Journal of Zhejiang University-SCIENCE C (Computers & Electronics)
ISSN 1869-1951 (Print); ISSN 1869-196X (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Void defect detection in ball grid array X-ray images using a new blob filter*

Shao-hu PENG, Hyun Do NAM‡

(*Department of Electronics and Electrical Engineering, Dankook University, Yongin 448-701, Korea*)

E-mail: pengshaohu@hotmail.com; hdnam@dankook.ac.kr

Received Mar. 15, 2012; Revision accepted Aug. 9, 2012; Crosschecked Sept. 11, 2012

Abstract: Ball grid arrays (BGAs) have been used in the production of electronic devices/assemblies because of their advantages of small size, high I/O port density, etc. However, BGA voids can degrade the performance of the board and cause failure. In this paper, a novel blob filter is proposed to automatically detect BGA voids presented in X-ray images. The proposed blob filter uses the local image gradient magnitude and thus is not influenced by image brightness, void position, or component interference. Different sized average box filters are employed to analyze the image in multi-scale, and as a result, the proposed blob filter is robust to void size. Experimental results show that the proposed method obtains void detection accuracy of up to 93.47% while maintaining a low false ratio. It outperforms another recent algorithm based on edge detection by 40.69% with respect to the average detection accuracy, and by 16.91% with respect to the average false ratio.

Key words: Ball grid array (BGA), X-ray, Defect detection, Blob detection, Void detection
doi:10.1631/jzus.C1200065 **Document code:** A **CLC number:** TP391; TN45

1 Introduction

Due to its advantages in high interconnection density, small size, high I/O port density, and good electric characteristics, the ball grid array (BGA) has been widely used in printed circuit assemblies (PCAs) requiring a large number of I/O devices. However, during the BGA assembly process, solder joint void defects usually occur, which may greatly degrade the performance of the circuit.

BGA voids, which are caused by outgassing flux that gets entrapped in the solder joint during reflow, are defined as cavities formed in the solder joint (Lin, 2007). Voids in solder joints are one of the most critical defects, since they can affect the reliability and the electrical and thermal conducting performance of the solder joint.

X-ray radiography is a powerful and non-destructive technique used to detect the invisible defects in a BGA package. It has been widely used in many manufacturing factories. In a conventional inspection system, after capturing the BGA X-ray images, the images are then manually examined by an operator to determine if there are defects in each ball. Since there can be more than 15 images for an operator to examine per board, the radiologists can experience fatigue due to the large number of X-ray images. Furthermore, manual defect inspection is time consuming, and the inspection result can vary according to the skill and alertness of the operator. Therefore, to improve the inspection results and reduce the workload of the operator, the development of an automatic inspection system based on image processing is definitely required.

There are several challenges to overcome for automatic void detection using image processing techniques: (1) some of the voids have poor contrast, which makes void detection difficult even to the

‡ Corresponding author

* Project supported by the Dankook University 2010 Funding for Research Institute of Information and Communication Convergence Technology (RICT), Korea

human eye, (2) voids have different sizes and random positions in the ball, (3) due to various voltage settings in the X-ray machine and structure characteristics of the ball, BGA images have non-uniform brightnesses, and (4) there can be interference from the other components on the board. Fig. 1 shows some examples of these difficulties. As can be seen, it is difficult to detect the void shown in Fig. 1a due to the poor contrast (marked with a rectangle). Fig. 1b shows different void sizes and void positions. The problem of non-uniform brightness is illustrated in Fig. 1c, showing that the outer part of the ball is brighter than the inner part. In Fig. 1d, the ball has interference from other components, which causes spurious lines. In addition, the balls shown in Figs. 1a and 1d are brighter than those in Figs. 1b and 1c.

