SURFACE DEFECTS DETECTION FOR CERAMIC TILES USING IMAGE PROCESSING AND MORPHOLOGICAL TECHNIQUES

Rashmi Mishra*

C. L. Chandrakar**

Ravi Mishra**

ABSTRACT

Quality control in ceramic tile manufacturing is hard, labour intensive and it is performed in a harsh industrial environment with noise, extreme temperature and humidity. It can be divided into colour analysis, dimension verification, and surface defect detection, which is the main purpose of our work. Defects detection is still based on the judgment of human operators while most of the other manufacturing activities are automated so, our work is a quality control enhancement by integrating a visual control stage using image processing and morphological operation techniques before the packing operation to improve the homogeneity of batches received by final users. An automated defect detection and classification technique that can ensured the better quality of tiles in manufacturing process as well as production rate.

Keywords: *Quality control, Defects detection, Visual control, Image processing, Morphological operation.*

*SSCET, Bhilai.

**Senior Assistant Professor, SSCET, Bhilai.

INTRODUCTION

THE ceramic tiles manufacturing process has now been completely automated with the exception of the final stage of production concerned with visual inspection. This automated classification method helps us to acquire knowledge about the pattern of defect within a very short period of time and also to decide about the recovery process so that the defected tiles may not be mixed with the fresh tiles. This paper is concerned with the problem of automatic inspection of ceramic tiles using computer vision. It must be noted that detection of defect in textured surfaces is an important area of automatic industrial inspection that has been largely overlooked by the recent wave of research in machine vision applications. Humans are able to find such defects without prior knowledge of the defect-free pattern. Defects are viewed as in-homogeneities in regularity and orientation fields. Two distinct but conceptually related approaches are presented. The first one defines structural defects as regions of abruptly falling regularity, the second one as perturbations in the dominant orientation. Both methods are general in the senses that each of them is applicable to a variety of patterns and defects. Human judgment is, as usual, influenced by expectations and prior knowledge. However, this problem is not specific to structural defects. In many detection tasks for example, edge detection, there is a gradual transition from presence to absence. On the other hand, in "obvious" cases most naïve observers agree that the defect is there, even when they cannot identify the structure. Such a monitoring task is of course tedious, subjective and expensive but it is based on a long experience and can utilize the huge appreciation and recognition abilities of the human brain.

Any machine vision system will never advantageously replace the visual inspection if it is not able to:

1. Analyze the colour of the product with reliability.
2. Detect every type of manufacturing defects, with at least the same accuracy as the human eye.
3. Measure with high precision the dimensions of the tiles

The defect detection operation induces that the entire surface of every tile must be imaged and analyzed. The goal of the inspection is to give a statistical analysis of the production batches. So, all batches of tiles will be imaged individually without any sampling operation. The image acquisition achieved directly on line, in the real time. The image analysis algorithm must be fast enough to follow the production rate .This paper aims to create a visual system that is capable of detecting the surface defects for the fired ceramic tiles. That ensure the products are free from defects for the classifying process. Classifying process must be effectively, objectively and repeatedly, with sufficient rapidness and low costs. It must have the ability to adapt autonomously to changes in materials. The techniques used range from Long crack, Crack, Blob, Pin-hole and Spot detectors algorithms for plain, and textures tiles. This therefore reduces the number of complaints tiles. The presented inspection procedures have been implemented and tested on a number of tiles using synthetic and real defects.

The results suggested that the performance is adequate to provide a basis for a viable commercial visual inspection system which we will see it in the next sections.

We generally have found total eight types of defects from the existing defect detection methods. These types of defects are shown in the following Table I.

