Image Processing for Detecting Tile Defects Based on Adaptive Threshold

Dr. Heydat Toossian Shandiz  Assistant professor at Shahrood University of Technology hshandiz@shahrood.ac.ir, Hadi Hadizadeh hadipardis@yahoo.com, Shahrood university of technology, 7th Tir Square, Po Box 36155-316, Shahrood, Iran

Abstract-This paper addresses a new method of detecting tile defects. The tile image is changed to binary image by using a float threshold. Using linear regression of binary data gives a measure for classifying the defective. In order to detect a defective tile the defined measure is compared against a threshold which is estimated adaptively through a learning scheme using 4-folded cross validation algorithm. The choice of threshold \( \Gamma \) is determined through a simple training or parameter estimation stage depending on the type of tile texture. The results of applying this method to detect defects on both random and regular textured tiles are highly acceptable.

Keywords: Tile defect detection, Surface inspection, Texture analysis.

I. INTRODUCTION

The automatic detection of defects is one of major problems in industry. In this regard the inspection of tiles on production line has been received a lot of interest from industry side. Although considerable efforts for automatic detection of tile defects have been reported in computer vision literature, there is no algorithm doing this task automatically, with accuracy, low cost and fast. There are three main problems with manual inspection: human tiredness which degrades accuracy, high employment cost and time consuming. Hence, the demand for automated inspection systems is great.

In the computer vision literature there are several works which address the problem of detecting defects on ceramic and tile surface with regular patterns. But very little effort has been carried out on tiles with random texture. Some prominent methods among them will be described briefly here.

Monadjemi et al [1] introduced a method based on restructured eigenfilters for texture abnormality detection within a novelty detection framework which demands only a minimal training stage using a few normal samples.

Boukouvalas et al [2] used optimal filters to detect abnormal lines and spots in tiles. They also used the Wigner distribution to combine the advantages of both spatial and spatial frequency domains to detect cracks. Furthermore in [3], the authors presented a method for detecting random texture tile defects consisting of K-means clustering, followed by perceptual merging of clusters in Luv space and morphological analysis. This was computationally expensive although a promising result was reported. Lopez et al [4] studied the registration methods for ceramic tiles. The basic algorithm was edge detection of the test tile, then obtaining the boundary rectangle, followed by a simple geometrical rotation/displacement to map the test image on the reference. Costa and Petrou [5] employed the Hough Transform (HT) to extract long and straight lines within a tile image. Then a Fourier phase correlation was utilized to register the test and the reference images. Smith and Stamp[6] investigated vision techniques for ceramic tile inspection. Their algorithm attempted to analyze complex surfaces which might include 3D topographic features, by separating the topographic and chromatic maps. Penaranda et al [7] introduced a practical color machine vision system for ceramic tile inspection. The algorithm contains a simple registration by finding four corners of the test tile using a simple procedure, following a background subtraction.

The paper has been organized as follows: in Section 2 our methodology will be described, the results of experiments are presented in Section 3 and finally we will draw the paper to conclusion in section 4.

II. METHODOLOGY

Observing examples of defects on tile surface show that in many cases small spots, thin strips or imperfect edges at the tile borders are considered as defects. These defects have normally very light or very dark intensity in compare with normal texture on tile surface. Considering a dark spot as a defect on tile surface, if the image of tile is changed to binary image using a gradually increased threshold, the defect appears on the binary image with a low threshold level, whereas normal textures appear in binary image with a higher threshold levels. Figure 1 shows the original image of a tile and changed to binary images at different levels. Similar circumstance is also valid for low intensity defects in darker tile texture. There are some well-known defects on tile or ceramic surfaces such as cracks, scratches, pin-holes, spots and chips which usually have dark nature whereas defects such as water drops, color grade and blobs which usually have light nature [2].

Now based on above reasoning we aim to detect defects in the tile image. Let us assume that a \( M \times N \) gray level image of the tile is known. For each threshold level within minimum intensity zero to maximum intensity 255 following five steps are performed.

Step 1: Concatenate image rows to find a new representation of the input image as a \( 1 \times MN \) vector \( Y \). Moreover let assign vector \( X \), as set of indices for vector \( Y \)

\[ X = [1, 2, 3, \ldots, MN] \]

Now define \( D = \{X, Y\} \) as data set.
Step 2: Y as input image is changed to binary using threshold level T, which is initialized to zero.