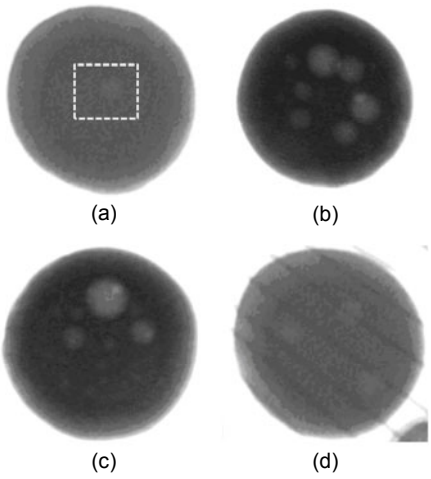


Fig. 1 Several challenges to overcome for automatic void detection using image processing techniques
(a) Poor image contrast (marked with a rectangle); (b) Different void sizes and random position; (c) Non-uniform brightness; (d) Interference from other components

Automatic detection of BGA defects, such as ball deformation, missing balls, solder bumps, and solder bridges, has been studied in the past decade (Rooks *et al.*, 1995; Sumimoto *et al.*, 2002; Xia *et al.*, 2004; Teramoto *et al.*, 2007). However, only a few relevant studies have focused on void defect detection. Sankaran *et al.* (1998) applied neural networks to BGA images to identify different defect classes, such as insufficient solder and voids. Sa-nguannam and Srinonchat (2008) proposed a new image processing technique that can detect defects such as short circuits and ball size, as well as voids. They located the solder balls by image binarization and a labeling technique. The information regarding the solder balls such as location, centroid, area, perimeter, and length were extracted and used to judge the defects. The voids were detected using the solder area information. Due to the non-uniform brightness problem, this method has limitations in precise void detection. Recently, Said *et al.* (2010) proposed a robust automatic void detection algorithm that detects voids with different sizes, different brightness conditions, and board component overshadowing. First, an automatic thresholding method based on a histogram analysis was employed to separate the balls. The occluded areas were then checked and the complete balls were extracted. Finally, the voids were detected using the Laplacian of the Gaussian edge detector and a false void region rejection. However, the algorithm may fail to obtain clear void edges in the case of low image contrast, which can cause missed detections.

The motivation of this study is to develop an automatic void detection system that can overcome the mentioned difficulties and precisely detect the voids. In response, a blob filter based on the image gradient magnitude and a multi-scale analysis is presented in this paper. The proposed blob filter uses the local image gradient magnitude to overcome the brightness, void position, and component interference problems. To detect different sizes of voids, different sized average box filters are employed that set up an image pyramid for multi-scale analysis. Fig. 2 shows the proposed automatic void detection system. In the pre-processing step, the input image noise is reduced by a Gaussian filter. The balls are then extracted using image binarization and a region growing method. Then, for each extracted ball, the contrast is enhanced by the histogram stretch method; the result is sent to the second step. In the next step, the proposed blob filter is performed. First, the input enhanced image is used to form an integral image for fast computation. The box filters are then applied to build an image pyramid for the multi-scale analysis. The blob filter works based on the image gradient magnitude in the image pyramid. A blob image is then formed using the outputs of the blob filter in the image pyramid. In the final step, the blob image is binarized and the void candidate region is obtained by region growing. For each void candidate region, an outer rectangle is set

and the image contrast within the rectangle is enhanced again. Finally, the voids are determined by their characteristics.

The main contributions of this paper are as follows:

1. Since the proposed blob filter uses the image local gradient magnitude, it is invariant to image brightness and can overcome the random void position and component interference problems.

2. By employing a multi-scale analysis, the proposed blob filter is invariant to void size.

2 Pre-processing

Since there is noise in the BGA image, the input image is first smoothed using the Gaussian function. To efficiently detect the voids in the BGA ball, the balls in the BGA image need to be located for fast and precise detection. In this study, the balls are located using the image binarization and region growing method. As discussed earlier, BGA images have different brightnesses, which makes it difficult to use a fixed threshold to precisely separate the balls. Therefore, the OTSU method (Otsu, 1979), which gives a proper threshold used to minimize the inside variance of the white and black pixels, is applied to the smoothed image to obtain a binary image.

