LOCALISING SURFACE DEFECTS IN RANDOM COLOUR TEXTURES USING MULTISCALE TEXEM ANALYSIS IN IMAGE EIGENCHANNELS

Xianghua Xie and Majid Mirmehdi

Department of Computer Science,
University of Bristol, Bristol BS8 1UB, England
{xie,majid}@cs.bris.ac.uk

ABSTRACT

A novel method is presented to detect defects in random colour textures which requires only a very few normal samples for unsupervised training. We decorrelate the colour image by generating three eigenchannels in each of which the surface texture image is divided into overlapping patches of various sizes. Then, a mixture model and EM is applied to reduce groupings of patches to a small number of textural exemplars, or texems. Localised defect detection is achieved by comparing the learned texems to patches in the unseen image eigenchannels.

1. INTRODUCTION

Methods such as cooccurrence matrices and Gabor filters have been widely applied for defect detection, but principally on regular and periodic textures [1, 2, 3]. In [4], Varma and Zisserman argued that a large variety of signals can be analysed by just looking at small neighbourhoods. They used 7×7 patches to generate a texton representation and achieved better performance than filtering methods when classifying material surface images. The results demonstrated that textures with global structures can be discriminated by examining the distribution of local measurements. In [5], the authors also advocated the use of local pixel neighbourhood processing in the shape of local binary patterns as texture descriptors. Other works based on local pixel neighbourhoods are those which apply Markov Random Field (MRF) models, e.g. [6], where the inspection process was treated as a hypothesis testing problem on the statistics derived from the GMRF model.

In a random texture application such as ceramic tile production, the images under inspection may appear different in textural pattern from one to another, but their visual impression remains consistent, (see Fig. 1). In other words, there exist textural primitives that impose consistency within the product line. Instead of recovering all the variations amongst images from a relatively large number of samples in a supervised manner for a traditional classification approach, e.g. as in [7], we recently proposed a new approach to detect and localise defects on random greylevel textured surfaces [8]. We learned, in unsupervised fashion, textural primitive information from a small number of training samples within a Gaussian mixture modelling (GMM) and Expectation Maximisation (EM) framework. We named these representations *texture exemplars* or *texems* [8]. In this paper we extend our work by generating colour texems for detecting defects in randomly textured colour surfaces. This is necessary as some defects are chromatic by nature. A typical approach would be to directly generate texems from each RGB channel independently, however, this proved to miss defects as the channels are indeed correlated. Instead, we first build a reference eigenspace from a small sample set of defect-free textures which we then use to project the RGB channels of a training image into to generate three uncorrelated eigenchannel images. We then learn texems in these eigenchannels. By projecting the testing images into the same reference eigenspace, we can compare the textures in the resulting eigenchannels against the texems to localise defects. This process is shown schematically in Fig. 2 and will be described in detail in the next section.

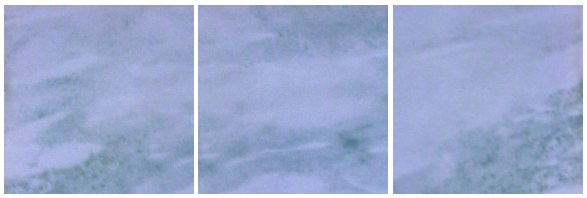

Fig. 1. Example marble tiles from the same family whose patterns are different but visually consistent.

Some defect types are unpredictable and occur only during production. Hence, we perform novelty detection, important from this practical view-point, based on the texems and the (automatically derived) lower bounds likelihoods of defect-free samples. To save computational costs we also implement the overall method in a multiscale framework.

This work is funded by EC project G1RD-CT-2002-00783 MONOTONE, and X. Xie is partly funded by the ORSAS, UK.

2. PROPOSED METHOD

Similar to the work of Jojic et al. [9], we consider an image as a collection of overlapping patches of various sizes. As the images of the same product line contain similar textural elements (see Figure 1), one image can be generated from the patches extracted from other images. Thus, for a few given samples we can easily obtain a large number of patches of various sizes (which can in turn generate a large set of new images with the same visual impression). However, it is computationally prohibitive to perform defect detection based on such a large number of patches. Also, the patches themselves contain lots of redundant information. We can reduce the number of patches by learning a relatively small number of primitive representatives, i.e. texture exemplars or texems [8].

