.5.

6. Conclusion

Generative description of regularity and local anisotropy of the textures are obtained by ICA and used for defect detection purpose. Regularity feature of the texture brings about independent components composed of periodical oriented bars similar to periodic pattern selective grating cells. On the other hand, directionality (anisotropy) feature gives rise to single oriented bar like independent components similar to aperiodic pattern selective bar cells. The developed method makes use of these descriptions to capture imperfections in regularity and local anisotropy by tracking coefficients of independent components, which facilitates rotation and translation invariant defect detection.

References

[1] A.R. Rao, G.L. Lohse, Identifying high level features of texture perception, CVGIP Image Process. 55 (1993) 218–233.

[2] D. Chetverikov, A. Hanbury, Finding defects in texture using regularity and local orientation, Pattern Recognition 35 (2002) 203–218.

[3] D. Hubel, T. Wiesel, Receptive fields, binocular interaction and functional architecture in the cat's visual cortex, J. Physiol. (London) 160 (1962) 106–154.

[4] D. Hubel, T. Wiesel, Receptive fields and functional architecture of monkey striate cortex, J. Physiol. 195 (1968) 215–243.

[5] H.B. Barlow, Sensory mechanisms, the reduction of redundancy, and intelligence, The mechanisation of thought processes, H.M.S.O., London, (1959) 535–539.

[6] D.J. Field, What is the goal of sensory coding?, Neural Comput. 6 (1994) 559–601.

[7] B.A. Olshausen, D.J. Field, Emergence of simple-cell receptive field properties by learning a sparse code for natural images, Nature 381 (1996) 607–609.

[8] B.A. Olshausen, D.J. Field, Sparse coding with an overcomplete basis ser: a strategy employed by V1, Vision Res. 37 (1997) 3311–3325.

[9] A.J. Bell, T.J. Sejnowski, The independent components' of natural scenes are edge filters, Vision Res. 37 (1997) 3327–3338.

[10] J. Hurri, Independent component analysis of image data, MS Thesis, Helsinki University of Technology, Helsinki, Finland, 1997.

[11] J.H. van Hateren, A. van der Schaaf, Independent component filters of natural images compared with simple cells in primary visual cortex, Proc. R. Soc. London B 265 (1998) 359–366.

[12] G. Sclar, P. Lennie, D.D. DePriest, Contrast adaptation in striate cortex of macaque, Vision Res. 29-7 (1989) 747–755.

[13] M. Volgushev, T.R. Vidyasagar, X. Pei, A linear model fails to predict orientation selectivity of cells in the cat visual cortex, J. Physiol. (London) 496 (1996) 597–606.

[14] R. von der Heydt, E. Peterhans, M.R. Dürsteler, Periodic-pattern-selective cells in monkey visual cortex, J. Neurosci. 12 (1992) 1416–1434.

[15] N. Petkov, P. Kruizinga, Computational models of visual neurons specialized in the detection of periodic and aperiodic oriented visual stimuli: bar and grating cells, Biol. Cybern. 76 (1997) 83–96.

[16] C. Jutten, J. Herault, Blind separation of sources, Part I, An adaptive algorithm based on neuromimentic architecture, Signal Process. 14 (1991) 1–10.

[17] A. Hyvarinen, Survey on independent component analysis, Neural Comput. Survey 2 (1999) 94–128.

[18] A. Hyvarinen, E. Oja, P. Hoyer, J. Hurri, Image feature extraction by sparse coding and independent component analysis, Helsinki University of Technology, Finland (ICPR 1998).

[19] P.O. Hoyer, A. Hyvärinen, Independent component analysis applied to feature extraction from colour and stereo images, Network Comput. Neural Systems 11-3 (2000) 191–210.
[20] J.J. Attick, A.N. Redlich, Convergent algorithm for sensory receptive field development, Neural Comput. 5 (1993) 45–60.
[21] A. Hyvarinen, Fast and robust fixed-point algorithms for independent component analysis, Neural Comput. 10-3 (1999) 634–636.
[22] A.L. Amet, Texture defect detection using wavelet transforms, MS Thesis, Boğaziçi University, Istanbul, Turkey, 1997.
[23] B. Aras, A. Ertüzün, A. Erçil, Higher order statistics based texture analysis method for defect inspection of textile products, in: Proceedings of the IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing (NSIP99), Antalya, Turkey, 20–23 June 1999, pp. 858–862.
[24] Workgroup on Texture Analysis of DFG. TILDA Textile Texture Database ⟨http://lmb.informatik.uni-freiburg.de/research/dfg-texture/tilda⟩.
[25] M.S. Bartlett, J.R. Movellan, T.J. Sejnowski, Face recognition by independent component analysis, IEEE Trans. Neural Networks 13 (6) (2002) 1450–1464.
[26] J. Tukey, Exploratory Data Analysis, Addison Wesley, Reading, MA, 1977.
[27] F.S. Cohen, Z. Fan, S. Attali, Automated inspection of textile fabrics using textural models, IEEE Trans. Pattern Anal. Mach. Intell. 13-8 (1991) 803–808.
[28] M.K. Mihcak, R. Venkatesan, M. Kesal, Watermarking via optimization algorithms for quantizing randomized statistics of image regions, Proceedings of the 40th Annual Allerton Conference on Communications, Computing and Control, Invited Paper, Monticello, IL, October 2002.
[29] O.G. Sezer, A. Ertüzün, A. Erçil, Independent component analysis for texture defect detection, Pattern Recognition Image Anal. 14-2 (2004) 303–307.
[30] D. Chetverikov, Fundamental structural features in the visual world, Proceedings of the Workshop Fundamental Structural Properties in Image and Pattern Analysis, Budapest, 1999, pp. 47–58.