Figs. 3a and 3b show an original image and the result from the OTSU method. The BGA balls are well separated from the background in Fig. 3b. The black region in Fig. 3b is then determined through region growing. Finally, the average area of all the black regions is calculated and used to remove the

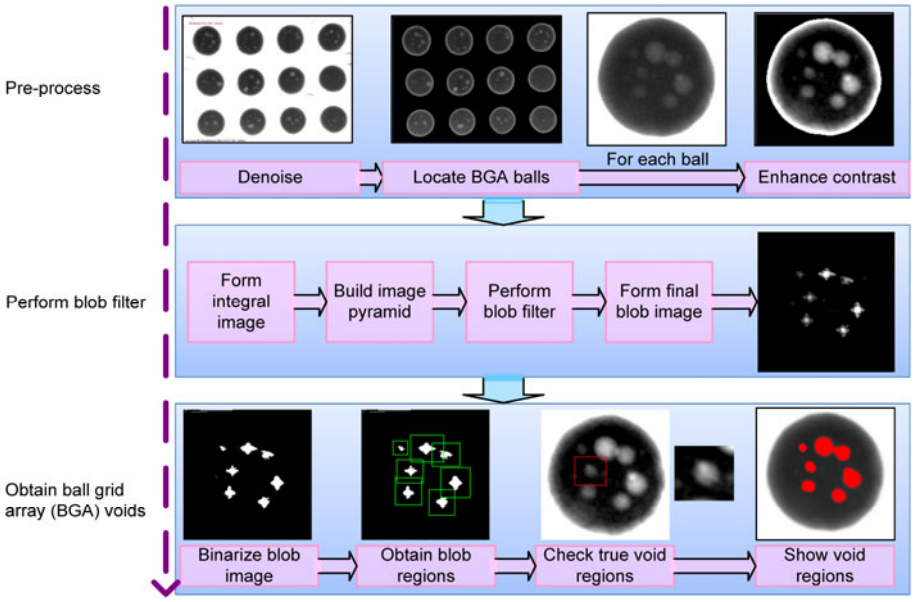


Fig. 2 Automatic void detection system

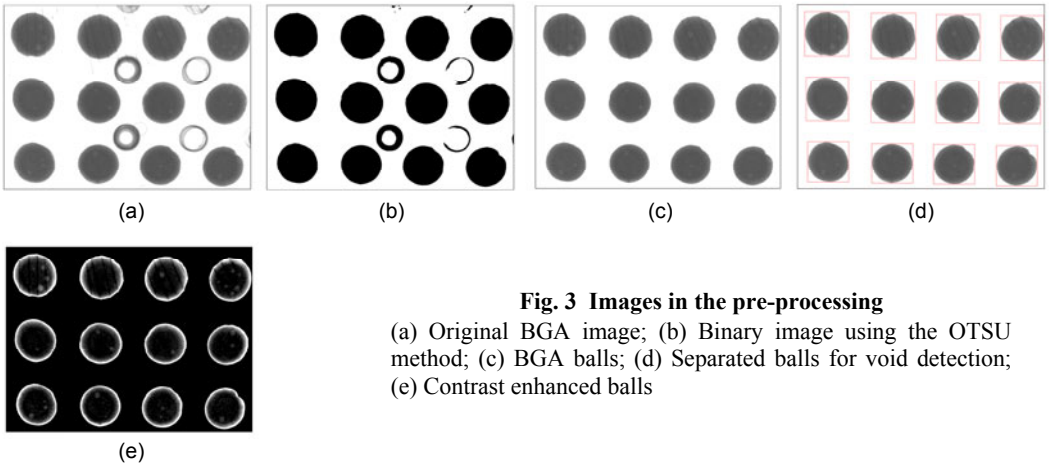


Fig. 3 Images in the pre-processing

(a) Original BGA image; (b) Binary image using the OTSU method; (c) BGA balls; (d) Separated balls for void detection; (e) Contrast enhanced balls

regions whose sizes are less than 70% of the average area (Fig. 3c). For each ball, a minimum enclosing rectangle is set to extract the separated balls from the BGA image for the ensuing processing (Fig. 3d). Since some of the voids have very poor contrast, the histogram stretch method (Seul *et al.*, 2008) is applied to enhance the contrast of each ball (Fig. 3e).

3 The proposed blob filter

As shown in Fig. 3a, the voids in the BGA ball possess blob-like regions. They have the following characteristics:

1. Its shape is blob-like.
2. It is brighter than its background.
3. The voids have different sizes.

Since the brightness of the ball is not uniform (Fig. 1c), it is not proper to set a fixed threshold in separating the void regions from the ball. Therefore, we propose a blob filter to detect the blob-like regions based on the local image gradient magnitude. We apply the average grey value of a rectangular region to approximately represent a blob region. The average grey value of a rectangular region of the proper size can be applied to evaluate the brightness of a blob. As shown in Fig. 4a, if the center region (0) is a blob, then it is brighter than its neighboring regions (1, 2, …, 8).