Step 3: Associated with the data set D, regression line, which is called \( f \) is calculated using the following formula [10]:

\[
\begin{align*}
\alpha &= \mu_x + \beta \mu_y \\
\beta &= \frac{\sum x_i y_i - (\sum x_i \sum y_i)/MN}{\sum x_i^2 - (\sum x_i)^2 / MN}
\end{align*}
\]

Where \( \mu_x, \mu_y \) are the mean values for vectors X and Y respectively.

\( e_{\text{max}} \) Characteristics: Figure 2 shows \( e_{\text{max}} \) as a function of threshold \( T \in \{0,1,2,3, ..., 255\} \) for a typical example of defect on tile surface. As mentioned earlier, up to a certain level of threshold T there is no pixel with intensity value lower than threshold, so the linear regression on the data results a line at the maximum level 255. By increasing the threshold T, at some level \( T_0 \) dark pixels begin to appear in the binary image so the result of linear regression is a line slightly below the maximum level.

Step 4: Error vector E is computed as follows:

\[
E = |y - f|
\]

Step 5: Define \( e_{\text{max}} \) as the maximum error or deviation from the regression line as follows:

\[
e_{\text{max}}(T) = \max\{E\}
\]

With further increasing the threshold level T, at some level the data regression line reduces abruptly which results falling \( e_{\text{max}} \) rapidly. The result of more investigation on the data revealed that rapid decreasing of \( e_{\text{max}} \) is associated with binary natural texture on the tile which constitutes considerable part of the tile surface.

As the level \( T_1 \) is about the intensity of the most texture part on the tile, it is a good feature for characterizing the natural texture from the defects. Based on this observation we state that the difference between level \( T_0 \) and \( T_1 \) can be used as a measure for classification between normal and defective tiles. In presence of dark defects such as spots and cracks (very common) \( e_{\text{max}} \) rises up at \( T_0 \) (the level of defect intensity). This level is quiet far from \( T_1 \) at which natural texture of tile causes a sharp decrease in \( e_{\text{max}} \). In contrast, for normal tiles \( T_0 \) corresponds to darkest part of natural texture which is statistically related to the average intensity of the texture which determines \( T_1 \). The results of experiments showed that the difference between these thresholds (\( T_0 \) and \( T_1 \)) for wide range of textures is lower than for the defective tiles.

Classifying defective tiles: \( \Delta T = T_1 - T_0 \) is used as a parameter to distinguish defectives from quality tiles. We emphasis that our main focus in this paper is, detecting small defects such as spots, thin cracks which appear as high contrast regions on tile surface.

In order to classify image of a give tile one can compare extracted feature \( \Delta T \) against a threshold \( \Gamma \) if \( \Delta T \geq \Gamma \) the tile is labeled as defective, otherwise it is normal. The main question is how we can determine a threshold to classify tiles correctly? As measurements \( T_0, T_1 \) for a normal tile are characteristics of texture on tile surface, setting a predefined value as threshold \( \Gamma \) for detection of defects in different types is a difficult task.

We design a classifier in which threshold \( \Gamma \) is determined adaptively for each type of tile. This threshold
is evaluated through a training procedure using defective and normal tile examples. We specifically use 4-folded cross validation algorithm to train our system.

Let assume that for a specific type of tile there are two training sets: the set of normal examples, P, and the set of defective tiles, Q. If \( \mu_P \) and \( \sigma_P \) denote the mean and standard deviation of the measurement \( \Delta T \) evaluated for normal examples \( P \), respectively. The threshold \( \Gamma \) is modeled as follows [8,9]:

\[
\Gamma = \mu_P + \alpha \sigma_P
\]  

(5)

In other word the optimum threshold \( \Gamma \) is assumed to be \( \alpha \) weighted deviation from the mean of \( \Delta T \) for normal tiles. The parameter \( \alpha \) is estimated using 4-fold cross-validation method [8, 9] as follows.

Let make a training set including both normal and defective examples (mixing \( P \) and \( Q \) sets). We randomly divide the mixed training set into four equal subsets. Each time we select one subset as the training set and three remaining subsets as the validation set. Clearly we have four combinations to select the training and the validation subsets. From each of four cases, we use examples in the training subset to find weight \( \alpha \) in Equation (5) so that the threshold gives equal rates for false positives and false negatives.