2.1. Computing the reference eigenspace

To extend our texture exemplars to colour images, the computational needs of a 3D texems model would be phenomenal. Instead, we perform PCA to transform the red, green, and blue channels into three decorrelated channels for better representation. As mentioned above the patterns on each image within the same texture family can still be different, hence the individually derived principal components can also differ from one image to another. Furthermore, defective regions can affect the principal components resulting in different eigenspace responses from different training samples. Thus, instead of performing PCA on each training image separately, we derive a single eigenspace from several training images; a reference eigenspace in which normal samples are represented (see Fig. 2). All the images under inspection will be projected onto this eigenspace. Thus, the transfomed channels share the same principal components.

Each colour pixel is denoted as $x_i = [r_i, g_i, b_i]^T$. Let $X = \{x_i \in \mathcal{R}^3, i = 1, 2, ..., q\}$ be the set of q 3D vectors made up of the pixels from several defect free samples, e.g. 5 in this work. Let $\bar{x} = \frac{1}{q}\sum_{x \in X} x$ be the mean vector of X. Then, PCA is performed on the mean-centred colour feature matrix X to obtain the eigenvectors $E = [e_1, e_2, e_3]$, $e_j \in \mathcal{R}^3$. Singular Value Decomposition can be used to obtain these principal components. The colour feature space determined by these eigenvectors are referred to as the reference eigenspace $\Phi_{\bar{x},E}$, where the colour features are fully represented. The tile images are then projected onto this reference eigenspace $\Phi_{\bar{x},E}$:

$$X' = \overrightarrow{PCA}(X, \Phi_{\bar{x},E}) = E^T(X - \bar{x}J_{1,q}), \qquad (1)$$

where $J_{1,q}$ is a $1 \times q$ unit matrix consisting of all 1s. The texems are then learned in the resulting three eigenchannels.

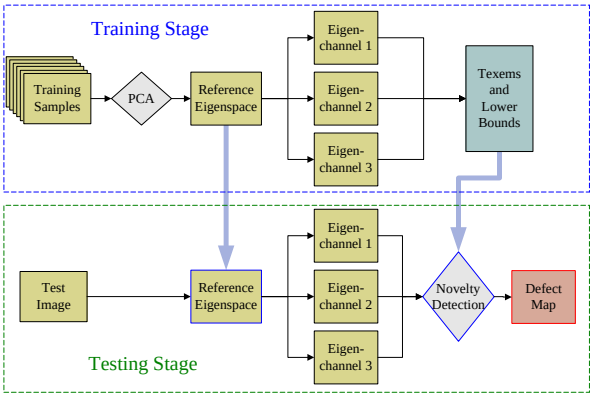

Fig. 2. Flow chart of proposed method.

2.2. Generating texems

The texems are texture exemplars at various sizes that encapsulate the textural content of a given image. Each texem, $\mathbf{m}$, is defined by a mean, $\boldsymbol{\mu}$, and a corresponding covariance matrix, $\boldsymbol{\omega}$, i.e. $\mathbf{m} = \{\boldsymbol{\mu}, \boldsymbol{\omega}\}$. The original image $\mathbf{I}$ is broken down into a set of P patches $\mathbf{Z} = \{\mathbf{Z}_i\}_{i=1}^{P}$, each containing pixels from a subset of image coordinates. The shape of the patches can be arbitrary, but we used square patches ($d = N \times N$) which may overlap and be of various sizes, e.g. as small as 5×5 to as large as required (here 20×20). We assume that there exist K texems, $\boldsymbol{\mathcal{M}} = \{\mathbf{m}_k\}_{k=1}^{K}$, $K \ll P$, for image $\mathbf{I}$ such that each patch in $\mathbf{Z}$ can be generated from a texem with certain added variations.

To learn these texems the P patches are projected into a set of higher dimensional spaces. The number of these spaces is determined by the number of different patch sizes and their dimensions are defined by the corresponding value of d. Each pixel position contributes one coordinate of a space. Each point in a space corresponds to a patch. Then each texem represents a class of patches in the corresponding space. We assume each class is a multivariate Gaussian distribution with mean $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\omega}_k$, which corresponds to $\mathbf{m}_k$ in the spatial domain. Thus, the probability density function for a particular patch $\mathbf{Z}_i$ given that it belongs to the kth texem $\mathbf{m}_k$, is:

$$p(\mathbf{Z}_i|\mathbf{m}_k, \theta) = \frac{1}{\sqrt{(2\pi)^d|\boldsymbol{\omega}_k|}}\exp\{-\frac{1}{2}(\mathbf{Z}_i - \boldsymbol{\mu}_k)^T\boldsymbol{\omega}_k^{-1}(\mathbf{Z}_i - \boldsymbol{\mu}_k)\}, \qquad (2)$$

where $\theta = \{\boldsymbol{\alpha}_k, \boldsymbol{\mu}_k, \boldsymbol{\omega}_k\}_{k=1}^{K}$ is the parameter set containing $\boldsymbol{\alpha}_k$, which is the *prior* probability of kth texem constrained by $\sum_{k=1}^{K}\boldsymbol{\alpha}_k = 1$. The parameter set θ can be determined first by marginalising the joint distribution by summing across the texems, and then optimising the data log-likelihood expression of the entire set $\mathbf{Z}$, given by

$$\log p(\mathbf{Z}|K, \theta) = \Sigma_{i=1}^{P}\log(\Sigma_{k=1}^{K}p(\mathbf{Z}_i|\mathbf{m}_k, \theta)\boldsymbol{\alpha}_k). \qquad (3)$$