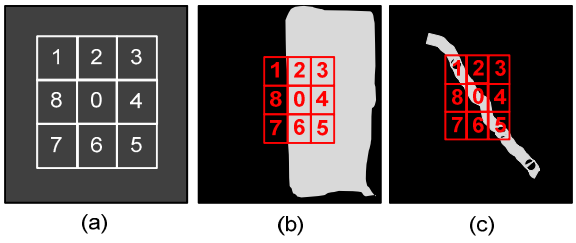


Fig. 4 Blob, edge, and line patterns

(a) The blob region and its neighbors; (b) The image edge case example; (c) The line region example

Since a blob is brighter than its background, one way to detect the blob region is to compare its brightness to its neighbors'. In this study, we use the brightness differences between a blob region and its neighbors for blob enhancement. Therefore, the gradient magnitudes based on the image local regions are defined with respect to four directions: horizontal, vertical, and two diagonal directions. They are denoted as

$$g_{\mathrm{h}} = r(0) - \max[r(4), r(8)], \tag{1}$$

$$g_{\mathrm{v}} = r(0) - \max[r(2), r(6)], \tag{2}$$

$$g_{\mathrm{ld}} = r(0) - \max[r(1), r(5)], \tag{3}$$

$$g_{\mathrm{rd}} = r(0) - \max[r(3), r(7)], \tag{4}$$

where g_{h}, g_{v}, g_{ld}, and g_{rd} represent the gradient magnitudes in the horizontal, vertical, left diagonal, and right diagonal directions, respectively, and $r(i)$ denotes the average grey value of the local region i. For a blob region, it is obvious that g_{h}, g_{v}, g_{ld}, and g_{rd} are all greater than 0. The brighter is the blob, the greater will be the values of g_{h}, g_{v}, g_{ld}, and g_{rd}. In Eqs. (1)–(4), the reason that we define the gradient magnitudes as the grey value differences between the blob region and the maximum regions in the four directions (regions 4 and 8, 2 and 6, 1 and 5, and 3 and 7) is to avoid a response to the image edge. As shown in Fig. 4b, g_{h}, g_{v}, g_{ld}, and g_{rd} will generate small values.

The output of the proposed blob filter is defined as

$$f(x,y) = (g_{\mathrm{h}} + g_{\mathrm{v}})L_1 + (g_{\mathrm{ld}} + g_{\mathrm{rd}})L_2, \tag{5}$$

where $g_{\mathrm{h}}+g_{\mathrm{v}}$ and $g_{\mathrm{ld}}+g_{\mathrm{rd}}$ are the gradient magnitude information of a local region. L_1 and L_2 are the likelihood of a blob, defined as

$$L_1 = \frac{\min(|g_{\mathrm{h}}|, |g_{\mathrm{v}}|)}{\max(|g_{\mathrm{h}}|, |g_{\mathrm{v}}|)}, \tag{6}$$

$$L_2 = \frac{\min(|g_{\mathrm{ld}}|, |g_{\mathrm{rd}}|)}{\max(|g_{\mathrm{ld}}|, |g_{\mathrm{rd}}|)}. \tag{7}$$

For a blob region, L_1 and L_2 are closely equal to 1, whereas for a line, one of the values is closely equal to 0. Fig. 4c is an example of a line region. In this case, L_2 is closely equal to 0.

To detect the different sizes of voids, multi-scale analysis (Liu *et al.*, 2010) is applied to the blob filter. Hence, different sized average box filters are employed to set up an image pyramid to detect the different sizes of voids. Suppose that the minimal size of a void is $S_{\min}\times S_{\min}$ and the maximal size of a void is $S_{\max}\times S_{\max}$. The number of average box filters can then be determined as

$$N = \frac{S_{\max} - S_{\min}}{2}. \tag{8}$$

The size of the ith average box filter is $S_i \times S_i$ ($0 \le i < N$), where

$$S_i = S_{\min} + 2i. \tag{9}$$

Applying the average box filter to an input image is a smoothing process, in which the average grey value of the pixels inside a box region is calculated. Therefore, the integral image technique (Viola and Jones, 2001) is applied to speed up the computation process when a mass of region based summation is required (Bay *et al.*, 2008; Peng *et al.*, 2010a; 2010b). The value of a point $I(x, y)$ in the integral image is defined as the summation of the left-up pixel values of x, y:

$$I(x,y) = \sum_{x' \le x, y' \le y} i(x', y'), \tag{10}$$

where $i(x', y')$ is the pixel value in the original image at (x', y'). Therefore, only three simple additions are needed for calculating the summation of a rectangular region in the image. Fig. 5 shows an example of computing the summation of region *ABDC*.

