

DETECTION OF DEFECTS IN COLOUR TEXTURE SURFACES

J Kittler, R Marik, M Mirmehdi, M Petrou, J Song

University of Surrey
Guildford GU2 5XH, UK

ABSTRACT

An important area of automatic industrial inspection that has been largely overlooked by the recent wave of research in machine vision applications is the detection of defects in textured surfaces. In this paper, we present different algorithms for the detection of surface abnormalities both in the chromatic and structural properties of random textures. We present very promising results on the detection of cracks, blobs, and chromatic defects in ceramic and granite tiles.

1 INTRODUCTION

In problems of automatic surface inspection, one often has to handle material surfaces which have the appearance colour texture information. The inspection problem is then to detect any deviations from the normal texture by automatic image analysis techniques. The abnormalities can be divided into two basic categories: blob-like, and thin structures, i.e. cracks. For blob-like or regional defects in colour texture, faults can be caused by either chromatic abnormalities, structural abnormalities, or both simultaneously. It is therefore of paramount importance that any texture representation method must exhibit sensitivity to both (chromatic and structural) types of perturbations.

Early texture studies divided textures into two main types: micro and macro textures. To describe or model them, there are many methodologies developed and a detailed survey can be found in a paper by Haralick [1]. Many existing texture descriptors are based on the view that texture is a regular pattern composed of repeated primitive elements. Unfortunately, most natural images do not generally show such regularity, although, in most cases, one can immediately identify different textured areas. Some prototypical examples of irregular textures which are known as random macro textures, are ceramic, marble and granite images, which are of particular concern in this paper. As a matter of fact, the placement of the primitives within these is purely random and highly irregular.

In terms of image texture representation, very few algorithms exploit chromatic properties of texture. Tan *et al.* [2] proposed a method using eight Discrete Cosine Transform (DCT) texture features extracted from each of the three RGB colour bands for classifying colour textures. This method is designed to be sensitive to local variations. However, for the case of colour macrottextures, it requires a very large macro window to compute the DCT texture features. It is also very computationally intensive. In an-

other approach [3] eight Discrete Cosine Transform texture features are computed on the intensity image only and they are augmented by six colour features derived from the colour histogram. In this method, the 3D colour histogram was approximated by three 1D histograms and the colour information embedded in these histograms was used for the colour features. Unfortunately, histograms only convey coarse global information. Hence, the colour information is not fully exploited by this method. In addition, more than three distinct colours are generally found in many image textures, suggesting that the data distribution may be multi-modal. Using three 1D histograms to approximate the 3D histogram is therefore inappropriate. Caelli *et al.* [4] recently proposed a method which estimates colour texture features individually from three spectral channels by using multi-scale isotropic filters. The filters extract the first order statistics from the source colour histogram and second order statistics, amplitudes and orientations from the filtered response histograms in each channel separately. However, for the case of random macro colour textures, the orientation information is not significant. Thus, the extracted colour texture features do not adequately represent such textures.

Motivated by these considerations, we are presenting a novel algorithm to tackle the problem of surface inspection on random macro colour textures, in particular, granite images. The basic idea of this algorithm is to represent the random macro colour texture by means of colour texture features. These features are extracted from various chromatic classes associated with the colour image texture, rather than from the RGB bands as in Caelli's method [4].

The other category of defects which are also difficult to detect are small pin-holes, and short and long-length surface cracks on randomly textured ceramic and granite surfaces. There has been almost no report of any investigation of this kind. Since cracks or scratches usually occupy only 1% or less of the surface of an object, local methods for the analysis of image texture are important to capture the local information content.

One of the most sophisticated approaches to texture is that based on the Wigner distribution where the attributes computed for each pixel encapsulate both the local spectral and phase properties of the local Fourier transform in a unique real spectrum. Thus, they achieve a spatial/spatial frequency representation of the texture pattern. This approach is based on neurophysiological studies that support the view that representation in the human vision system involves a Fourier like decomposition of the visual stimulus into spatial frequency components [5]. These studies had a seminal influence on the development of spectral represen-

tation of texture in the form of either the energy of the outputs of a bank of filter tuned to different spatial frequency bands [6, 7] or the power spectrum itself. The Wigner distribution provides a spectral representation of texture which enjoys the highest resolution both in the spatial and spatial frequency domain. As a result, it is very sensitive to localised deviations from nominal texture properties such as those caused by cracks and pin-hole defects. For this reason it has been adopted and developed to provide a tool for the detection of such defects. We do not use any chromatic information in crack defect detection since short or long cracks are well characterised by their structural features. In any case the chromatic properties of cracks are irrelevant.

Next in section 2, we start by considering the Wigner distribution approach and associated post-processing optimal line filtering. In section 3 the colour texture defect detection algorithm is described including colour clustering, morphological smoothing, and structural analysis. Finally, results and conclusions are given in Sections 4 and 5 respectively.

2 WIGNER DISTRIBUTION REPRESENTATION

The short time Fourier transform is a commonly used method with which one can compute the frequency content of an image in the vicinity of a pixel by placing a window around it and taking the Fourier transform of the windowed function. The problem with this approach is that the Fourier transform produced this way is a complex array and usually only its magnitude (ie the spectrum) is used to associate a set of frequency domain features to each pixel. The Wigner distribution on the other hand, produces a real valued set of features which encapsulate both the magnitude and the phase information that characterise a signal in the frequency domain. This is achieved by creating first a symmetric function from the signal and taking its Fourier transform which, as a consequence, is real.