Name of Defects	Description
Crack	Break down of tile
Pinhole	Scattered isolated black-white pinpoint spot
Blob	Water drop spot on tile surface
Spot	Discontinuity of color on surface
Corner	Break down of tile corner
Edge	Break down of edge
Scratch	Generally scratch on surface
Glaze	Blurred surface on tile

Table I. Types of Ceramic Tile Defects

METHODOLOGY

This project aims to create a visual system that is capable of detecting the surface defects for the fired ceramic tiles. That ensure the products are free from defects for the classifying process. Classifying process must be effectively, objectively and repeatedly, with sufficient rapidness and low costs. It must have the ability to adapt autonomously to changes in materials. The flowchart of my method is shown below:-

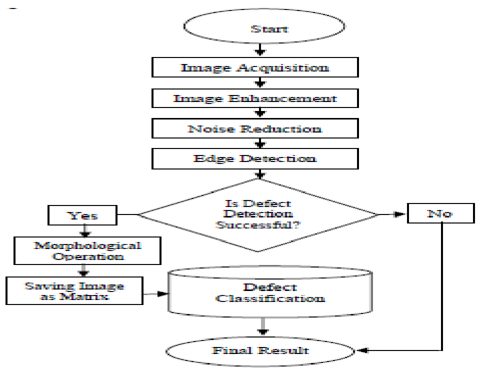


Fig 1:- General flowchart

IMAGE ACQISITION AND CAPTURING

IMAGE ACQUISITION:

Image acquisition is the process of obtaining a digitized image from a real world source. Each step in the acquisition process may introduce random changes into the values of pixels in the image which is referred to as noise.

A ceramic tile image is captured and stored into the computer for further processing. This may be achieved by taking a photograph with a conventional camera, having the film made into a print and scanning the print into a computer.

We intended to create images that are more suitable the human visual perception object detection and target recognition. We used the principles of Image processing and morphological operations on the ceramic tiles images.

Therefore, we get new images that contain the surface defect only to make easier for the detecting process and classification operation via the judgment of the operator.

IMAGE CAPTURING:

The ceramic tiles have been captured through the online camera held on the line production at the industry. The image captured will convert to another kinds of images (Binary, and Gray scale) to be suitable for the various defect detection algorithms used for the different types of defects.

A bag of tricks is used rather than standard algorithms and formal mathematical properties will be discussed like Edge detection, Morphology operations, Noise reduction, smoothing process, Histogram equalization, and intensity adjustment.

The effects of unequal lighting and of the space sensitivity of the TV camera CCD are corrected analyzing a sample tile made of white Plexiglas whose image has been previously divided in 8x8 sectors. This number represents a compromise between spatial resolution distribution and computing time

Image Enhancement

Actually, image enhancement technique is to make the image clearer so that various operations can be performed easily on the image. For this, at first the captured image is converted to the gray level image.

Contrast stretching (often called normalization) is a simple image enhancement technique that attempts to improve the contrast in an image by stretching' the range of intensity values it contains to span a desired range of values, e.g. the full range of pixel values that the image type concerned allows. Low contrast images can be found due to the poor illumination, lack of dynamic range in the imaging sensor, or due to the wrong setting of the lens.

The idea behind the contrast stretching is to increase the dynamic range of intensity level in the processed image. The general process of the contrast stretching operation on gray scale image is to apply the following equation on each of the pixels in the input image to form the corresponding output image pixel:

$$O(x,y) = (I(x,y) - \min)\left(\frac{n_i}{\max - \min}\right) + i$$

where, O(x,y) represents the output image, I(x,y) represents the xth pixel in the yth column in the input image. In this equation, ni represents the number of intensity levels, I represents the initial intensity level, "min" and "max" represent the minimum intensity value and the maximum intensity value in the current image respectively. Here "no. of intensity levels"shows the total number of intensity values that can be assigned to a pixel.

EDGE DETECTION

An edge may be regarded as a boundary between two dissimilar regions in an image. These may be different surfaces of the object, or perhaps a boundary between light and shadow falling on a single surface. In principle, an edge is easy to find since differences in pixel values between regions are relatively easy to calculate by considering gradients.