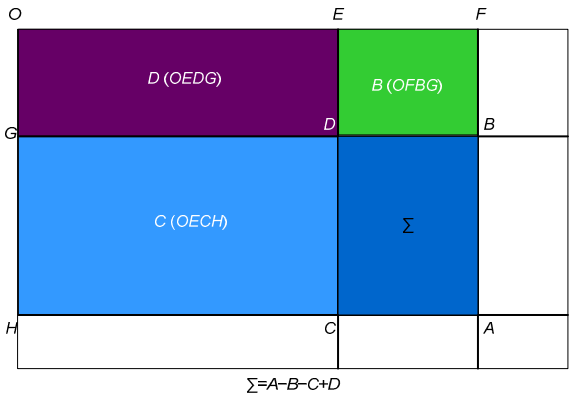


Fig. 5 The process of calculating a summation of a rectangular region by using the integral image technique

The process of generating the final blob image using the proposed method is shown in Fig. 6. The input image is first transformed into an integral image. Different sizes of average box filters are then performed on the integral image, which generates an image pyramid including the different smoothed images. For each smoothed image, the local image gradient magnitude is calculated and a blob image is generated using Eq. (5). Finally, the final blob image is generated by selecting the maximal points of the blob images (blob images 1, 2, …, N). The scale information (S), which corresponds to the box size of the maximal point in the blob image, is also recorded for each point in the final blob image. Figs. 7a, 7b, and 7c are the original input image, contrast enhanced image, and final blob image, respectively.

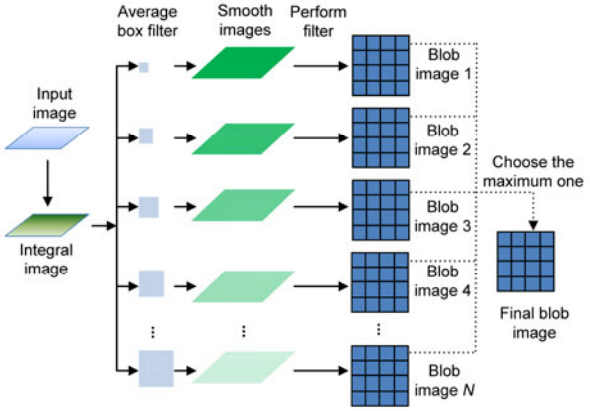


Fig. 6 The proposed blob filter process

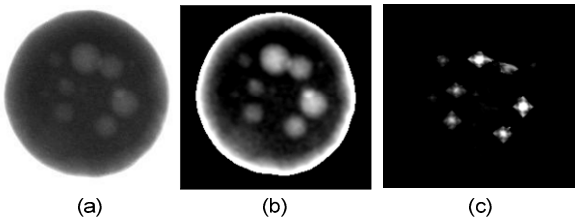


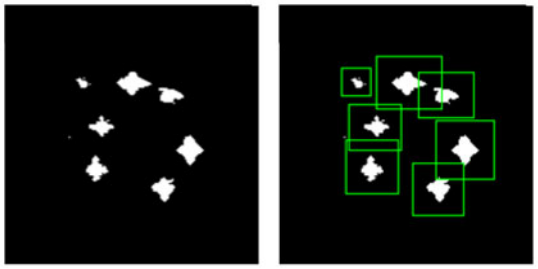
Fig. 7 The proposed blob filter results
(a) Original input image; (b) Contrast enhanced image; (c) Final blob image