The Wigner distribution was initially defined as a co-joint time and time frequency representation of an infinite one dimensional signal [8, 9]. Its two dimensional extension suggested in [10] is defined by:

$$\begin{aligned} WD(x, y, \xi, \zeta) & \equiv \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f\left(x + \frac{\alpha}{2}, y + \frac{\beta}{2}\right) f^*\left(x - \frac{\alpha}{2}, y - \frac{\beta}{2}\right) \\ & \quad \exp[-j(\xi\alpha + \zeta\beta)] d\alpha d\beta \\ & = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x + \alpha, y + \beta) f^*(x - \alpha, y - \beta) \\ & \quad \exp[-j2(\xi\alpha + \zeta\beta)] d\alpha d\beta \end{aligned} \quad (1)$$

where $f(x, y)$ is a two dimensional image function, $f^*(x, y)$ its complex conjugate, ξ and ζ are the angular frequencies in the x and y directions respectively and α, β are some spatial displacement parameters.

In the above expression the image function $f(x, y)$ is treated like a continuous function. In reality of course, we have a sampled version of it from which the continuous

function must be reconstructed:

$$\begin{aligned} f_s(x, y) & = f(x, y) \text{comb}(x, y, \Delta x, \Delta y) \\ & = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f(m\Delta x, n\Delta y) \\ & \quad \delta(x - m\Delta x, y - \Delta y) \end{aligned} \quad (2)$$

where Δx and Δy are the sampling intervals. The discrete Wigner distribution then can be shown to be:

$$\begin{aligned} WD_s(x, y, u, v) & = \sum_{r=-\infty}^{\infty} \sum_{s=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \\ & \quad f(m\Delta x, n\Delta y) f[(r-m)\Delta x, (s-n)\Delta y] \\ & \quad \delta\left(x - r\frac{\Delta x}{2}, y - s\frac{\Delta y}{2}\right) \\ & \quad \exp\{-j2\pi[u(2m-r)\Delta x + v(2n-s)\Delta y]\} \end{aligned} \quad (3)$$

In the above expression u and v are frequencies.

Since the limits of integration in the definition of the Wigner distribution are infinite, it is almost uncomputable. Accordingly, Martin et al. [11] introduced a computable approximation to the Wigner distribution that they called the pseudo-Wigner distribution, the 2D extension of which is defined as:

$$\begin{aligned} PWD(x, y, \frac{\pi u}{P}, \frac{\pi v}{P}) & \equiv 4 \sum_{\alpha=-N}^N \sum_{\beta=-N}^N H(\alpha, \beta) \sum_{r=-M}^M \sum_{s=-M}^M g(r, s) \\ & \quad f(x + r + \alpha, y + s + \beta) f^*(x + r - \alpha, y + s - \beta) \\ & \quad \exp\left(\frac{-2\pi j(u\alpha + v\beta)}{P}\right) \end{aligned} \quad (4)$$

where

$$\begin{aligned} u, v & = 0, \pm 1, \dots, \pm N, \\ P & = 2N + 1, \end{aligned}$$

and $H(\alpha, \beta)$ and $g(r, s)$ are windowing functions and $(2N + 1) \times (2N + 1)$ and $(2M + 1) \times (2M + 1)$ are the sizes of the corresponding windows. It is desirable to choose windowing functions to eliminate or reduce the undesirable effects of aliasing and Gibbs phenomenon due to sampling and truncation. A windowing function in the Fourier domain should be a reasonable approximation of an impulse (delta) function with compromise between making the width of the delta function as small as possible and the amplitude of the ripple side lobes as small as possible. A prolate spheroidal wave function which is optimal in spectral energy within a specific bandwidth is the best candidate [12]. Kaiser [13] has shown that in the one dimensional case, the prolate spheroidal wave function may be well approximated by the modified Bessel function of zero order, appropriately scaled. The Bessel function approximation is nearly optimal and much easier to compute than the prolate spheroidal wave function.

Henceforth, following [14], the $H(\alpha, \beta)$ windowing function in the pseudo-Wigner distribution is chosen to be a Kaiser window and is defined as :

$$H(k, l) = \frac{I_0\left[\gamma\left[1 - \left(\frac{k}{N}\right)^2\right]^{\frac{1}{2}}\right] I_0\left[\gamma\left[1 - \left(\frac{l}{N}\right)^2\right]^{\frac{1}{2}}\right]}{I_0(\gamma) I_0(\gamma)} \quad (5)$$

where

$$I_o(x) = \sum_{i=0}^{\infty} \left[\frac{(x/2)^i}{i!} \right]^2 \quad (6)$$

and $-N \leq k, l \leq N$

with $(2N + 1) \times (2N + 1)$ being the size of the kernel which is zero outside this region. γ is the parameter that governs the trade off between the main lobe width and the side lobe ripple amplitude of the spectrum. Typical values of γ are in the range $4 \leq \gamma \leq 9$.

The other windowing function $g(r, s)$ appearing in equation (4) is for allowing local averaging. Any averaging will smooth the signal and may make the crack we wish to detect blurred and undetectable. Thus, in this paper this function was not used at all. In order to stick to the proper formalism, we may say that we chose a rectangular data window defined as:

$$g(r, s) = \delta_{ri} \delta_{sj} \quad (7)$$

where r and s are integers, δ_{ij} is Kronecker's delta and i, j are also integers that take values such as to identify positions within the smoothing window of size $(2M + 1) \times (2M + 1)$ around pixel (r, s) .

2.1 Texture Crack Detection Algorithm

The crack detector that we propose is able to detect cracks on random or regular textural backgrounds. Basically, it consists of three parts:

- System training for the learning of the underlying texture.
- Analysis of the test image and calculation of the Mahalanobis distance map.
- Post-processing to isolate the crack pixels.