Many edge extraction techniques can be broken up into two distinct phases:

• Finding pixels in the image where edges are likely to occur by looking for discontinuities in gradients.

• Linking these edge points in some way to produce descriptions of edges in terms of lines curves etc.

Thresholding produces a segmentation that yields all the pixels that, in principle, belong to the object or objects of interest in an image. An alternative to this is to find those pixels that belong to the borders of the objects.

Applying the Defect Detection Process: All pre-processing operations are applied to the reference image, stored in the computer database to compare with the test image.

Let, the resulting image is I2. Now we consider I1 as the resulting image found from the test image after applying all pre-processing operations. We propose here a new technique. We store I1 and I2 as matrix form to a file. Let, these two matrices are named as m1 and m2. Then we count the total number of black pixels (in binary, it is represented as 1) in m1and that in m2. These two are then compared. If the number of black pixels in m1 is greater than the number of black pixels in m2 then we can make decision that defect is found in the test image, otherwise we can say that no defect is present to the test image.

To understand this concept clearly, consider the following example:

Let, we have a test image and a reference image of equal size (5×5) . Now applying preprocessing steps on the test image we find matrix m1 whose value is:

Again, applying the preprocessing operations on the reference image another matrix m2 is found:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Here, the number of black pixels for the reference image is 2 and for the test image this number is 6. So, here obviously 6>2 and we can make decision that defect is found on the test image. The detailed block diagram of the proposed defect detection step is shown in the following Fig.

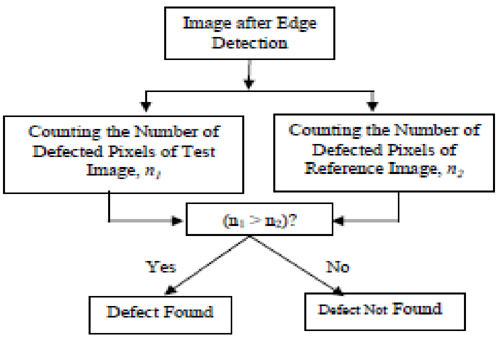


Fig 2:- block diagram of the proposed defect detection step

Noise Reduction

Noise reduction is a process of removing noise from a captured image. To remove noise some filtering techniques can be proposed as follows:

One method to remove noise is by convolving the original image with a mask that represent a low-pass filter or smoothing operation. For example, the Gaussian mask comprises elements determined by a Gaussian function. This convolution brings the value of each pixel into closer harmony with the values of its neighbours. In general, a smoothing filter sets each pixel to the average value, or a weighted average, of itself and its nearby neighbours; the Gaussian filter is just one possible set of weights. But smoothing filters tend to blur an image, because pixel intensity values that are significantly higher or lower than the surrounding neighbourhood would "smear" across the area. Because of this blurring, linear filters are seldom used in practice for noise reduction. For the above reason, we proposed to use a non-linear filter which is called median filter. It is very good at preserving image detail if it is designed properly. To run a median filter:

a) Consider each pixel in the image
b) Sort neighbouring pixels into order based upon their intensities
c) Replace the original value of the pixel with the median value from the list

A median filter is a rank-selection (RS) filter, a particularly harsh member of the family of rank-conditioned rank-selection (RCRS) filters ; a much milder member of that family, for example one that selects the closest of the neighbouring values when a pixel's value is extremely in its neighbourhood, and leaves it unchanged otherwise, is sometimes preferred, especially in photographic applications. Median filter technique is good at removing salt and pepper noise from an image, and also causes relatively little blurring of edges, and hence is often used in computer vision applications.

MORPHOLOGICAL OPERATIONS

Morphological operations are methods for processing binary images also for gray scale images based on shapes.

