Reference-free detection of semiconductor assembly defect

Ada N.Y. Ng[a], Edmund Y. Lam[a], Ronald Chung[b], Kenneth S.M. Fung[c], W.H. Leung[c]

[a]University of Hong Kong, Pokfulam Road, Hong Kong, China
[b]Chinese University of Hong Kong, Hong Kong, China
[c]ASM Assembly Automation Ltd, Hong Kong, China

ABSTRACT

This paper aims at developing a novel defect detection algorithm for the semiconductor assembly process by image analysis of a single captured image, without reference to another image during inspection. The integrated circuit (IC) pattern is usually periodic and regular. Therefore, we can implement a classification scheme whereby the regular pattern in the die image is classified as the acceptable circuit pattern and the die defect can be modeled as irregularity on the image. The detection of irregularity in image is thus equivalent to the detection of die defect. We propose a method where the defect detection algorithm first segments the die image into different regions according to the circuit pattern by a set of morphological segmentations with different structuring element sizes. Then, a feature vector, which consists of many image attributes, is calculated for each segmented region. Lastly, the defective region is extracted by the feature vector classification.

Keywords: defect detection, semiconductor assembly process, image analysis, feature detection

1. INTRODUCTION

With the advancement of electronic technology, ICs become the fundamental component of all electronic devices. There is notable growth in the ICs production in recent years. Throughout the semiconductor assembly and packaging processes, even a small dust particle or dirt can cause malfunction in the ICs at the end. Those ICs with defects cause loss in the production yield. In order to enhance the production yield, defect detection of the IC dies is ubiquitous during the manufacturing process.

It is often difficult for human inspector to detect the tiny defect on the small integrated circuit with complex circuit pattern. Different human inspectors often have different standards and judgments on the defects. Also the human detection is a labor intensive step and could become the critical bottleneck in the production process. Therefore, human detection is not competitive enough when compared with the automated machine vision system, which is often more effective, efficient and easier for quality control.

The machine vision system acquires the digital image of the die and checks if there is any defect. The common reference-based defect detection algorithm requires two or more images during the inspection. For instance, the golden image template reference approach needs an extra defect-free image for each type of die to be the reference.[1] In addition, the rotating, translating, adjusting brightness and aligning processes between the reference image and the inspect image are needed before the comparison. Obviously, all these necessitate some pre-processing before the inspection and sometimes the reference database is simply not available. Therefore this can be a burden on the whole detection system. An automatic and reference-free defect detection approach can alleviate this burden. It can detect and classify the die defect automatically through the use of only one die image. There exists a non-referential defect detection algorithm for the semiconductor wafers.[2,3] However, instead of distinguishing between the non-defective and the defective die images, that algorithm is mainly targeted on the redetection of defects within the historical defect imagery.

The reference-free defect detection algorithm proposed here is a novel algorithm for the classification of non-defective dies and defective dies by analyzing a single captured image, without reference to any image during inspection in the semiconductor assembly process. It is known that the IC pattern is usually periodic and very regular. In other words, the circuit pattern appears repeatedly in a die. But the die defect, likes dirt, dust particle, probe mark and ink, is aperiodic and can be random in shape. Moreover, the defect appears locally and abnormally in a die. Therefore, we can implement a classification scheme whereby the regular pattern in the die image is classified as the acceptable circuit pattern and the die defect can be modelled as irregularity on the image. The detection of irregularity in image is thus equivalent to the detection of die defect.

The irregularity detection algorithm first segments the die image into different regions according to the circuit pattern. To optimize the segmentation for different defect sizes and shapes, a set of morphological segmentations with different structuring element sizes are used. By carefully linking and comparing the adjacent levels of the morphological segmentation, we can identify a proper segmentation with minimal number of segments. Then, a feature vector is calculated for each segmented region. The feature vector consists of many image attributes, such as size and intensity. Some of these attributes are very outstanding for different classes of defects. Regularity measurement of each region is classified based on the feature vector. Lastly, the irregular region is extracted and classified as a defect. Fig. 1 shows the block diagram of the proposed algorithm.