In the first stage, the pseudo-Wigner spectrum at each pixel position of a defect-free image is calculated. Each local Wigner spectrum is normalised to have unit dc spectral component. In other words, the absolute magnitudes of the spectral components are not used as in many other cases (e.g. [15, 16]). This is because it was noticed that the information needed for the detection of cracks was best encapsulated by the general shape of the spectrum and not necessarily by the exact value of each Wigner spectral component. The Wigner distribution is a real function and the phase information is implicitly encapsulated in the negative parts of the spectrum. Therefore, we do not lose any phase information after normalisation. The normalised amplitude of each spectral component is then considered as our local texture feature and only half of those features need to be retained due to symmetry. Generally, defective pixels can be isolated in the feature space by using some sort of optimal distance measure from the distribution of the pixels of the underlying texture. The Mahalanobis distance seems to be appropriate. However, when the covariance matrix of the distribution was computed, it was found to be singular, an indication that the features used were not independent. It became obvious, therefore, that a new set

of features needed to be computed such that the covariance matrix in the feature space was invertible.

Let us denote by f the local feature vector associated with each pixel of the defect-free image and Σ the covariance matrix of their distribution. We can diagonalize Σ by writing :

$$\Sigma = U \Lambda U^T \quad (8)$$

where U is the matrix made up from the eigenvectors of Σ used as columns, U^T is its transpose and Λ is a diagonal matrix of the eigenvalues of Σ arranged in the descending order of their magnitude along its diagonal. Suppose now that we retain only the m largest eigenvalues and we set the rest to zero. The corresponding transformation matrix \tilde{U} then will consist of the corresponding m eigenvectors only.

We can thus define new feature vectors \tilde{f} assigned to each pixel by using the linear transformation matrix \tilde{U}^T :

$$\tilde{f} = \tilde{U}^T f \quad (9)$$

The new feature vector \tilde{f} consists of m components only which are uncorrelated with each other and encapsulate the most important features of the distribution. In the second stage, the local texture features were computed from the test image as described in the training phase. In the new feature space with the reduced dimensionality, we can use the Mahalanobis distance to measure the distance of each pixel of the test image from the cluster of pixels of the training image. The new covariance matrix of the distribution is the truncated matrix $\tilde{\Lambda}$. Thus we can create a residual map of the test image which contains the Mahalanobis distance of each pixel from the distribution of the defect-free image in the feature space. Let d be a Mahalanobis distance function defined. Then for each pixel location $[i, j]$, we have

$$d_{ij} = |(\tilde{f}_{ij} - M_j)^T \tilde{\Lambda}^{-1} (\tilde{f}_{ij} - M_j)| \quad (10)$$

where M_j is the transformed mean feature vector of the underlying texture. Clearly, pixels with large distance measures are potential crack pixels. One could simply threshold the distance map to isolate the defective pixels. This, however does not create a very clean output and in some applications, one may require to identify the crack lines accurately. Henceforth, some post processing is necessary and this is described in the next subsection.

2.2 Optimal Line Filtering

The post-processing method we used on the residual map is the optimal line filtering approach. Given the assumptions that cracks are mostly generated due to sudden exertion of external force or material fatigue, the crack features embedded in the Mahalanobis map should have a dominant orientation, that is, horizontal or vertical, instead of becoming spiral in shape. We can then convolve the Mahalanobis distance map with a line filter in the direction normal to the basic orientation of the linear features. The orientation is estimated by comparing variances of responses computed from the distance map in the horizontal and vertical directions. Obviously, the direction that has

the smallest variance is the basic orientation of the linear features.

The line filter that we used [17] is a one dimensional directional filter which detects lines. All linear features with widths within a factor of 2 of the width of the feature for which the filter is optimal can be detected. The filter parameters are designed optimally by modelling the intensity profile of the linear features in the Mahalanobis distance map and maximising a composite performance criterion [17]. When a local maximum in the output is detected, a hypothesis is generated that there is a linear feature passing through it. Since the filter is developed around the assumption that the linear feature we want to detect is adequately described by a certain model, we know what sort of output is expected from the filter when a true linear feature is encountered. Thus, when the hypothesis of the presence of a linear feature is generated, a template of the expected filter response is invoked and a matching procedure is applied similar to a χ^2 test. If the value of the residual of this template matching is below a certain threshold, a linear feature is marked. The strength of the linear feature marked is calculated as the difference between the response of the filter at the position of the central pixel minus the average response of the filter at two neighbouring positions symmetrical about the centre where the expected response is known to have another local extremum of the opposite sign from the central one. This number is considered to be the contrast of the linear feature. Subsequently hysteresis thresholding is applied to these contrast values.

The filters as described in [17] are one dimensional, ie they are only 1 pixel wide. In some cases the result could be improved if some smoothing was applied in the direction orthogonal to the direction of convolution, before the convolution with the line filter. The line filter effectively smooths the signal (in the direction of convolution) and at the same time estimates its second derivative in the same direction. It was considered, therefore, as most appropriate to use for just smoothing the line filter twice integrated. Such a filter would be expected to be "optimal" for smoothing, in the sense that it would preserve the linear feature to be detected as best as possible. Thus, what we effectively do is to convolve the image with a two dimensional filter $h(y)f(x)$ (with $f(x)$ being the line detection filter and $h(x)$ the function $f(x)$ integrated twice) which is separable and thus very efficient. The smoothing filter $h(x)$ is given by:

$$h(x) = \frac{1}{2\alpha^2}[-K_2e^{\alpha x} \cos(\alpha x) + K_1e^{\alpha x} \sin(\alpha x) + K_4e^{-\alpha x} \cos(\alpha x) - K_3e^{-\alpha x} \sin(\alpha x)] + \frac{1}{2}K_7x^2 + \frac{A_1K_5}{l^2}e^{lx} - \frac{A_2K_6}{s^2}e^{sx} + L_1x + L_3 \quad (11)$$

$$\text{for } -d \geq x > -w$$

and

$$h(x) = \frac{1}{2\alpha^2}[-N_2e^{\alpha x} \cos(\alpha x) + N_1e^{\alpha x} \sin(\alpha x) + N_4e^{-\alpha x} \cos(\alpha x) - N_3e^{-\alpha x} \sin(\alpha x)] +$$