These operations take the binary and gray scale images as input, and return it as output. The value of each pixel in the output image is based on the corresponding input pixel and its neighbours. By choosing the neighbourhood shape appropriately, you can construct a morphological operation that is sensitive to specific shapes in the input image. As binary images frequently result from segmentation processes on gray level images, the morphological processing of the binary result permits the improvement of the segmentation result. Defects are extracted from the background by thresholding the image and classified according to size and shape parameters. Existing machines commonly detect the following defaults:

1. Chips (edges and corners)
2. Cracks
3. Scratches
4. Glaze faults
5. Holes and pitting
6. Lumps

The sensitivity of the imaging system is linked to the local roughness contrast induced by the defect; it has nothing to do with the colour contrast. Because they rely on two independent physical properties of the material, colour defects and surface defect inspection are complementary. The tiles used in our experiments are of size 200 x 200mm and are either plain or textured. In the testable images, some defects may not be easily visible and we have randomly encircled some of them for saliency. In most defect images, a dilation operation is carried out to enhance the results. All detected faults correspond to true faults.

DETECTION ALGORITHMS FOR FIRED CERAMIC TILES

We will see in this section a number of techniques developed for the detection of multifarious range of ceramic tile defects. Figure 3 shows the first part of the algorithm which takes the captured image for the defective tile then the output is an intensity adjustable histogram equalized image to be the input to the second part of the algorithm. The second part of the main algorithm include many of individual complementary algorithms differs due to the various kinds of defects.

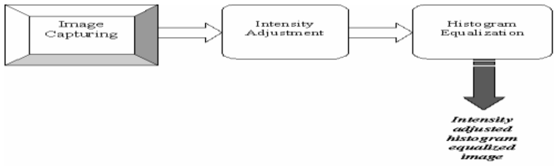


figure 3 first part of the algorithm

Now we will see the different complementary individual algorithms for the detection of variable defects. These algorithms take the intensity adjusted histogram equalized image as the input image. The input image in these algorithms passes through many stages to get the final image which include only the defect. These stages differ from algorithm to another due to the kinds of defects.

Figure 4 shows the Crack and Long Crack defect detection algorithm. The input image converted to a black/white image. An edge detection operation has been done to detect the defect pixels producing approximately areas to the defects so we follow it by a fill gaps operation to discriminate the defect pixels. Some morphology operations have been done to discriminate the defect pixels more accurately followed by Noise reduction and Smoothing object processing to give a clear image containing the defect only.

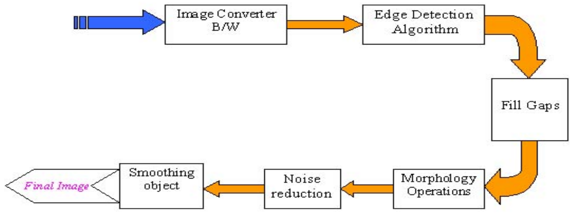


figure 4 crack defect, and long crack defect detection algorithm

Algorithm to Determine Crack Defects

Let c_length as the range of crack.

Step 1. Check every pixel coordinate (i, j) from left to right up to the last pixel element.

Step 2. If any (i, j) has value 1 then

(a) Consider its adjacent eight pixels and find which are 1.

(b) If any adjacent pixel has value 1 then Current pixel coordinate will be updated to it.

(c) Apply the backtracking process to find out all connected pixels and count the length.

Step 3. Apply step 2 to all pixels and for each pixel find out the length of connected pixels.

Step 4. Counting all length of the connected pixels found from step 2 and step 3, find out the maximum number and set it to c_count.

Step 5. Finally, apply step 2 to specify the crack defected pixel coordinates so that other types of defects are not affected to it.

Step 6. If c_count > c_length, then make decision that crack is found, otherwise crack is not found.

The same operations have been applied to the other kinds of defects detection algorithms but without the same arrange and the number of processing cycles to the images.

Figure 5 shows the Spot, Longitudinal Spot, and Depression Spot detection algorithm.

