Coarse Resolution Defect Localization Algorithm for an Automated Visual Printed Circuit Board Inspection

Zuwairie Ibrahim
Zulfakar Aspar
Syed Abdul Rahman Al-Attas
Musa Mohd Mokji
Universiti Teknologi Malaysia
81310 Skudai, Johor, MALAYSIA
(zuwairie, zulfakar, syed, musa)@fke.utm.my

Abstract – **One of the backbones in electronic manufacturing industry is the printed circuit board (PCB) manufacturing. Current practice in PCB manufacturing requires etching process. This process is an irreversible process. Printing process, which is done before etching process, caused most of the destructive defects found on the PCB. Once the laminate is etched, the defects, if exist would cause the PCB laminate become useless. Due to the fatigue and speed requirement, manual inspection is ineffective to inspect every printed laminate. Therefore, manufacturers require an automated system to detect the defects online which may occur during the printing process. Hence, this paper proposes an algorithm for automated visual PCB inspection that is able to automatically detect and locate any defect on a PCB laminate. The defect is detected by utilizing wavelet-based image difference algorithm. The coarse resolution differenced image then is used in order to locate the defective area on the fine resolution tested PCB image.**

I. INTRODUCTION

There exist numerous algorithms, techniques and approaches in the area of automated visual PCB inspection nowadays. As proposed by Moganti [1-2], these can be divided into three main approaches: referential approaches, rule-based approaches and hybrid approaches.

For the referential approaches, there are two major techniques. The first one is image comparison technique and the other one is model-based technique. Image comparison technique consists of comparing the tested PCB image against the reference PCB image using simple XOR logic operator. Model-based technique on the other hand, matches the tested PCB image with a predefined model.

Rule-based approaches test the design rule of the PCB traces to determine whether each PCB trace fall within the required dimensions or not. Mathematical morphological operation is frequently used where dilation and erosion are the basic operation.

Lastly, the hybrid approaches combines the referential approaches and the design-rule approaches to make use the advantages and to overcome the shortcomings of each approach.

This paper is organized as follows. Section 2 mentions about the research methodology of this project. The proposed algorithm entirely can be divided into two stages: the defect detection and the defect localization. The wavelet-based PCB defect detection is addressed clearly in section 3. On the other hand, the coarse resolution PCB defect localization is described in detail in section 4. Section 5 contains the conclusions of this paper. Lastly, the references of this research work are placed in section 6.

II. RESEARCH METHODOLOGY

From the literature review, it is noted that there has been an increasing number of applications for wavelets and multiresolution analysis including (but not limited to) image compression [3], image denoising [4] and edge detection [5]. Up till now there is still no clear advantages of wavelets in industrial inspection application, especially for PCB inspection.

Numerous techniques have been proposed under model-based method so far. Some of them are graph-matching technique, tree representation techniques [6], connectivity technique [7] and RLE-based technique [8].

This paper proposes a slightly different technique in such a way that the PCB images will be modelled by two-dimensional wavelet transform. Then, the image difference operation is applied to the images in the wavelet domain. These techniques are cited in [9-11]. The first advantage of the proposed technique is that wavelet transform can be treated as image-to-image transformation. This will enable the wavelet coefficients to be treated as an image and thus allow the image difference operation to be carried out. Furthermore, wavelet transform provides less data for image difference operation. Thus, less computation time should be expected.

After that, based on the coarse differenced image, the defect localization will be computed in the coarse resolution domain. However, the defective areas are marked on the fine resolution original image of the tested PCB. The defect localization algorithm is able to extract each defective area to provide adequate information for the subsequent inspection stage. The proposed algorithm comprises of two major parts. The first one is wavelet-based image difference algorithm and the second one is coarse resolution PCB defect localization algorithm.

0-7803-7474-6/02/$17.00 ©2002 IEEE

III. WAVELET-BASED PCB DEFECT DETECTION ALGORITHM

A. Wavelets

Wavelet is a zero mean function [12] and satisfies the so-called *admissibility condition* [5],

$$C_\psi = \int_{-\infty}^{\infty} \frac{|\hat{\psi}(\omega)|^2}{\omega} d\omega < \infty \qquad (1)$$

where ψ is a fixed function, called 'mother wavelet' and $\hat{\psi}$ is the Fourier transform of ψ. The constant c_ψ designates the admissibility constant. According to Mallat [12], the continuous wavelet transform (CWT) of a function f is given by:

$$f_\psi(a,b) = \int_{-\infty}^{\infty} f(t) \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) dt \qquad (2)$$

The parameter a is called the dilation factor and has a constraint such that $a > 0$ and b, a real number, is the translation parameter. A wavelet transform decomposes a signal $f(t)$ into many coefficients, which are the function of scale (dilation) and position (translation).