Parameters	$s = 2$	$s = 3$	$s = 4$
L_1	-153.6941	-81.3208	-132.7542
L_2	-180.2321	-43.82824	4.411499
L_3	1337.822	716.452	1092.654
L_4	-2003.133	-814.4626	-1057.186
Parameters	$s = 5$	$s = 6$	$s = 7$
L_1	-106.0458	-135.4477	-119.2798
L_2	29.46426	86.54103	86.88538
L_3	900.1965	941.3357	997.8079
L_4	-792.1042	-779.5654	-778.3885

Table 1: Values for features of varies sharpness s

$$\frac{1}{2}K_7x^2 + \frac{A_3K_6}{s^2} \cosh(sx) - \frac{A_4K_5}{l^2} \cosh(lx) + \frac{1}{2}N_6x^2 + L_2x + L_4 \quad (12)$$

for $0 \geq x > -d$

The values of the constants L_1-L_4 are chosen so that the two branches of the solution match smoothly at $x = -d$ and the filter vanishes smoothly at $x = -w$ where w is its half size. All other parameters that appear in the above expression are as defined in [17] and it is beyond the scope of the present work to go into more detail about them. The only parameters that are not discussed in [17] are L_1-L_4 and we give their values here in Table 1 for features of varied sharpness (expressed by parameter s) and calculated for feature half-width $d = 1$ and $l = 10$. This filter should be scaled and used the same way as the line detection filter described in [17]

3 CHROMATO-STRUCTURAL APPROACH TO DEFECT DETECTION

In this section a hybrid chromato-structural approach to colour texture representation is proposed where structural colour texture features are extracted from various chromatic classes associated with the colour image texture. It combines colour clustering with a binary blob image analysis to capture the relevant information content of the textures.

The colour texture defect detection algorithm consists essentially of two stages: the first stage (training stage) is where the system is trained on textured images or image regions which are void of defects. The second stage is where the system is analysing the given image for the presence of any defects, as well as detecting their locations in the image.

As we are interested in chromatic macrotexture, rather than microtexture, in the second stage of our approach, we aim to extract structural texture information from various chromatic categories by measuring the structural statistics on blobs of similar colour so that fine local chromatic variations are ignored. This can be achieved by classifying the image pixels into chromatic categories defined during the training stage. For each chromatic category, we identify all the pixels that can be confidently associated with it

by setting a single bit binary flag to unity. Thus for each class, we obtain a binary image of pixels that have been assigned to it. An additional binary image is generated for the reject class which contains all the pixels that have not been accepted by any of the chromatic categories. Thus, we transform the colour macrotexture image into a stack of binary blob images. As these will invariably be noisy, they will be subjected to morphological smoothing before any structural analysis (blob size, shape and distribution) can be carried out.

In our approach, we have developed a new colour clustering method, described later, to define chromatic categories. Each chromatic category \mathcal{W}_k is associated with one or more sub classes. Each sub class is defined by its inferred mean μ_i and the probability $P(\omega_i)$ which allows us to invoke the Bayes minimum error rule for pixel classification in the colour inspection stage. The binary images of blobs of different colour are then processed in order to calculate their area, size, elongatedness and spatial distribution as reflected by the inter-blob distance. It is assumed that these attributes are distributed normally with the inferred mean μ_i and covariance matrix Σ_i .

3.1 Colour Clustering Scheme

Segmenting or clustering a colour image into different classes in the absence of *a priori* information is still a fundamental issue in image processing. The main difficulty is that the model and its parameters are unknown and need to be computed from the given image before segmentation. Moreover, the clustering results are used for subsequent processing in our colour texture defect detection algorithm. Therefore, the entities of interest in the colour image texture not only need to be well extracted, but also have to be well represented by the individual classes. These stringent requirements make the clustering process very difficult.

Several techniques have been proposed in order to tackle these problems [18, 19, 20]. Here we adopt a clustering approach. The accuracy of pixel data representation in terms of clusters depends a great deal on the number of clusters generated as shown in figure (1) for a typical granite texture image. Motivated by this consideration, a novel colour clustering algorithm is developed which essentially consists of two stages: initial clustering and perceptual merging. Generally speaking, we segregate the colour image texture into fine or small clusters in the RGB space, and merge them according to some meaningful property, i.e. their perceptual colour similarity. With appropriate merging strategy and termination criterion, a super cluster or class will be represented by a group of sub-clusters. In other words, the data are represented by a group of sub-clusters associated with the same class label. Less smoothing and better class representation of the true data distribution will be the result. Prior knowledge regarding the actual number of clusters that must be formed is no more necessary in this case.

we perform the initial clustering in the RGB space. Since RGB are the principle colours digitized during image acquisition, noise should be uniformly distributed. The resulting clusters formed in RGB space should be more accurate and less sensitive to noise. We assume that the number of ini-

tial clusters that we choose is sufficiently large to give good data representation and capture fine chromatic variation. Once the initial clusters are formed in the RGB space, we have to merge them in some meaningful way. Inter-cluster distance in the RGB space does not convey any physical information on colour difference. More explicitly, the distance between any two points in the RGB space does not give the measure of colour difference between two colour perceptions. Since merging of clusters based on the colour difference is the best approach, we transform all the data of the clusters into a uniform CIE colour space and retain the class labels obtained in the RGB space. Once the transformation is carried out, we merge clusters by means of measuring the colour difference between two cluster means. One should note that by forming the clusters first in the RGB space and merging them according to their cluster means in the uniform colour space, we are in fact eliminating the non-linear noise effect on clusters. Details of initial clustering and perceptual merging are given in the next two subsections.