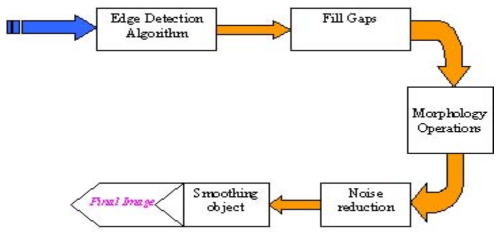


figure 5 spot, longitudinal spot, and depression spot detection algorithm

Algorithm to Determine Spot Defects

Let, matx as size of spot, row as the maximum number of image pixels along any row and col as the maximum number of image pixels along any column.

Step 1. Let, start=(matx/2)+1; Here start is the middle element of (matx*matx) .

Step 2. Check every pixel coordinate (i, j) from left to right up to the last pixel element.

for row consider the range from start to row-start+1

for column consider the range from start to col-start+1

(a) If any pixel coordinate (i,j) is 2, then

(i) Considering it as the middle element and check the total (matx*matx) elements around it to find out how many 2 exists into these region.

(ii) Let, the total number of 2 is equal to s_length.

(iii)If s_length = (matx*matx), then Make decision that spot defect is found and exit From loop.

(b) Otherwise, switch to next pixel coordinate at step 2.

Step 3. After searching every pixel coordinate, if there is nos_length matches to (matx*matx), then make decision that spot defect is not found.

PIN HOLE DEFECT DETECTION

It is easier to detect the Pin-hole defects. That by applying some morphological operations directly to the input image followed by SDC morphological operation (morphology operations specialized for gray scale images). Finally in this algorithm the image passes to Noise reduction processing to get a clear image for the defect.

Figure 6 shows the Pin-hole defect detection algorithm.

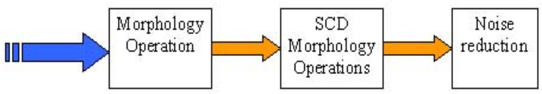


figure 6 pin hole defect detection algorithm

Algorithm to Determine Pinhole Defects

Let, p_count as a variable for pinhole count, c_range as the range of corner, e_range as the range of edge and row as the maximum number of image pixels along any row and column as the maximum number of image pixels along any column.

Step 1. Set, temp_a =c_range , and temp_b=e_range

Step 2. Divide the total searching area for pinhole into three regions.

Step 3. For left side region,

for row consider the range from temp_a+1 to rowc_range-1

for column consider the range from temp_b+1 to c_range

(a) Check every pixel coordinates whether it is 0 or not.

(b) If it is true then

(i) For each coordinate (i,j) check all of its eight neighbours.

(ii) If (i,j-1), (i,j+1), (i-1,j), (i+1,j) position values are 1 and the rest are 0, then P_count will be incremented by 1.

Step 4. For right side region, range for row is from temp_a+1 to row-c_range-1 and range for column is from coltemp_a+1 to col-e_range. The rests are same as step-3.

Step 5. For other middle side elements, range for row is from temp_b+1 to row-e_range and range for column is from col-temp_a+1 to col-c_range. The rests are same as step-3.

Step 6. Finally, check value of p_count. If p_count>0, then pinhole is found, otherwise not found.

Because the blob defects always have no little pixels so it has a discriminated area needs only to display it. The input image complemented by an inverse operation to display more clearly the blob defect pixels followed by some morphological operations and nose reduction processing to get a final image for the blob defect pixels only. A fill gaps operation may be added to the final image to increase the clearance of the defects pixels than others.

Figure 7 shows Blob defect detection algorithm.

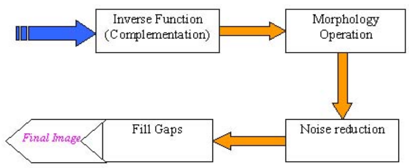


figure 7 blob defect detection algorithm

Algorithm to Determine Blob Defects Let, matx as size of blob, row as the maximum number of image pixels along any row and column as the maximum number of image pixels along any column.