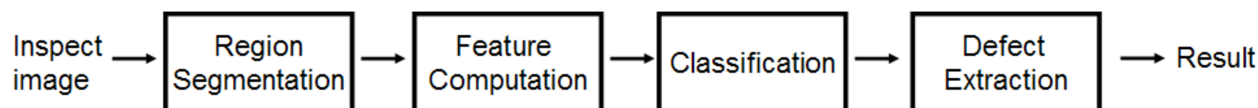

Figure 1. The block diagram of the reference-free detection of semiconductor assembly defect algorithm.

This paper is organized as follows: Section 2 describes the morphological image segmentation algorithm. Section 3 discusses the generation of the feature vector. Section 4 explains the classification algorithm used. Section 5 illustrates the results and evaluation and Section 6 concludes the paper.

2. SEGMENTATION OF DIE IMAGE

To inspect the die image, first the machine vision system needs to know what the image looks like. The image segmentation is typically used for this kind of analysis in the digital image. It can divide an image into different homogeneous regions, where the pixels are connected and have similar properties, such as the pixel intensity[4] within the region, and sketch out the regions' boundaries according to the image components. In the die images, these homogeneous regions represent the circuit pattern and any possible defect on the die.

A diversity of image segmentation algorithm exists, however different types of images need different algorithms to obtain reasonable results. For instance, using thresholding to segment the images usually works well in images with clear background and foreground, but unsatisfactory results are often obtained in images with complex pattern. In order to optimize the segmentation for various circuit patterns and defects, which usually have different sizes and shapes, a multiscale morphological segmentation is used in our work.

The multiscale morphological segmentation utilizes the method of mathematical morphology[5] by using structuring elements with different sizes. A small structuring element is aimed at extracting the small image components, while a large structuring element is targeted for the large image components. Together they help to extract image components of different size.[6]

2.1. Multiscale morphological segmentation algorithm

There are two major steps in the multiscale morphological segmentation algorithm for the die images.[6] In the first step, the image components of different sizes are extracted by using structuring elements with ascending size and the top-hat and bottom-hat morphological operations. Then a binary image is formed to record the extracted image components for each size of the structuring element. In the second step, the boundaries of the homogeneous regions are decided based on the information in the binary images.

2.1.1. Extraction of various-size image components

The noise of the input die image is first smoothed by the local monotonicity preprocessing method outlined in Ref. 7. This method can decrease the noise impact in the segmentation while preserving the edges of the circuit patterns and defects. Then the preprocessed die image X goes through the top-hat and bottom-hat transformations with structuring elements S_i of ascending scale i to produce the different pairs of morphologically transformed images P_i and Q_i[6] with

$$P_i = X - X \circ S_i \tag{1}$$

$$Q_i = X \bullet S_i - X, \tag{2}$$

where P_i represents the set of morphological transformed die image from the top-hat operation and structuring element with scale i, '$\circ$' represents the morphological opening operation and '$\bullet$' represents the morphological closing operation. P_i contains bright image components that are the image components with high pixel intensity and with similar size to the corresponding structuring element. And Q_i is the set of transformed image from the bottom-hat operation and contains dark image components that are the image components with low pixel intensity.

It is known that the intensity values of the pixels within a circuit pattern or defect are not all in the same value. Due to this small fluctuation of pixel intensity, some image components with low contrast in P_i and Q_i are often the brightest or darkest part of some large circuit patterns or defects on the die image. As they cannot represent the whole circuit pattern or defect and they like the noise in the segmentation, so removing them can provide better segmentation result. Therefore the P_i and Q_i are then converted into binary images T_i and B_i respectively, to preserve the image components with significant contrast only.

Instead of using thresholding, which ignores the connectivity of the image components, to decide the T_i and B_i, the morphological H-maxima transformation H_{max} is introduced with

$$H_{max}(f, c) = R_f^{\delta}(f - c), \tag{3}$$

where R_f^{δ} is morphological reconstruction of f by dilation, f is the image to be transformed and c is the value of the significant contrast.[8] Therefore,

$$T_i = H_{max}(P_i, c) \tag{4}$$

$$B_i = H_{max}(Q_i, c). \tag{5}$$

This is because the H_{max} can discard the image components extracted in the P_i and Q_i with contrast less than c and can maintain the connectivity. That means the shape of the image components can be preserved under the H_{max}.