3.2 Initial Clustering

The technique used for initial clustering is the histogram based K-means algorithm, with a very large number of clusters to avoid grouping together pixels which in the perceptually uniform space would be distinct. In the case of the three dimensional space of colour images, this histogram based approach is advantageous only if it is combined with a coding scheme that does not require any memory allocation for empty histogram bins. We have adopted the technique advocated in [21]. The basic idea is to project the 3-D histogram into 1-D using for each pixel a unique number computed from its RGB values as follows:

$$\text{Unique number}[i] = R[i] + G[i] * L + B[i] * L^2 \quad (13)$$

where i identifies the pixel and L is the maximum gray level value of the colour image which is normally 255. To minimise the number of memory storage accesses, a B-tree data structure [22] is implemented to store the 1-D histogram. The idea of the B-tree data structure is to construct a balanced multi-way tree. With each access during the multi-point reassignment, a block that has several records will be accessed and from this block, a multi-way decision is made about which block to access next. Notice that this 1D projection is only a storage saving trick and no information is lost.

As a poor initial partition could result in unrepresentative clustering we adopt an efficient initialisation procedure for highly correlated data. The idea behind the advocated initialisation scheme is based on the recognition that the data distribution can be characterised by its principal directions and the eigenvalues which are the variances of the distribution along the corresponding axes. Principal component analysis of several granite images showed that their distribution has the shape of an elongated pancake with its longest axis in the direction of the RGB line, i.e. the line along which we measure grayness. In our application, 25 seeds are chosen, a number that is found to be sufficient to capture all the fine chromatic variations. Figure 2 shows

schematically such a distribution and the location of the 25 seed points used to partition the distribution into 25 clusters. Once the cluster centres are determined an iterative reassignment of pixels is carried out until a stable data partition is reached.

3.3 Perceptual Merging

The initial clustering stage produces a large number of small clusters many of which could be perceptually identical. To identify the perceptually similar clusters, we transform the data into the CIE colour space. This step is necessary because, as mentioned earlier, the Euclidean distance in RGB space does not reflect perceptual similarity. The transformed data keep their respective class labels obtained in the initial clustering. By measuring the colour difference between all pairs of clusters represented by their respective cluster means, we merge clusters with colour difference which is not greater than some predetermined threshold. When any two clusters are merged, a super cluster label is assigned to the merged clusters. The new cluster mean $\vec{\mu}_k$ for the super cluster is calculated by:

$$\vec{\mu}_k = \frac{N_i \vec{\mu}_i + N_j \vec{\mu}_j}{N_i + N_j} \quad (14)$$

where $\vec{\mu}_i, \vec{\mu}_j$ and N_i, N_j are the cluster means and the number of pixels in cluster i^{th} and j^{th} respectively. This updating tends to create more compact cluster associations which is consistent with the fact that Euclidean distance in the perceptual colour space reflects perceptual colour discrimination. The merging process is iterative and is repeated several times until no more clusters exist which are closer than the predetermined threshold. At the end of the merging process, every sub-cluster will be associated with a super cluster label.

3.4 Morphological Smoothing

One should note that both during the training and the identification stage of our algorithm, a colour image will be decomposed into a stack of binary images by means of chromatic clustering or classification. This process is based solely upon the feature space distribution and does not take into consideration the spatial distribution of pixels. Therefore, the resulting binary images may have small holes and individual loose pixels which may create the impression of a filamentary texture, contrary to the blob-like texture humans perceive when they see these images. To eliminate them, each binary image is subjected to morphological smoothing. Two secondary morphological operations are used, that is opening and closing. *Opening* generally smoothes the contour of a region, breaks narrow isthmuses and eliminates thin protrusions. *Closing* also tends to smooth sections of a region but, as opposed to opening, it generally fuses narrow breaks and long thin gulfs, eliminates small holes, and fills gaps in the region. Hence, the closing of the opening operation $(A \circ B) \bullet B$ is performed on each chromatic category or binary image [23].

$(A \circ B)$ is the opening operation defined as :

$$(A \circ B) \oplus B = dilate[erode(A, B), B] \quad (15)$$

$(A \bullet B)$ is the closing operation defined as :

$$(A \oplus B) \ominus B = erode[dilate(A, -B), -B] \quad (16)$$

where A is the binary blob image, B is a simple filled 3x3 mask, $-B$ is the same as B in our case and \ominus, \oplus are the Minkowski addition and subtraction respectively. Once the morphological operation is completed, the binary images of blobs will be free of noise and the blobs will be better defined.

3.5 Structural Texture Analysis

Each blob in a binary image is characterised by four numbers:

- Area f_a .
- Perimeter fractality f_c .
- Elongatedness f_e .
- Spatial information f_s .

The area is the number of pixels associated with the blob. The perimeter fractality of a blob is defined by :

$$f_c = 2\pi A/P \quad (17)$$

where A and P are the area and perimeter of the blob respectively. When the area of two blobs is the same, the one with the more ragged perimeter will have a smaller f_c value. Also, if two blobs have the same perimeter length, the one with the less smooth perimeter will have the smaller value of f_c .

Elongatedness is defined in terms of the moments of the blob. Let us denote with M_{11}, M_{20} and M_{02} the second order moments of the blob. Then, in a coordinate system rotated by an angle θ

$$\theta \equiv \frac{1}{2} \tan^{-1} \frac{2M_{11}}{M_{20} - M_{02}} \quad (18)$$

with respect to the image coordinate system and centered at the centroid of the blob, the mixed second order moment M'_{11} vanishes. The remaining two second order moments M'_{20} and M'_{02} can then be used to characterise the elongatedness of the blob.

$$f_e = \left| \frac{M'_{02} - M'_{20}}{M'_{02} + M'_{20}} \right| \quad (19)$$

When the blob is of circular shape, the elongatedness feature is zero.