Step 1. Let, start=(matx/2)+1; Here start is the middle element of (matx*matx) .

Step 2. Check every pixel coordinate (i,j) from left to right up to the last pixel element.

for row consider the range from start to row-start+1

for column consider the range from start to col-start+1

(a) If any pixel coordinate (i,j) is 2, then

(i) Considering it as the middle element and check the total (matx*matx) elements around it to find out how many 2 exists into these region.

(ii) Let, the total number of 2 is equal to b_length.

(iii) If b_length = (matx*matx), then Make decision that blob defect is found and exit from loop.

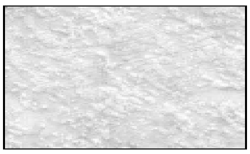
(b) Otherwise, switch to next pixel coordinate at step 2.

Step 3. After searching every pixel coordinate, if there is no b_length matches to (matx*matx), then make decision that blob defect is not found.

EXPECTED OUTPUT

We had described in the previous section optimal Crack, Long-Crack, Pin-hole, Blob, and spot detection algorithms used independently also as post-processing stages to our other techniques. It is a very accurate approach but it is computationally demanding. We apply these algorithms on a several number of tiles, which are plain tiles and textured tiles. In addition, we apply these algorithms on a dozen of tiles images. These tiles images have the same kind of defect whereas the operating conditions (speed of the line with all its irregularities, vibrations etc.) were similar to real conditions. That is for logistic reasons to see if the algorithm give the same result or not and see the differences. When applying the individual algorithms for defect detection we found the results as in the next figures.

Figure 8 and 9 show the defective tile image containing Crack defect and Isolated Crack defect tile image using Image processing and Morphology operations Crack defect detection algorithm.



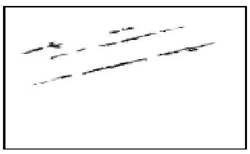


figure 8 & figure 9

Detection Algorithm

Figures 10 and 11 show the defective tile image containing Long crack defect and Isolated Long crack defect tile image using Image processing and Morphology operations Long crack defect detection algorithm.

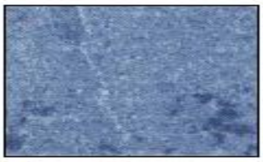
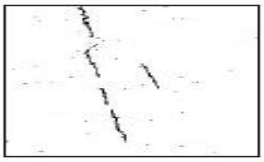
 

figure 10 & figure 11

Figure 12 and 13 show the defective tile image containing Blob defect and Isolated Blob defect tile image using Image processing and Morphology operations Blob defect detection algorithm

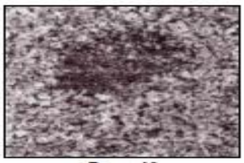
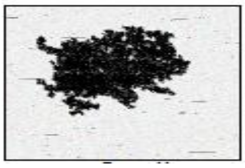
 

figure 12 & figure 13

Detection Algorithm

Figure 14 and 15 show the defective tile image containing Pin-hole defect and Isolated Pin-hole defect tile image using Image processing and Morphology operations Pin-hole defect detection algorithm.

figure 14 

figure 15 

We analyze dozens of tiles including various types of defects. We choose one tile image from each dozen of defective tile images to drag it in this paper. Other typical conditions of ceramic factories (dust, high temperatures, etc) were taken into account during design by applying the first part of Image Processing and Morphology operation detection algorithm which gives an accurate and clear image to be processed in all complementary individual algorithms.

We success in designing individual algorithms that can detect almost all kinds of defects in the fired ceramic tiles. When we tried to apply all individual algorithms as a unique algorithm or an overall algorithm regardless the sequence of these algorithms on the previous defective tile images, we get good results but not as when we had applied individual algorithms. These results are not accurate and clear like the previous results. In addition some defect pixels disappeared therefore it is not described the defect accurately which affect in classification or sorting process that depend on the area of defect.