The larger the i, the larger the circuit pattern or defect can be extracted. However, when i is larger than the circuit pattern or defect with the largest size, further increase in i is meaningless. Therefore it is very important to automatically stop the growth of the i, so that the repeating of the morphological top-hat and bottom-hat transformation and the H-maxima transformation with increasing i can be stopped automatically when enough image components are extracted in T_i and B_i. But the size information of the circuit pattern or defect is often unknown before the segmentation. In order to achieve this, an edge map comparison method is proposed.

The edge image E of the input die image is employed to measure when the number of image components extracted in the T_i and B_i is enough. The E is generated by Canny's edge detector[9] and it contains almost all the edges of the circuit patterns or defects on the die image. Since each structuring element is associated with two binary images T_i and B_i, therefore the pair of the binary images are united together temporarily to form a single image U_i for the edge measurement

$$U_i = T_i \bigcup B_i. \tag{6}$$

If the number of edges of the extracted image components in U_i is equal to or greater than the number of edges in the E, almost all the circuit patterns or defects are extracted out. However the edges in E can be disconnected, so the number of edge segments in E cannot represent the number of real edges of the die image. Therefore the number of pixels is used as the metric for the edge measurement. The intersection between U_i and E is employed to form the edge map A_i, which indicates how many image components are extracted in scale i with reference to E. Then the ratio of pixels in A_i to that in E is calculated, where k ranges from 0 to 1 and num is the function that counting the number of pixels in an image:

$$A_i = U_i \bigcap E \tag{7}$$

$$k_i = \frac{num(A_i)}{num(E)}. \tag{8}$$

The higher value of k_i, the more image components are extracted. Therefore the growth of i and the morphological transformations can be stopped when k_i is larger then or equal to a certain threshold, for example, 0.9.

2.1.2. Decision of the segment boundaries

T_i and B_i contain the location information of the extracted image components under various scales i. The image components in small scale is usually growing across the larger scales until their sizes are similar to their corresponding circuit patterns or defects. However, some of the extracted image components from different circuit patterns or defects may be fused into one large image component in some large scale. If this kind of fusion happens in a certain scale, this scale is regarded as the "unreasonable growth scale". In order to decide the boundaries of the segments, it is necessary to check how the image components grow across all the scales.

Different circuit patterns and defects are mainly distinguished by their own pixel intensities. Therefore the mean pixel intensity of the image component is adopted to measure the growth of the component. The fusion of different components from different circuit patterns or defects will change the corresponding mean intensities across the two scales significantly. So the change of the mean pixel intensity can indicate how homogeneous a region is.

The steps for the segment boundaries decision are described as follows: first the mean pixel intensity of each extracted image component across all the scales is calculated based on the location information in T_i and B_i. Then a mean-intensity trace for each extracted image component is formed from the smallest scale to the largest scale. By checking how this mean-intensity trace changes across all the scales, the correct scale for that image component can be decided. When the maximum change of the mean-intensity trace across two certain scales is larger than a threshold, this indicates that there is an unreasonable growth or fusion of the components. Therefore the larger scale is considered to be the "unreasonable growth scale". Then the image component in the smaller scale is chosen and its boundary is extracted. Fig. 2 shows the segmentation result.

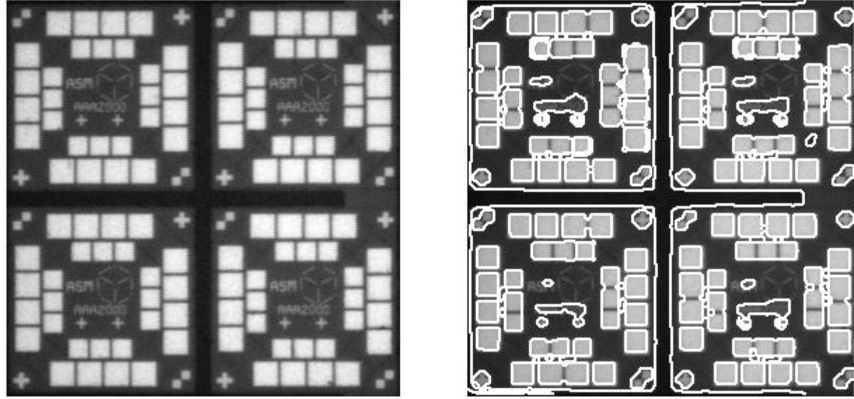

Figure 2. Left: The original die image. Right: The multiscale morphological segmentation result. The boundary is in white color.