Hence, the objective is to estimate the parameter θ for a given number of texems. The EM technique can be applied. That is to find $\hat{\theta}$ where

$$\hat{\theta} = \arg\max \log(\mathcal{L}(\theta|\mathbf{Z})) = \underset{\theta \in \Omega^{(K)}}{\arg\max} \log p(\mathbf{Z}|K, \theta). \quad (4)$$

The E-step involves a soft-assignment of each patch $\mathbf{Z}_i$ to texems, $\boldsymbol{\mathcal{M}}$, with a random initial guess or simple K-means clustering. We denote the intermediate parameters as $\theta^{(t)}$. The probability that patch $\mathbf{Z}_i$ belongs to the kth texem may then be computed using Bayes rule:

$$p(\mathbf{m}_k|\mathbf{Z}_i, \theta^{(t)}) = \frac{p(\mathbf{Z}_i|\mathbf{m}_k, \theta^{(t)})\boldsymbol{\alpha}_k}{\Sigma_{k=1}^{K} p(\mathbf{Z}_i|\mathbf{m}_k, \theta^{(t)})\boldsymbol{\alpha}_k}. \quad (5)$$

The M-step then updates the parameters by maximising the log-likelihood function as given in (4). The new estimates are denoted by $\hat{\boldsymbol{\alpha}_k}$, $\hat{\boldsymbol{\mu}_k}$, and $\hat{\boldsymbol{\omega}_k}$ where

$$\begin{aligned}
\hat{\boldsymbol{\alpha}_k} &= \frac{1}{P}\Sigma_{i=1}^{P} p(\mathbf{m}_k|\mathbf{Z}_i, \theta^{(t)}), \\
\hat{\boldsymbol{\mu}_k} &= \frac{\Sigma_{i=1}^{P} \mathbf{Z}_i p(\mathbf{m}_k|\mathbf{Z}_i, \theta^{(t)})}{\Sigma_{i=1}^{P} p(\mathbf{m}_k|\mathbf{Z}_i, \theta^{(t)})}, \\
\hat{\boldsymbol{\omega}_k} &= \frac{\Sigma_{i=1}^{P} (\mathbf{Z}_i - \hat{\boldsymbol{\mu}_k})(\mathbf{Z}_i - \hat{\boldsymbol{\mu}_k})^T p(\mathbf{m}_k|\mathbf{Z}_i, \theta^{(t)})}{\Sigma_{i=1}^{P} p(\mathbf{m}_k|\mathbf{Z}_i, \theta^{(t)})}.
\end{aligned} \quad (6)$$

The E-step and M-step are iterated until the estimations are stabilises. Then, the texems can be easily obtained by projecting the parameters back to the spatial domain. Various sizes of texems can be used and they can overlap to ensure they capture sufficient textural characteristics.

2.3. Multiscale texems

The dimension of the space we transform patches $\mathbf{Z}$ into will increase dramatically as the patch size increases. This means that a very large number of samples and high computational costs are needed in order to accurately estimate the pdf in very high dimensional spaces. Therefore, instead of generating variable-size texems, we learn fixed size texems in a multiscale image pyramid. This results in (multiscale) texems with a very small size, e.g. 5×5. A simple multiscale approach by using a Gaussian pyramid is sufficient.

Let us denote $\mathbf{I}^{(n)}$ as the nth level of the pyramid, $\mathbf{Z}^{(n)}$ as image patches extracted from $\mathbf{I}^{(n)}$, l as the number of levels, and $S^{\downarrow}$ as the downsampling operator. We then have

$$\mathbf{I}^{(n+1)} = S^{\downarrow} G_{\sigma}(\mathbf{I}^{(n)}), \qquad \forall n, n = 1, 2, ..., l-1, \quad (7)$$

where G_{σ} denotes the Gaussian convolution. The finest scale layer is the original image, $\mathbf{I}^{(1)} = \mathbf{I}$. We then extract multiscale texems from the image pyramid using the method presented in the previous section. Similarly, let $\mathbf{m}^{(n)}$ denote the nth level of multiscale texems and $\theta^{(n)}$ the parameters associated at the same level, which will then be used for novelty detection at the corresponding level of the pyramid.