A good estimation for unknown parameter \( \alpha \) is the mean of weights provided from the four cases. Note that the validation subset in each case is used to validate the classification performance. If the estimated \( \alpha \) does not give an accepted classification rates the 4-folded algorithm can be repeated by another choice of the training and validation subsets (randomizing the mixed set).

As an example, Figure 6 depicts the distributions of normalized \( T_0 \) for normal and mixed examples of similar type. In this case, the statistics for parameter \( T_0 \) in normal training set \( P \) are \( \mu_P = 0.75098 \) and \( \sigma_P = 0.0036 \). Using the cross-validation algorithm we estimate weight \( \alpha = 0.67 \) from which the optimum threshold \( \Gamma \) is given 0.75342 using Equation (5). The subsequent correct classification rate for this example was 95%.

In order to evaluate the classification correctness we define the following measure as the performance:

\[
C_A\% = \frac{N_{nat} + N_{at}}{N_{tot}} \times 100
\]  

(6)

Where \( N_{nat} \) and \( N_{at} \) are the number of normal and defective tiles which are classified correctly and \( N_{tot} \) is the total test samples.

III. EXPERIMENTAL RESULT

In this section we present the result of experiments for classification of defective from normal tiles using our proposed method. We also compare the performance of our method with a neural network classifier. In figure 6, there are six different cases of defected tiles with their corresponding processed output image. Table 1 show some important parameters of the tiles which depicted at figure 6. From this table one can see that for the tile number 6, the value of \( \Delta T \) in comparison with other tiles...
is very small because the color or gray level of its defect is close to the main texture of it. But the value of it’s $\Delta T^*$ however, is not very different from others. In this table the accuracy of k-fold classifier is showed. It can be seen that the average accuracy of this method is about 92.4%.

It is worth nothing that an attractive feature of the proposed method is its running speed. For instance our algorithm for classification of a 320x240 tile image on machine Pentium IV, 2.8 GHz takes about 0.34 second.

<table>
<thead>
<tr>
<th>Tile</th>
<th>$T_0$</th>
<th>$T_1$</th>
<th>$\Delta T$</th>
<th>$T_m$</th>
<th>$\Delta T^*$</th>
<th>$T^*$</th>
<th>k-fold accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>131</td>
<td>224</td>
<td>93</td>
<td>188</td>
<td>57</td>
<td>165</td>
<td>88.2%</td>
</tr>
<tr>
<td>2</td>
<td>126</td>
<td>245</td>
<td>119</td>
<td>185</td>
<td>59</td>
<td>160</td>
<td>92%</td>
</tr>
<tr>
<td>3</td>
<td>126</td>
<td>250</td>
<td>124</td>
<td>205</td>
<td>79</td>
<td>179</td>
<td>89%</td>
</tr>
<tr>
<td>4</td>
<td>126</td>
<td>246</td>
<td>120</td>
<td>181</td>
<td>55</td>
<td>154</td>
<td>97.6%</td>
</tr>
<tr>
<td>5</td>
<td>193</td>
<td>253</td>
<td>40</td>
<td>231</td>
<td>38</td>
<td>217</td>
<td>95%</td>
</tr>
<tr>
<td>6</td>
<td>171</td>
<td>252</td>
<td>81</td>
<td>228</td>
<td>57</td>
<td>218</td>
<td>93.1%</td>
</tr>
</tbody>
</table>

IV. CONCLUSION

In this paper a new method for identification of high contrast defects in tile surfaces is proposed. As the result shows, no need darker color of defects in respect to their neighbors, they can have lighter color in respect to their neighbors, such as a white defect in a dark textured background. In a real-time processing, in order to speed up and to facilitate the surface inspection process, measuring only the closeness of $T_0$ to $T_m$ or measuring the thinnness of width $\Delta T$ is adequate.

This new method is simple and high speed. In this method the kind of the tile background texture is not important. So it can be implemented directly, regardless of the type of background texture of the input tile image. Moreover this method is not depended on the size of the defects. In other words the recognition accuracy for a black spot defect is the same as a complete black crack defect.

REFERENCES


Figure 6. Some examples for showing the performance of proposed method, left pictures in each column are the original tile image and right pictures are the corresponding outputs.