3. GENERATION OF FEATURE VECTOR

After the region segmentation, instead of pixels with different intensity values, the die image is represented by different segments. In order to classify the segments through the computer, we need to describe the segment mathematically. Therefore each segment is represented by a feature vector according to the segment's properties. The feature vector has ten dimensions and the elements are listed in Table 1.

Area	Compactness
Eccentricity	Major axis length
Minor axis length	Mean intensity
Rectangularity	Rectilinearity
Solidity	Standard deviation of intensity

Table 1. The elements in the feature vector.

Each element in the feature vector can characterize the segment in a certain aspect. However, using each of them alone is not enough to classify different circuit patterns and defects. Therefore they are combined together to form a feature vector to model each segment. Only some of the elements in the feature vector are defined based on the ellipse, but they can also be obtained in other shapes. In addition, in order to minimize the time requirement for the feature vector calculation, the simplest and most efficient way to calculate the element is used even if there exists another slower but more accurate method to calculate it. The following is a brief explanation of each element in the feature vector:

- The area of the segment is the number of pixels in the corresponding segment. It can be used to group circuit patterns with similar sizes together.
- The compactness is the measurement of how efficient a boundary can enclose the area. A circle encloses the maximum area for a given perimeter. Therefore the compactness of a circle equals to 1 and other shapes will have smaller values of compactness. The formula of the compactness[10] is

$$Compactness = \frac{4\pi Area}{Perimeter^2}. \tag{9}$$

- The eccentricity is the ratio of the distance between the foci of an ellipse and its major axis length.[11]
- For the major axis length and the minor axis length, an ellipse that has the same second-moments as the segment is used. The major axis length is the length in pixels of the major axis of that ellipse, while the minor axis length is the length of the minor axis[11] as shown in Fig. 3.

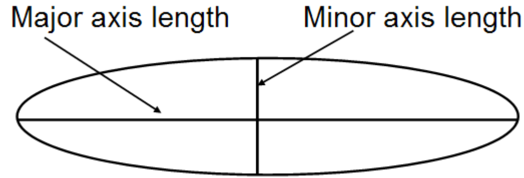

Figure 3. The illustration of major axis length and minor axis length.

- The mean intensity is the average pixel intensity value of the segment. It is expected that the defect has a different mean intensity with the other circuit patterns.
- The rectilinearity ranges from 0 to 1.[12] For a rectilinear polygon, the rectilinearity of it is 1. And a polygon is rectilinear if its interior angles are either 90° or 270°. Fig. 4 shows a cross, which is an example of the rectilinear polygon.

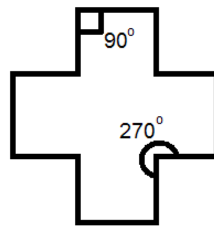

Figure 4. A rectilinear polygon.

- The rectangularity is the area ratio of the segment to the segment's minimum rectangular bounding box.[13] This bounding box is the minimum rectangle that can enclose the segment.
- The solidity is the area ratio of the segment to the segment's convex hull. The convex hull is the smallest convex polygon that can contain the segment.[11]
- The standard deviation of intensity measures how homogeneous a segment is by calculating the standard deviation of the pixel intensities within the segment.

4. CLASSIFICATION OF DIE

Subsequent to the computation of feature vector in each segment, we classify the segments into non-defective segments or defective segments. Defect does not belong to the circuit pattern, therefore it is expected that the feature vector of the defective segment is different from that of the normal circuit pattern. If there is any defect on the die, then after the classification of feature vector, the abnormal feature vector can be found out and the defect can be detected.