During the EM process, the stabilised estimation of a coarser level is used as the initial estimation for the finer level, i.e. $\hat{\theta}^{(n,t=0)} = \theta^{(n+1)}$, which helps speed up the convergence and achieve a more accurate estimation.

2.4. Novelty detection and defect localisation

Once the texems are obtained, we can work out automatic minimum bounds of normal samples at each resolution level across the eigenchannels for novelty detection in unseen images. A small set of defect free samples (e.g. 4 or 5 only) are arranged within a multiscale framework using (7), and patches with the same texem size in each eigenchannel are extracted. The probability of a patch $\mathbf{Z}_{i,e_j}^{(n)}$ in the eigenchannel e_j belonging to texems of the same channel in the corresponding nth scale is:

$$p(\mathbf{Z}_{i,e_j}^{(n)}|\theta_{e_j}^{(n)}) = \Sigma_{k=1}^{K} p(\mathbf{Z}_{i,e_j}^{(n)}|\mathbf{m}_{k,e_j}^{(n)}, \theta_{e_j}^{(n)})\boldsymbol{\alpha}_{k,e_j}^{(n)}, \quad (8)$$

where $p(\mathbf{Z}_{i,e_j}^{(n)}|\mathbf{m}_{k,e_j}^{(n)}, \theta_{e_j}^{(n)})$ is a multivariate Gaussian distribution function as shown in (2). The minimum probability of a patch $\mathbf{Z}_{i,e_j}^{(n)}$ at level n, channel e_j across the training images is treated as the lower bound of the data likelihood, denoted as $\Lambda_{e_j}^{(n)} = \min(p(\mathbf{Z}_{i,e_j}^{(n)}|\theta_{e_j}^{(n)})), \forall\ \mathbf{Z}_{i,e_j}^{(n)} \in \mathbf{Z}_{e_j}^{(n)}$. This completes the training stage.

In the testing stage, the image under inspection is again layered into a multiscale framework and patches at each pixel position (x, y) at each level n are examined against the learned texems. The probability for each patch is then calculated according to (8) and compared to the lower bound, $\Lambda_{e_j}^{(n)}$, at the corresponding level and channel. Let $Q_{e_j}^{(n)}(x, y)$ be the probability map at the nth resolution level. Then, the potential defect map, $\mathcal{D}_{e_j}^{(n)}(x, y)$, at level n is:

$$\mathcal{D}_{e_j}^{(n)}(x, y) = \begin{cases} 0 & \text{if } Q_{e_j}^{(n)}(x, y) \geq \Lambda_{e_j}^{(n)} \\ \Lambda_{e_j}^{(n)} - Q_{e_j}^{(n)}(x, y) & \text{otherwise.} \end{cases} \quad (9)$$

We need to combine the information coming from all the resolution levels and channels to build the certainty of the defect at position (x, y). Similar to [1], we assume that a defect must appear in at least two adjacent resolution levels for it to be certified as such. Using a logical AND, implemented through the geometric mean, of every pair of adjacent levels, we initially obtain a set of combined pairs as:

$$\mathcal{D}_{e_j}^{(n,n+1)}(x, y) = [\mathcal{D}_{e_j}^{(n)}(x, y)\mathcal{D}_{e_j}^{(n+1)}(x, y)]^{1/2}. \quad (10)$$

This operation reduces false alarms and yet preserves most of the defective areas. Then, the defect candidates from each eigenchannel and each pair are combined using logical OR, as the arithmetic mean, to provide a final map for

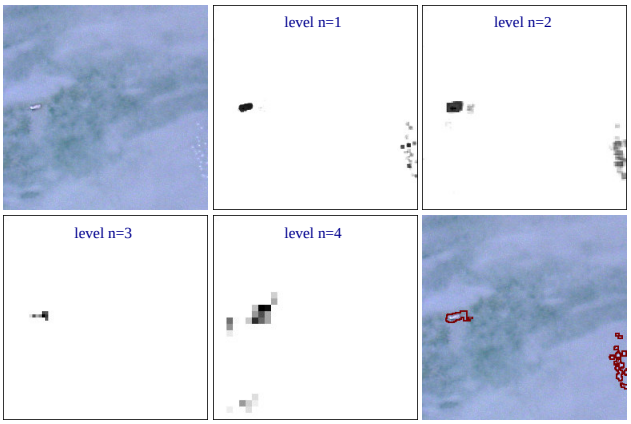

Fig. 3. Localising textural defects - from top left to bottom right: original defective tile image, detected defective regions at different levels $n = 1, 2, ..., 4$, and the final defective regions superimposed on the original image.