The computation of the wavelet transform of a two-dimensional signal, an image, is applied as a successive convolution by a filter entry of row/column followed by a column/row as depicted by Fig. 1. Thus, for two-dimensional wavelet transform, after the first level wavelet transform operation, the input image can be divided into 4 parts: approximation, horizontal detail, vertical detail and diagonal detail where the size of each part is reduced by the factor of two compared to the input image as depicted by Fig. 2.

Approximation is a compressed and coarser part than the original input image. Meanwhile, the horizontal detail, vertical detail and diagonal detail contain the horizontal, vertical and diagonal components of the input image. When the second level wavelet transform is applied, the approximation part of the first level will be further decomposed into four components as depicted by Fig. 3.

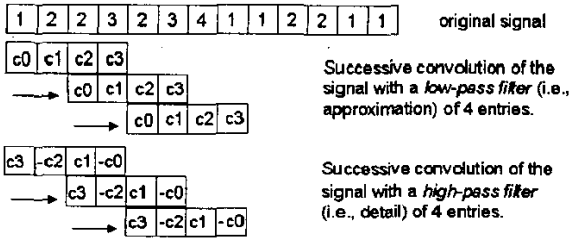

Fig. 1. Successive convolution of wavelet transform.

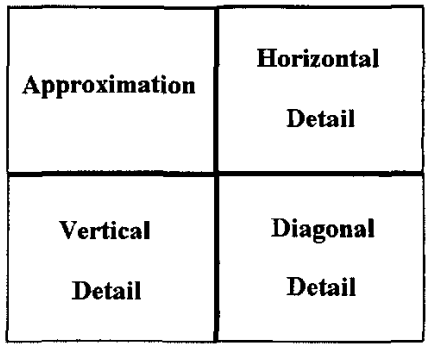

Fig. 2. First level wavelet transform.

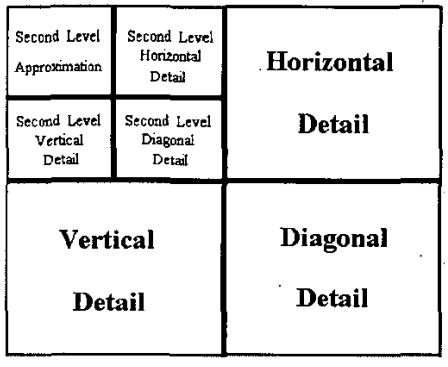

Fig. 3. Second level wavelet transform.

Haar wavelet transform has been chosen for simulation in MATLAB. Haar wavelet has two filter entries as shown in Table 1. By using the Haar wavelet for wavelet-based image difference algorithm, the wavelet transform can be calculated as a block-by-block moving average operation.

B. Image Difference Operation

Consider two binary images named as *Img1* and *Img2* of size M x N where M is the length and N is the height of the image respectively. For each pixel at corresponding location (x,y) that is x from 1 to M, y from 1 to N where both x and y are integers, *Img1[x,y]* and *Img2[x,y]* has a value of either one or zero. In this case, it is assumed that one is identified as background pixel, whereas zero is identified as foreground pixel.

In this project, image difference operation is applied onto the output of the second level wavelet transform and not on the input binary image. The input data is no longer an integer

Table 1. Analysis filter for Haar wavelet (2-filter entry).

	Coefficient 1	Coefficient 2
Approximation Filter (Low Pass)	0.707	0.707
Detail Filter (High Pass)	0.707	-0.707

value but rather a real value. Therefore, for every pixel of *Img1* and *Img2*:

- For *Img1*, grab a pixel value at location *Img1(x,y)* and named it as *PixVal1*.
- For *Img2*, grab a pixel value at location *Img2(x,y)* and named it as *PixVal2*.
- If *PixVal1* is equal to *PixVal2*, assign the corresponding pixel at location *Difference(x,y)* as background pixel.
- If *PixVal1* is *not* equal to *PixVal2*, assign the corresponding pixel at location *Difference(x,y)* as foreground pixel.