Let x_{i1} be the value of the first element in the feature vector of segment i, x_{i2} be the second element and so on. Then in each segment, the x_{ij} (for $j = 1, 2, ..., 10$) is associated with a weighting coefficient a_j, which can be found by using a neural network which models the Equation (10):

$$a_1x_{i1} + a_2x_{i2} + a_3x_{i3} + ... + a_{10}x_{i10} \begin{cases} \leq 0 & \text{if the segment is non-defective ;} \\ > 0 & \text{if the segment is defective.} \end{cases} \tag{10}$$

The type of neural network used is the perceptron network.[14] The input of the neural work is the values of the ten elements in each feature vector. There is one hidden layer in the neural network and one node in that layer. Each input element value is connected to the node through the weighting coefficient a_j. Then by training the neural work, the set of weighting coefficient a_j can be obtained.

Through computing Equation (10), the defect classification can be achieved. If there is no defect on the die, then all the segments' values calculated in the Equation (10) are smaller or equal to zero. If there is any defect likes the ink or dirt on the die, then that value of the defect's segment is greater than zero.

5. DETECTION RESULTS AND DISCUSSION

Several different types of dies are tested under the proposed defect detection algorithm. The circuit pattern and size of these dies are different. The input to our detection algorithm is only the digital image of each die. The output indicates whether the die is defective or not and locates the defect(s) on the defective die. Fig. 5 to Fig. 7 show the detection results by extracting the detected defects on the defective dies.

The dirt defect in Fig. 5 is successfully detected and the defect edge is correctly traced out. But for the bottom left defect in Fig. 6, only part of the defect is detected. It is because the value of Equation (10) for the undetected defect is very close but smaller than zero and not all part of the defect is segmented out. In Fig. 7, besides the large defect in the middle of the die, there are also some little defects around the die. They cannot be detected is because the little defects is too small to be segmented out where the minimum size of the feature to be segmented out depends on the size of structuring element S_i.

In our multiscale morphological segmentation, the square structuring element is used. It is because the circuit pattern is usually rectangular. The size of structuring element S_i is n_i pixel(s) $\times$ n_i pixel(s) and $\triangle n$ is the increment of n across two adjacent scales. If $\triangle n$ is equal to 1, the growth of the structuring element is slow. And it is possible that the image components extracted in scale i and $i + 1$ are the same when i is small. Therefore we set $\triangle n$ in 4 pixels and n_1 in size 4 pixels $\times$ 4 pixels. These values can extracted the circuit pattern and defect out with only little overlapping in the image components extracted in adjacent scales. In addition, by using larger $\triangle n$, the structuring element can growth faster. As a result, the number of iteration for the morphological transformations decreases and the segmentation becomes faster.

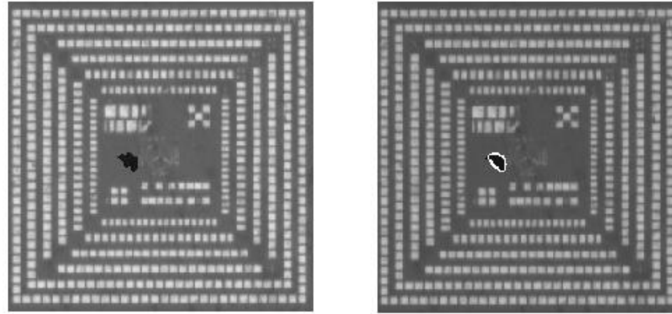

Figure 5. Left: The die with dirt in the middle. Right: The defect detection result. The defect's boundary is in white color.

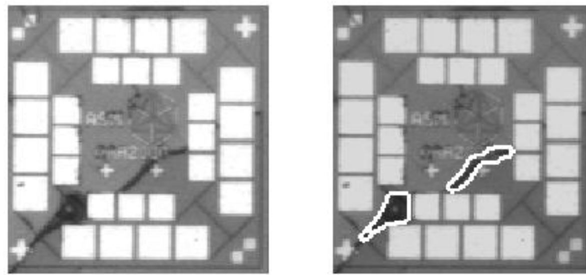

Figure 6. Left: The die with ink defects in the middle and bottom left corner. Right: The defects detection result. The defects' boundaries are in white color.