the defects detected across all the scales:

$$\mathcal{D}(x,y) = \frac{1}{3(l-1)} \sum_{n=1}^{l-1} \sum_{j=1}^{3} \mathcal{D}_{e_j}^{(n,n+1)}(x,y). \qquad (11)$$

3. EXPERIMENTAL RESULTS

The data set consisted of five different families of textures (i.e. images of the surface of 405 tiles) with different types of defects. In our experiments, only 5 good samples were used to extract the texems and the lower bound likelihoods. The number of texems at each resolution level were empirically set to 12, at 5×5 pixels each. The number of multiscale levels was $l = 4$. These parameters were fixed throughout the experiments.

Fig. 3 shows a random texture example, from the same family of tiles as in Fig. 1, with a physical defect on the left side and clusters of pin holes in the lower right region. The potentially defective regions detected at each resolution level across the eigenchannels are also shown, demonstrating that the texems have good sensitivity to defective regions at different scales. As the resolution progresses from coarse to fine, additional evidence for the defective region is gathered, e.g. the pin holes are missing in the defect maps of level 3 and 4 but are detected at finer scales. Also, a few false alarms at level 4 have no further support at other scales. This shows the combination rules can eliminate false positive regions from the final error map, which is shown superimposed on the original image. Three more examples are shown in Fig. 4, where the proposed method correctly detected both subtle individual pin holes and printing defects. The proposed method obatined 93.80% sensitivity, 90.22% specificity, and 91.36% overall accuracy across the data set.

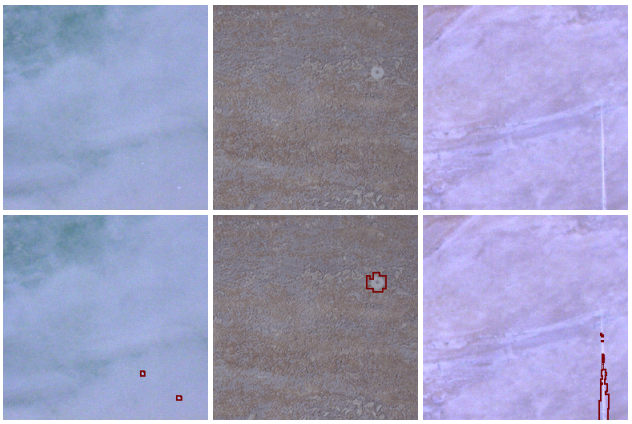

Fig. 4. Defect localisation on random texture examples.

4. CONCLUSIONS

We presented an automatic defect detection and localisation algorithm for random colour texture surfaces which extends the work in [8]. The proposed method requires training on very few defect-free samples. It decorrelates the training image colour bands and generates three independent eigenchannels in which texems are derived using a Gaussian density mixture model. The texems were then applied within a novelty detection framework to segment defects on unseen surfaces.

5. REFERENCES

[1] J. Escofet, R. Navarro, M. Millán, and J. Pladellorens, "Detection of local defects in textile webs using Gabor fi lters," *Opt. Eng.*, vol. 37, no. 8, pp. 2297–2307, 1998.

[2] D. Tsai and Y. Tsai, "Defect detection in textured surfaces using color ring-projection correlation," *Machine Vision Applications*, vol. 13, no. 4, pp. 194–200, 2003.

[3] I. Rossi, M. Bicego, and V. Murino, "Statistical classifi cation of raw textile defects," in *ICPR*, 2004, pp. 311–314.

[4] M. Varma and A. Zisserman, "Texture classifi cation: Are fi lter banks necessary?," in *CVPR*, 2003, pp. 691–698.

[5] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution grayscale and rotation invariant texture classifi cation with local binary patterns," *PAMI*, vol. 24, no. 7, pp. 971–987, 2002.

[6] F. Cohen, Z. Fan, and S. Attali, "Automated inspection of textile fabrics using textural models," *PAMI*, vol. 13, no. 8, pp. 803–809, 1991.

[7] A. Kumar, "Neural network based detection of local textile defects," *PR*, vol. 36, no. 7, pp. 1645–1659, 2003.

[8] X. Xie, M. Mirmehdi, and B. Thomas, "Texture exemplars for defect detection on greylevel random textures," submitted to ICAPR, 2005.

[9] N. Jojic, B. Frey, and A. Kannan, "Epitomic analysis of appearance and shape," in *IEEE ICCV*, 2003, pp. 34–42.