C. PCB Defect Detection Algorithm

Two images are needed, the reference image and the test image. Second level wavelet transform is applied to the reference image and then the image and also the wavelet outputs are stored in memory. This step is done offline once only as indicated by the dash box in Fig. 4.

For the test image, as same as the ideal reference image, second level wavelet transform is applied. The flow of the algorithm, which is illustrated in Fig. 4, consists of an example of a reference, a test image, the resultant wavelet transform output and also the defect detected in the output image of image difference operation. The coarse resolution differenced image is shown in Fig. 5.

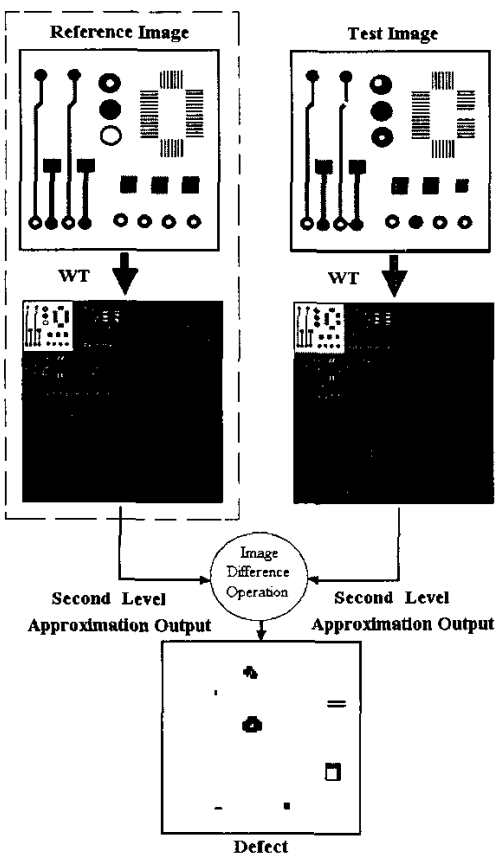

Fig. 4. Wavelet-based image difference operation.

The advantage of wavelet transform is that it preserves most of the information of the original image in the coarse image. Thus, it is possible to detect all the defects occur by simply applying the image difference operation between the coarse reference image and coarse test image. The coarse image processing is only applied to defects detection procedure. However, the defects localization should be highlighted on the original image for better visual perception.

IV. COARSE RESOLUTION PCB DEFECT LOCALIZATION ALGORITHM

The purpose of the defect localization algorithm is to highlight the defective areas on the tested PCB image. The defect localization is important in order to mention the users about the location of the defects detected for further procedures such as defect classification and defect marking.

The input for the defect localization algorithm is the coarse differenced image. The defect localization algorithm consists of four core operations named as connected-component labelling operation, window coordinates searching operation, mapping operation and windowing and defect extraction operation.

A. Connected-component Labelling Operation

Before the connected-component labelling operation is explained, it is essential to understand the meaning of connectivity in a two-dimensional array or image. For a two-dimensional array, there exist two types of connectivity. The first one is 4-connected pixel and the second one is 8-connected pixel [5].

Fig. 6(a) and Fig. 6(b) represent the concept of a 4-connectivity pixel and an 8-connectivity pixel respectively.

The connected-component labelling operation returns the information of the coarse differenced image (a binary image) to identify each object in the image. The output of the connected-component labelling operation is a two-dimensional output array named as labelled image. The size of the labelled image is exactly the same as the coarse

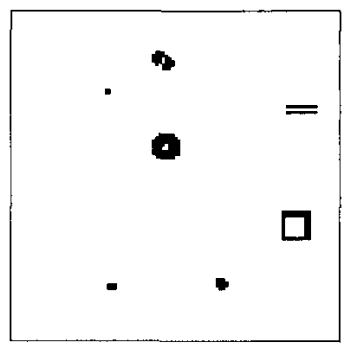

Fig. 5. The coarse resolution difference image.