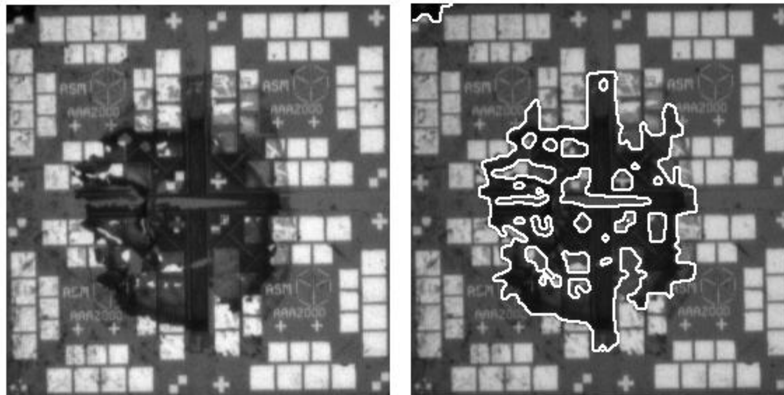

Figure 7. Left: The die with a large defect in the middle and a defect in the top left corner. Right: The defects detection result. The defects' boundaries are in white color.

6. CONCLUSION

A novel defect detection algorithm for the semiconductor assembly process by image analysis of a single captured image, without reference to another image during inspection, is proposed. It can distinguish between the defective die and non-defective die effectively. The detection result depends on both of the sensitivity of the segmentation algorithm and the setting of the threshold in the classification. Although the exact edge of some defect may not be able to trace out, the location of the defect can be found out successfully.

7. ACKNOWLEDGMENTS

This work is supported in part by the Innovation and Technology Fund of the Hong Kong Special Administrative Region Government in co-operation with ASM Assembly Automation Ltd.

REFERENCES

1. S.-U. Guan and P. Xin, "A golden block self-generating scheme for continuous patterned wafer inspections," in *10th International Conference on Image Analysis and Processing*, pp. 436–441, Sept. 27-29 1999.
2. R. K. Ferrell, S. S. Gleason, and K. W. Tobin, "Application of fractal encoding techniques for image segmentation," in *Sixth International Conference on Quality Control by Artificial Vision*, **5132**, pp. 69–77, SPIE, May 19-22 2003.
3. S. S. Gleason, R. K. Ferrell, T. P. Karnowski, and K. W. Tobin, "Detection of semiconductor defects using a novel fractal encoding algorithm," in *Design, Process Integration, and Characterization for Microelectronics, Proceedings of SPIE*, **4692**, pp. 61–71, March 6-7 2002.
4. B. Sumengen, B. S. Manjunath, and C. Kenney, "Image segmentation using curve evolution and region stability," in *16th International Conference on Pattern Recognition*, **2**, pp. 965–968, 2002.
5. R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Prentice Hall, second ed., 2002.
6. S. Mukhopadhyay and B. Chanda, "Multiscale morphological segmentation of gray-scale images," *IEEE Transactions on Image Processing* **12**, pp. 533–549, May 2003.
7. J. H. Bosworth and S. T. Acton, "Morphological image segmentation by local monotonicity," in *Thirty-Third Asilomar Conference on Signals, Systems, and Computers*, **1**, pp. 53–57, Oct. 24-27 1999.
8. P. Soille, *Morphological image analysis: principles and applications*, Springer, second ed., 2003.
9. J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **8**, pp. 679–698, Nov. 1986.
10. R. J. Schalkoff, *Digital image Processing and computer vision.* John Wiley and Sons Inc., 1989.
11. *Image Processing Toolbox User's Guide.* MATLAB 7.0.
12. J. Zunic and P. L.Rosin, "Rectilinearity measurements for polygon," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **25**, pp. 1193–1200, Sept. 2003.
13. P. L. Rosin, "Measuring rectangularity," *Machine Vision and Applications* **11**(4), pp. 191–196, 1999.
14. F. Rosenblatt, *Principles of Neurodynamics.* Washington D.C.: Spartan Press, 1961.