All the above features described so far do not convey any spatial information regarding how densely distributed each type of colour blobs is. The motivation for calculating the spatial features arises from the fact that if the same type of colour blobs are clustered together, human vision will also perceive this region as a defect. On the contrary, if the colour blobs are uniformly scattered, only colour blobs with abnormal shape or size will be considered as defects. Hence, we must include some spatial information in the blob features in order to quantify how closely the blobs are distributed within a certain specific region. We first

compute the centroid of each blob and then for each blob we compute the spatial feature f_s given by:

$$f_s = \frac{1}{W^2} \sum_{i=1}^m \xi_k \quad (20)$$

where m and ξ are the number and area of the blobs that fall within the local region of size $W \times W$ around the blob under consideration. To avoid having a biased feature, the area of the central blob is not included in the calculation of the spatial feature. However, some types of colour blobs may come in many sizes as shown in Figure 3. Therefore, it is reasonable to estimate the size of the local region in units of the radius of the central blob, given by:

$$W = K \sqrt{\frac{A}{\pi}} = K r \quad (21)$$

where A and r are the area and the effective radius of the blob respectively and K is some weighting factor.

In our defect detection algorithm, structure statistics defined in the training phase are extracted from these attributes on the assumption that these attributes are distributed normally with the inferred mean μ_{f_k} and covariance matrix Σ_{f_k} .

3.6 Defect Identification

In the final stage, we divide the defect identification into two main stages, ie. colour inspection and blob defect identification. We assumed that the class conditional probability density for each chromatic category \mathcal{W}_k ($\mathcal{W}_k = \{\omega_i; i = 1, \dots, n\}$) is constituted by a multi-modal function. Each subcluster of a chromatic category has a mean $\bar{\mu}_i$ and the a prior probability $P(\omega_i)$ for a pixel to belong to it is equal to N_i/N . These multi-modal models are designed during the training phase and are represented by the chromatic statistics obtained in the colour blobs segregation stage. The colour inspection based on this design can then be carried out using the Bayes minimum error rule discriminant function given by:

$$J_i(\vec{f}) = \log \frac{N_i}{N} - \frac{1}{2}(\vec{f} - \bar{\mu}_i)^T(\vec{f} - \bar{\mu}_i) \quad \forall \vec{f} \in \omega_i \quad (22)$$

where \vec{f} is the RGB feature vector of each pixel in the test image. The Bayes decision rule classifies a pixel to the chromatic category \mathcal{W}_k if and only if

$$J_j(\vec{f}) > J_i(\vec{f}) \quad \text{where } \omega_j \in \mathcal{W}_k, \forall (i \neq j) \quad (23)$$

and assigns a binary flag to each pixel in the corresponding chromatic category. Therefore, a stack of binary blob images is formed. Pixels with values of J_i for all classes \mathcal{W}_k below a certain threshold are rejected and considered as colour defects. These pixels form the *colour defects* output of the algorithm and they are not considered further in the process.

Those pixels that have been classified into the predetermined chromatic categories are then sent to the second stage of classification. Each binary image in the stack is smoothed by the morphological operators as it was done

during the training phase. The resulting binary blob images are then subjected to structural texture analysis. The structural features of each colour blob in each binary blob image are computed. Subsequently, blob defects in each chromatic category or each binary blob image are identified by means of the Mahalanobis distance discriminant function.

$$G_k(\vec{g}) = (\vec{g} - \bar{\mu}_{f_k})^T \Sigma_{f_k}^{-1} (\vec{g} - \bar{\mu}_{f_k}) \quad \forall \vec{g} \in \mathcal{W}_k \quad (24)$$

where $\vec{g} = [f_a, f_c, f_e, f_s]^T$ is the feature vector of a colour blob and $\bar{\mu}_{f_k}$ and Σ_{f_k} are the inferred structure statistics obtained in the training phase for the specific chromatic category \mathcal{W}_k . Note that at this stage blob classification is carried out instead of pixel classification and thus blobs with abnormal structural features are identified as defects.

4 EXPERIMENTAL RESULTS

The tiles used in our experiments were ceramic, granite, and marble of a size of at least $200 \times 200 \text{ mm}$. In the images shown in this paper, some defects may not be easily visible. In most crack defect images a dilation operation is carried out to enhance the results.

Figures 4 and 5 show pin-hole and small crack defects on lightly textured tiles. We used a modified version of the line filter to detect these spot-like faults without any other pre-processing.

In Figures 6, 7, 8, and 9, we present four images with surface cracks; the first of these was superimposed and consists of a single pixel wide crack running almost across the entire length of the tile. All of these cracks were detected by using the modified pseudo Wigner distribution and optimal line filter post-processing (displaying the capability of the filter in detecting lines of various widths). We experimented with various window sizes for the windowing function of the Wigner distribution and found a 7×7 size provides the best discrimination of defects. The line filter was tuned on a corresponding defective tile before application during the test stage.

Figure 10 shows texture abnormalities detected using the chromato-structural defect detection algorithm. As an example, the image in 10(a) is split into 11 distinct colour categories and following morphology and Mahalanobis distance comparison of blob characteristics, the fault in 10(b) is reliably detected. Thus, our algorithm seems to be very robust and picks up the abnormalities accurately.

In order to show the power of the chromatic discrimination in our algorithm we next show an untypical result in Figure 10(a) which contains a spot-like colour abnormality besides the obvious large blob defect. The encircled spot defect in Figure 10(b) was detected at the time of colour category classification when it was rejected as a colour not identified during the training process. Figure 10(c) shows a 5×5 zoomed region of the blue-band of the image around this defect to demonstrate the sensitivity of the algorithm. Finally, Figure 11 shows four tiles with varying sizes of structural and chromatic defects whose borders are outlined.

5 CONCLUSIONS

Automatic visual inspection of colour textures plays a crucial role in machine vision applications. In this paper we have shown two approaches for the detection of surface defects in colour textured images. These were the pseudo-Wigner distribution and the chromato-structural defect detection approach.