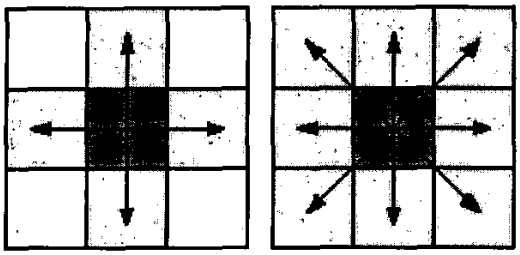

Fig. 6. (a) A 4-connectivity pixel (b) An 8-connectivity pixel.

differenced image are distinguished by different integer values in the labelled image. As an example, consider a small area of a coarse differenced image represented in two-dimensional 10 x 10 array as shown in Fig. 7.

The output array of the 4 connected-component labelling is depicted in Fig. 8. This figure obviously shows that the connected-component labelling operation successfully recognizes 3 objects in the coarse differenced image. In this case, each object is assigned with an identical value starting from 1 to 3. This identical value is depends on the number of objects in the coarse differenced image.

Fig. 9 shows the output array of the 8 connected-component labelling operation. Compared to the output in Fig. 8, the objects identified as 1 and 2 are merged to become one individual object because of the 8-connectivity factor

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	0
0	0	0	0	0	0	1	1	1	0
0	0	0	0	0	0	0	1	0	0
0	1	1	1	0	0	0	0	0	0
0	1	1	1	0	0	0	0	0	0
0	1	1	1	0	0	0	0	0	0
0	0	0	0	1	1	1	0	0	0
0	0	0	0	1	1	1	0	0	0
0	0	0	0	1	1	1	0	0	0

Fig. 7. An example of a small area in a coarse difference image.

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	3	3	3	0
0	0	0	0	0	0	3	3	3	0
0	0	0	0	0	0	0	3	0	0
0	1	1	1	0	0	0	0	0	0
0	1	1	1	0	0	0	0	0	0
0	1	1	1	0	0	0	0	0	0
0	0	0	0	2	2	2	0	0	0
0	0	0	0	2	2	2	0	0	0
0	0	0	0	2	2	2	0	0	0

Fig. 8. The output of the 4 connected-component labelling operation.

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	2	2	2	0
0	0	0	0	0	0	2	2	2	0
0	0	0	0	0	0	0	2	0	0
0	1	1	1	0	0	0	0	0	0
0	1	1	1	0	0	0	0	0	0
0	1	1	1	0	0	0	0	0	0
0	0	0	0	1	1	1	0	0	0
0	0	0	0	1	1	1	0	0	0
0	0	0	0	1	1	1	0	0	0

Fig. 9. The output of the 8 connected-component labelling operation.

chosen for the connected-component labelling operation. Thus, the total number of objects identified in the coarse differenced image is only 2. In this project, the 8-connectivity pixel for the connected-component labelling operation is selected based on the assumption that defects are normally far from each other. Thus, the computation complexity can be avoided.

B. Window Coordinates Searching Operation.

The resultant image of 8 connected-labelling operation is taken to be an input for the window coordinate searching operation. Here, the result in Fig. 9 is taken as an input for the ease of explanation. The most important objective of this operation is to search four coordinates of each object in Fig. 9 for the defective area windowing. The four coordinates of each object are named as: *RowMin, RowMax, ColMin* and *ColMax* correspond to minimum row, maximum row, minimum column and maximum column respectively. Note that this operation is done on the coarse resolution image. With Fig. 9 as the input image for window coordinate searching operation, the output of this operation is shown in Table 2. As an example, Fig. 10 represents the location of the *RowMin, RowMax, ColMin* and *ColMax* for the object identified by number 2 in a 10 x 10 array.

C. Mapping Operation.

According to the coordinates obtained in window coordinate searching operation, a number of windows are drawn on the fine resolution tested image. Recall that these four coordinates of each object is defined in a coarse resolution image. Hence, some sort of mapping equation is needed to map the coordinates in the coarse resolution image to the fine resolution image. The determination of the mapping equation is critical in the sense that ineffective mapping equation will cause the distortion problem happened to the individual drawn window. In order to derive the mapping equation employed in this paper, four coordinates *RowMin, RowMax, ColMin* and *ColMax* represented in Fig. 10 is selected again. These four coordinates are chosen to simplify the justification of the mapping equation.

Table 2. Output of window coordinate searching operation.