4 Void detection

After generating the final blob image, a threshold is set to obtain a binary image. Region growing is applied again to obtain the white regions and their minimal enclosing rectangles (MERs). An outer rectangle that relates to the MER and the scale information (S) is then defined as

$$\begin{cases} \mathrm{Top}_{\mathrm{OR}} = \mathrm{Top}_{\mathrm{MER}} - S, \\ \mathrm{Bottom}_{\mathrm{OR}} = \mathrm{Bottom}_{\mathrm{MER}} + S, \\ \mathrm{Left}_{\mathrm{OR}} = \mathrm{Left}_{\mathrm{MER}} - S, \\ \mathrm{Right}_{\mathrm{OR}} = \mathrm{Right}_{\mathrm{MER}} + S, \end{cases} \tag{11}$$

where Top_{OR}, $\text{Bottom}_{\text{OR}}$, Left_{OR}, and Right_{OR} are the four endpoints of the outer rectangle. Top_{MER}, $\text{Bottom}_{\text{MER}}$, Left_{MER}, and $\text{Right}_{\text{MER}}$ are the four endpoints of the MER. Figs. 8a and 8b are the binary image of Fig. 7c and its corresponding outer rectangle regions, respectively.



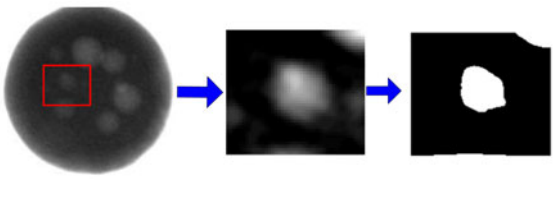
(a) (b)

Fig. 8 Binary image (a) and the corresponding outer rectangles (b)

To locate the voids, the defined outer rectangle is used to capture a local region from the original image (Fig. 9a); the contrast of the region is enhanced using the histogram stretch method (Fig. 9b). The contrast enhanced region is then binarized with a threshold (Fig. 9c). The morphological method (closing operation) with a structure size of five pixels is then applied to connect some of the open regions. Finally, the void candidate region is obtained through region growing and the true voids are determined according to the roundness of the region. The roundness is defined as

$$R = P^2/(4\pi A), \tag{12}$$

where P and A are the perimeter and area of the region, respectively. If the contour of the region is a perfect circle, R will be close to 1. In the final step, the void ratio for each ball is calculated as



(a) (b) (c)

Fig. 9 Void candidate location process

(a) A BGA void with its outer rectangle; (b) Magnified and contrast enhanced void region of (a); (c) Binary result of (b)

$$V_{\text{R}} = \sum_{i=0}^{N-1} \text{VS}_i \Big/ \text{BS}, \tag{13}$$

where N is the number of voids in a ball, VS_i is the size of the ith void, and BS is the size of the ball.

5 Experimental results

5.1 Data collection

To evaluate the performance of the proposed method, 355 BGA images were captured by a computed tomography (CT) machine provided by Tech-Valley Ltd. Co. For the CT machine, the size of the focus spot is 5 μm, which enables the detection of voids with a minimum size of 10 μm. The scanning area of the machine is 300 mm×300 mm and the maximum magnification is 150×. The diameters of the balls for these images are 0.5 mm and the minimum interval of the balls is 0.1 mm. Therefore, the maximum number of assembled balls that the machine can inspect at one time is 250 000.

The captured images were used to form five image datasets used for the evaluation. All of the images were collected by setting the current of the machine to 50 μA and stored in the 8-bit BMP format with a size of 1024×768. Table 1 shows the image dataset information. Fig. 10 is an example of the image samples. As we can see, the brightness of the image varies with respect to the voltages. The higher the voltage used, the brighter the image. The experimental environment is shown in Table 2.

5.2 Performance evaluation

The performance of the proposed method is evaluated by doing comparisons using Said *et al.* (2010)'s algorithm according to the detection accuracy and the false ratio, which are defined as

$$\text{Accuracy} = \frac{\text{TP}}{\text{VN}} \times 100\%, \tag{14}$$

$$\text{False ratio} = \frac{\text{FP}}{\text{VN}} \times 100\%, \tag{15}$$

where TP (true positive) denotes the pixel number of voids that are correctly detected, FP (false positive) is the pixel number of normal regions that are incorrectly

Table 1 Image dataset information

Dataset No.	Voltage (kV)	Number of images	Number of balls	Number of balls interfered by lines	Number of voids	IR (%)
1	95	57	657	90	1438	13.699
2	100	68	446	92	1331	20.628
3	105	160	1144	363	3696	31.731
4	110	39	447	2	2482	0.447
5	120	31	331	31	1306	9.366