The result of the project is a prototype analyzer technique with some major simplifications compared to the solutions currently available on the market using an algorithm for detecting the defects in fired ceramic tile. The acquisition system allows the system to be installed without having to make mechanical or electrical modifications to the sorting line. These results in lower costs and means that the system can be moved when required.

CONCLUSION

This paper concerned with the problem of detection of the surface defects included on the fired ceramic tiles using the image processing and Morphology operations. By using this technique we can develop the sorting system in the ceramic tiles industries from depending on the human which detects the defects manually upon his experience and skills which varies from one to one to the automated system depending on the computer vision. That affect mainly in the classification or sorting operation which also done by human in the industry. People can work effectively for short periods and many different operators are involved in checking the same batch of tiles. Continuity over time is not guaranteed and may result in overall poor quality, which may cause customers to complain or even to reject the batch. Miss-sorting is kept at an extremely low level. We success in isolating different kinds of defect in ceramic tiles images. Automated sorting systems would bring numerous benefits to the entire sector with major economic advantages, also guarantee product quality, increase plant efficiency and reduce fixed and periodic investments. The continuous measurement of surface defects gives line production operators to optimize temperature profile, speed and other operating parameters.

REFERENCES

1. Vincent LEBRUN,"Quality control of ceramic tiles by machine vision," Flaw Master 3000, Surface Inspection Ltd. 2001.

2. C. Boukouvalas, J Kittler, R Marik, M Mirmehdi and M Petrou, "Ceramic Tile Inspection for Color and Structural defects." I.E.E.E. Transactions on Pattern Analysis and Machine Intelligence, vol. 14, no. 1, March 1998.
3. Stewart Coe, "Automatic tile inspection." Surface Inspection Limited, International Ceramics, Bristol, U.K., Issue 1, 2000.
4. Costas Boukouvalas, Francesco De Natale, Josef Kittler, and Roberto Salgrai. "An Integrated system for Quality Inspection of tiles." University of Surrey, Guilford, GU2-5XH, England, 1999.
5. Yonghuai Liu and Macros Rodrigues, "A novel machine vision algorithm for a fast response quality control system." Department of Computer Science, the University of Hull, Hull, HU6 7RX, UK 2001.
6. Martin Coulthard, "Measuring and Classifying Tile Shades." Chairman, Surface Inspection Ltd, Great Britain, Jan., 2001.
7. Song K Y, Petrou M, and Kittler, "Texture crack detection." Machine Vision Applications, to appear Jan., 1995.
8. Esko Herrala, "SIMON; Spectral Imaging for On line Sorting." Specim, Spectral Imaging Ltd., Finland, August 1998.
9. Tony Lindeberg, "Edge Detection and Ridge Detection with Automatic Scale Selection." Department of Numerical Analysis and Computing Science, Royal Institute of Technology, International Journal of Computer Vision, S-10044 Stockholm, Sweden, Jan., 1996.
10. Mark E. Lehr and Keh-Shin Li, "Template basis techniques to Pattern recognition." University of California Riverside, Department of Statistics Riverside, California, USA, Vol. 2825, 1996.
11. Todd Reed and H. Wechsler, "Segmentation of textured images and gestalt organization using spatial / spatial-frequency representations." I.E.E.E. Transactions on Pattern Analysis and Machine Intelligence, vol. 12, no. 1 Jan., 1990.
12. W. Wen, and A. Xia, "Verifying Edges for visual Inspection Purposes." School of Mechanical and Manufacturing Engineering, School of Mathematics, The University of New South Wales, NSW 2052, Australia, July 1997.
13. Simon Baker, "Design and Evaluation of Feature Detectors." Doctor of philosophy thesis in the Graduate School of Arts and Sciences, Columbia University, 1998.

14. Richard Bridge, “Computer Image Processing.” Tessella support services PLC, Oxon, England, Issue V1.R2.M0, June, 2003.