Object	RowMin	RowMax	ColMin	ColMax
1	5	10	2	7
2	2	4	7	9

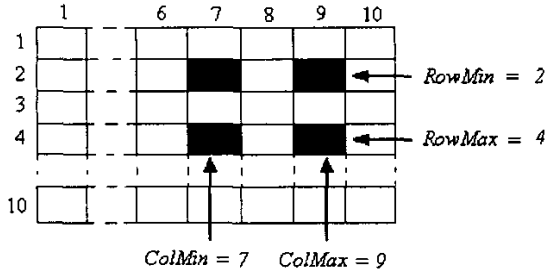

Fig. 10. Representation of *RowMin, RowMax, ColMin and ColMax* for the object number 2.

If the 10 x 10 array in Fig. 10 is to be enlarged into an 80 x 80 array, as an example, the output of the image enlargement should be as depicted in Fig. 11. In this case, the enlargement coefficient, E is 8. Apparently, each four point as small as one pixel is enlarged to be 8 x 8 pixels in the enlarged image. Then, each coordinate is mapped into 8 possible values actually as shown in Fig. 11. However, for the windowing operation, the mapping equation should be able to accomplish one point to one point mapping. In order to solve this matter, two simple set of rule are considered.

Suppose that if each coordinate *RowMin, RowMax, ColMin* and *ColMax* on the coarse resolution image is mapped into *RowL, RowH, ColL* and *ColH* on the fine resolution image, then:

1. The value of *RowL* and *ColL* should be a minimum value within the range of possible values.
2. The value of *RowH* and *ColH* should be a maximum value within the range of possible values.

Based on the rules, the equation 3, 4, 5 and 6 can be used for the mapping operation.

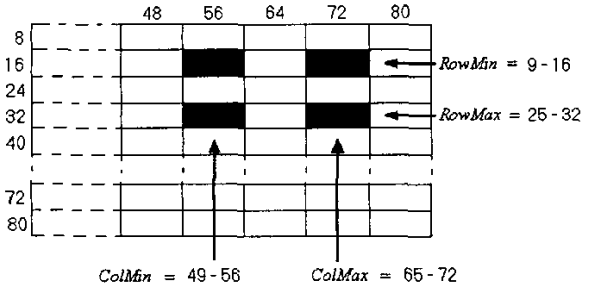

Fig. 11. The enlargement of the coarse image in Fig. 10 to a fine image (enlargement coefficient, $E = 8$).

$$RowL = (RowMin)(E) - (E-1)$$
$$RowL = (RowMin)(2^L) - (2^L - 1) \quad (3)$$

$$RowH = (RowMax)(E)$$
$$RowH = (RowMax)(2^L) \quad (4)$$

$$ColL = (ColMin)(E) - (E-1)$$
$$ColL = (ColMin)(2^L) - (2^L - 1) \quad (5)$$

$$ColH = (ColMax)(E)$$
$$ColH = (ColMax)(2^L) \quad (6)$$

where $E = 2^L$ and L denotes the iteration or level of wavelet transform used in the wavelet-based image difference algorithm.

D. Windowing and Defect Extraction Operation

For each *RowL, RowH, ColL* and *ColH* related to each defective area, a boundary line representing a defective window can be drawn on the fine resolution tested PCB image. Each window marks the defective areas where the defects are actually occurred. After the defective areas are windowed successfully, it is possible to segment each defective area for defect extraction where each defective area is shown in an individual image.

The result of the defect localization is depicted in Fig. 12. The black windows on the grey pattern highlight the defective areas on the tested PCB image. The flow of the proposed defect localization algorithm is well shown in Fig. 13.

V. CONCLUSIONS

This paper proposes an algorithm for PCB defect detection and localization for an automated visual PCB inspection. The localized area in the tested PCB image will be used as the inputs to the classification stage, which is the subsequent stage after the defect detection has been done. The continuation of this research is to implement the algorithm on hardware to ensure that the automated visual PCB inspection system can perform in a real-time environment with high efficiency.

VI. REFERENCES

[1] M. Moganti, F. Ercal, C. H. Dagli and S. Tsunekawa, "Automatic PCB inspection algorithms: A survey," *Computer Vision and Image Understanding*, vol. 63, no. 2, March. 1996, pp. 287-313.

[2] M. Moganti and F. Ercal, "Automatic PCB inspection systems," *IEEE Potentials*, vol. 14, no. 3, August. 1995, pp. 6-10.