Initially we described the pseudo-Wigner distribution which provides a conjoint spatial and spatial frequency representation of the texture surface in this application. For crack detection, it was found that the local crack information was best encapsulated in the general shape of the spectrum. Also, we discarded the local averaging window of the classic pseudo-Wigner distribution as it was found to smooth and blur the crack signals we wished to detect. Next, we described the crack detection algorithm consisting of an initial training stage and a testing stage. During the training stage the statistical distribution of the Wigner spectra of the underlying texture was computed using the pseudo-Wigner formulae. In the testing stage the residual map of the Mahalanobis distance of the local spectrum from that distribution was calculated followed by post-processing with the optimal line filter. Furthermore, the optimal line filter was described in some detail. This was used independently and as a post-processing stage to our other techniques.

Furthermore, we have introduced a unique framework to tackle the problem of defect detection in random textured images based on their colour and texture information. In order to segregate the colour image texture to various chromatic classes, a new colour clustering scheme which uses histogram based clustering and perceptual merging based on human colour perception was developed. Initially, we segregate homogeneous colour regions from a given texture image by means of K-means clustering with multi-point reassignment strategy, an efficient initialisation procedure and coding scheme designed to improve performance and reduce computational complexity respectively. This is followed by perceptual merging which allows more accurate statistical models to be inferred from the clusters and does not require *a priori* information regarding the number of colours. This is a very important feature in our application since it is very difficult to guess the number of colours associated with each image. In fact, if the colour space that we used for perceptual merging is perfectly uniform, then we are certain that our proposed algorithm will perform excellently for all colour images. Unfortunately, in the existing colourimetry field, both International colour standards recommended by the CIE only closely approximate uniform colour spaces. For some colours, the distance between them are still not according to the human colour perception. For example, gray colours. Finally, the resulting multiple images are then subjected to morphological filtering and globally represented by means of texture descriptors.

We also showed the results of the application of both approaches and can conclude that both perform extremely well. The pseudo-Wigner distribution for crack detection is computationally expensive and will benefit considerably if

it were to be tailored for a more powerful platform such as a network of parallel processors. Work on this is currently at hand.

More specific to the problem of ceramic, granite and marble inspection, it is hoped that the considerable advance achieved in overall production through the automation of tile inspection will eliminate an estimated 70-80% customer complaint rate regarding product quality[24]. Furthermore, the spin-offs of the findings of this project can have an impact in other industrial fields presenting similar problems; for instance in the textile industry for defect detection, loose threads detection, and colour shading classification on fabrics, the agro-food industry for visual analysis of crops such as apples/oranges/pears/etc, the wood industry for texture and colour classification, and in a number of other industries.

Acknowledgements

This work has been supported by the BRITE projects 0946 AVIS and 0260 ASSIST.

References

- [1] R. M. Haralick. Statistical and structural approaches to texture. *IEEE Proceeding*, 67(5):786-804, 1979.
- [2] S. C. Tan and J. Kittler. On colour texture representation and classification. *Proc. of 2nd Int. Conf. on Image Processing*, pages 390-395, 1992.
- [3] S. C. Tan and J. Kittler. Colour texture classification using features from colour histogram. *Proc. of 8th SCIA.*, 1993.
- [4] T. Caelli and D. Reye. On the classification of image regions by colour texture and shape. *Pattern Recognition*, 26(4):461-470, 1993.
- [5] G. J. Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two dimensional visual cortical filters. *J. Opt. Soc. Am.*, 2(7):1160-1169, 1985.
- [6] D. Gabor. Thoery of communication. *J. Inst. Elec. Eng.*, 93:429-459, 1946.
- [7] M. Unser. Local transforms for texture analysis. *7th ICPR*, pages 1206-1208, 1984.
- [8] M. J. Bastiaans. The wigner distribution function applied to optical signals and system. *Optics communications*, 25(1), 1978.
- [9] T. A. C. M. Claasen and W. F. G. Mecklenbrauker. The wigner distribution-a tool for time-frequency signal analysis; part 2: Discrete time signals. *Philips J. Res.*, 35(3):276-300, 1980.
- [10] L. Jacobson and H. Wechsler. The wigner distribution and its usefulness for 2d image processing. *Proc. 6th Int. Joint Conf. Pattern Recognition*, pages 19-22, Oct 1982.

- [11] W. Martin and P. Flandrin. Analysis of non-stationary process: short time periodograms versus a pseudo wigner estimator. *Schussler EURASI*, pages 455–459, 1984.
- [12] D. Slepian. Prolate spheroidal wavefunctions, Fourier analysis and uncertainty V: te discrete case. *B. S. T. J.*, 43:3009–3057, 1964.
- [13] J. F. Kaiser. *Digital filters*, volume Chap 7. Wiley, New York, F.F. Kuo and J.F. Kaiser edition, 1966.
- [14] L. D. Jacobson and H. Wechsler. Joint spatial/spatial-frequency representation. *Signal processing*, 14:37–68, 1988.
- [15] J. S. Weszka, C. R. Dyer, and A. Rosenfeld. A comparative study of texture measures for terrain classification. *IEEE Transactions on System, Man and Cybernetics*, 6(4):269–285, April 1976.
- [16] A. D’astous and M. E. Jernigan. Texture discrimination based on detailed measures of the power spectrum. *Proc of 7th ICPR*, 6(4):269–285, 1984.
- [17] M. Petrou. Optimal convolution filters and an algorithm for the detection of linear feature. *IEE Proc. I Communications, Speech and Vision*, 1993.
- [18] R. Ohlander, K. Price, and D. R. Reddy. Picture segmentation using a recursive region splitting method. *Computer Vision, Graphics and Image Processing*, 8:313–333, 1978.
- [19] J. M. Tenenbaum, T. D. Garvey, S. Weyl, and H. C. Wolf. An interactive facility for scene analysis research. *Technical Note 87*, 1974.
- [20] M. Celenk. A color clustering technique for image segmentation. *Computer Vision, Graphics and Image Processing*, 52:145–170, 1990.
- [21] J. Kittler and B. H. Ang. An iterative multispectral image segmentation. *Proc. of Int. Conf. on Image Analysis and Processing*, pages 78–85, 1990.
- [22] Robert L. Kruse. *Data structures and program design*. Prentice-Hall, India, 2nd edition, 1987.
- [23] R. C. Gonzalez and R. E. Woods. *Digital Signal Processing*. Addison-Wesley, 1992.
- [24] BRITE-EURAM II. Automatic system for surface inspection and sorting of tiles. Technical report, ANNEX I, 1992.