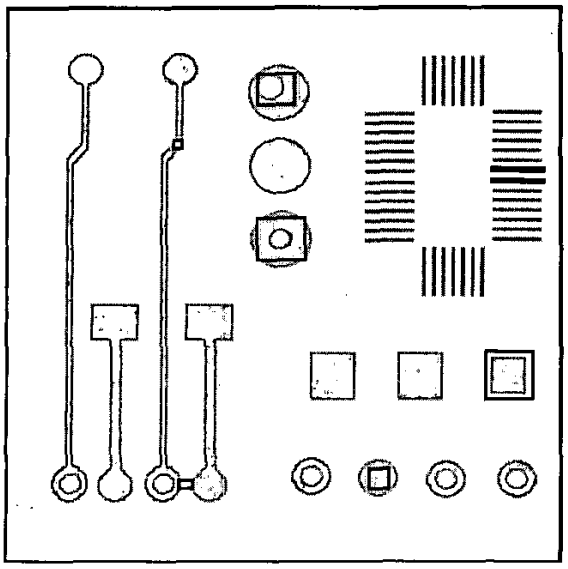

Fig. 12. Defect localization.

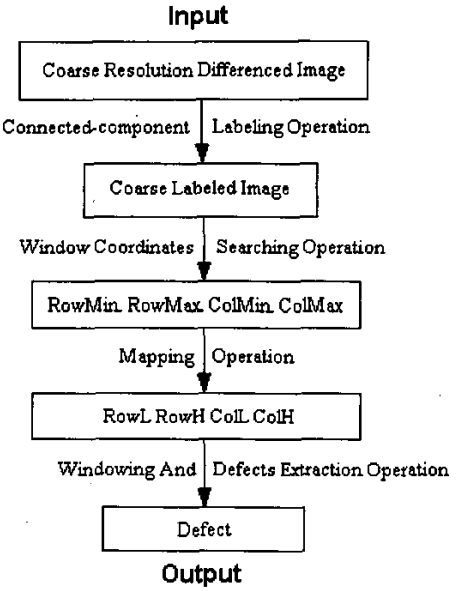

Fig. 13. Coarse resolution defect localization algorithm.

[3] F. G. Meyer, A. Z. Averbuch and J. Stromberg, "Fast adaptive wavelet packet image compression," *IEEE Transaction on Image Processing,* vol. 9, no. 5, 2000.

[4] F. Guoliang and X. Xiang-Gen, "Wavelet-based statistical image processing using hidden markov tree model," in *Proceedings of the 2000 Conference on Information Sciences and Systems.*

[5] R. L. Rao, *Multiresolution Technique in Image Processing,* Louisiana State University, PhD Thesis, 1995.

[6] H. K. Ja and S. I. Yoo, "A structural matching for two-dimensional visual pattern inspection," *IEEE International Conference on Systems, Man and Cybernatics,* vol. 5, 1998, pp. 4429-4434.

[7] M. H. Tatibana and R. A. Lotufo, "Novel automatic PCB inspection technique based on connectivity," in *Proceedings of the 1997 Brazilian Symposium on Computer Vision and Image Processing*, pp. 187-194.

[8] F. Ercal, F. Bunyak, H. Feng and L. Zheng, "A fast modular RLE-based inspection scheme for PCBs," in *Proceedings of the 1997 SPIE – Architectures, Networks and Intelligent Systems for Manufacturing Integration,* vol. 3203, pp. 45-59.

[9] Z. Ibrahim, S.A.R. Syed Abu Bakar and Z. Aspar, "Wavelet based algorithm for image comparison technique in application of coarse-to-fine inspection of printed circuit board (PCB)", *Malaysian Science and Technology Congress*, October. 2001, (Presented).

[10] Z. Ibrahim, S.A.R. Al-Attas, Z. Aspar and R. Ghazali, "Performance evaluation of wavelet-based algorithm for printed circuit board (PCB) inspection", *Jurnal Teknologi*, no. 35(D), December. 2001, pp. 39-54.

[11] Z. Ibrahim, S.A.R. Al-Attas and Z. Aspar, "Analysis of the wavelet-based image difference algorithm for PCB inspection," in *Proceedings of the 2002 41st Annual Conference (SICE 2002),* pp. 1525-1530.