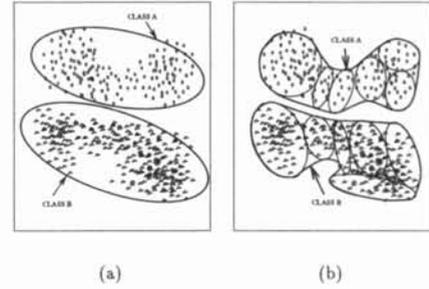


Figure 1: (a) Data are poorly represented by clusters A and B. (b) Super clusters A and B are formed by many sub-clusters. This gives a better data representation than (a).

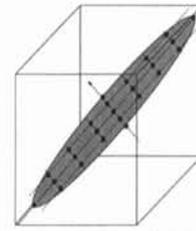


Figure 2: Schematic distribution of pixels in RGB space and initialization points for the clustering algorithm.

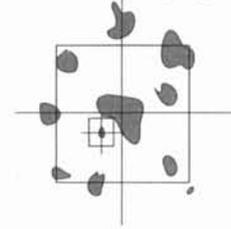
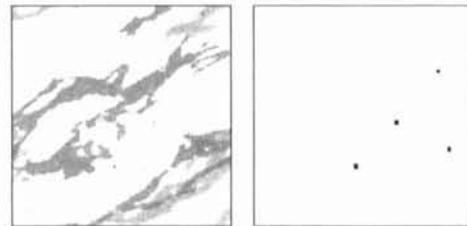
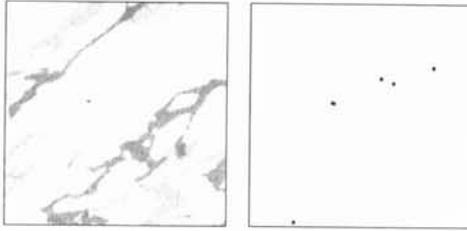


Figure 3: This figure illustrates that the local region used for calculating the spatial feature is dependent on the blob size.



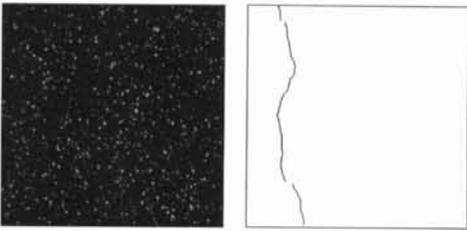
(a) Carrara textured tile (b) Hole defects

Figure 4:



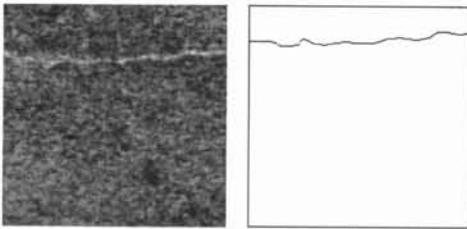
(a) Carrara textured tile (b) Crack and spot defects

Figure 5:



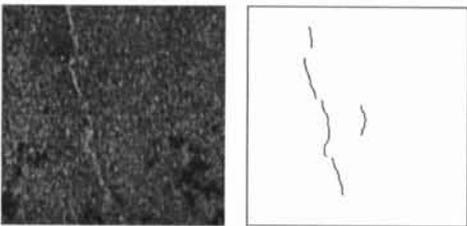
(a) Donizetti ceramic textured tile with superimposed crack (b) Crack defect

Figure 6:



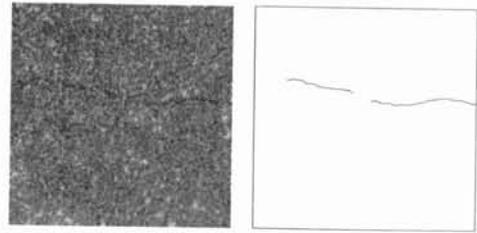
(a) Real crack on granite tile (b) Crack defect

Figure 7:



(a) Real crack on granite tile (b) Crack defect

Figure 8:

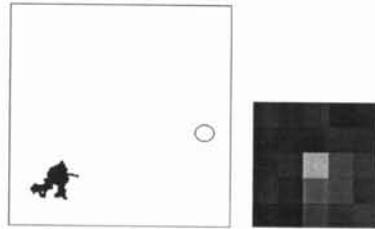


(a) Real crack on marble tile (b) Blob defects

Figure 9:



(a) Donizetti textured tile



(b) Blob and encircled colour defect (c) Defect pixel and neighbourhood in the blue band

Figure 10:

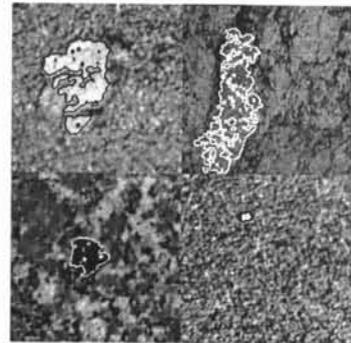


Figure 11: Four (granite and marble) tiles and their chromo-structural defects