IR: ratio between the number of line-interrupted balls and the total number of balls

Table 2 Experimental environment

Item	Value/Description
CPU	Dual Core™ 2.2 GHz
Operation system	Windows XP
Memory	2 GB
Programming tool	VC++ 6.0
Hard disk	200 GB

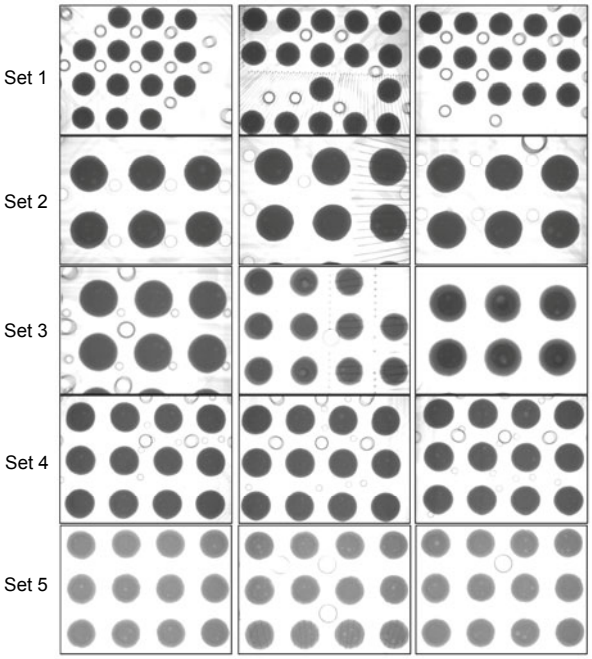


Fig. 10 Dataset image samples

detected as void pixels, and VN is the pixel number of true voids. Note that VN was obtained manually by two experienced radiologists from Techvalley Ltd. Co. and regarded as a gold standard. The void regions were first captured by the radiologists separately (Figs. 11b and 11c). The overlapped region of Figs. 11b and 11c is then considered as the true void (Fig. 11d). Suppose that the detected voids are $S_d=\{P(x_{d1}, y_{d1}), P(x_{d2}, y_{d2}), ..., P(x_{dn}, y_{dn})\}$ and the true voids are $S_t=\{P(x_{t1}, y_{t1}), P(x_{t2}, y_{t2}), ..., P(x_{tm}, y_{tm})\}$. $P(x_{di}, y_{di})$ and $P(x_{tj}, y_{tj})$ (i=1, 2, ..., n; j=1, 2, ..., m) are the pixels within the detected void regions and the true void regions, respectively. (x_{di}, y_{di}) and (x_{tj}, y_{tj}) denote the pixel position in the image. As a result, the true positive is the intersection of S_d and S_t ($S_d \cap S_t$, as shown in Fig. 12a) and the false positive is the complement of S_t (S_t') intersecting with S_d ($S_t' \cap S_d$, as shown in Fig. 12b). As shown in Eqs. (14) and (15), a good automatic method will achieve a high accuracy and a low false ratio.

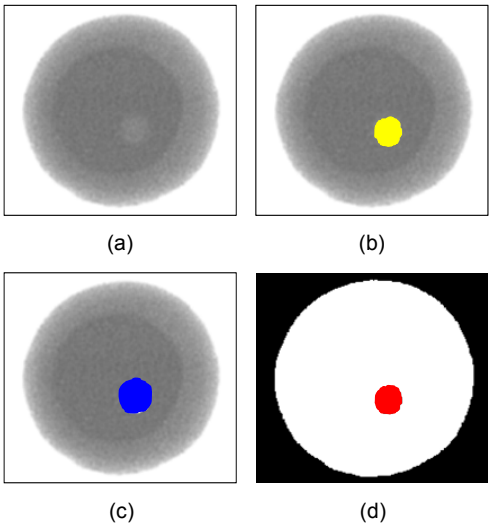


Fig. 11 The process of obtaining the true voids
(a) Original image; (b) Image captured manually by the 1st radiologist; (c) Image captured manually by the 2nd radiologist; (d) Overlapped region of (b) and (c)

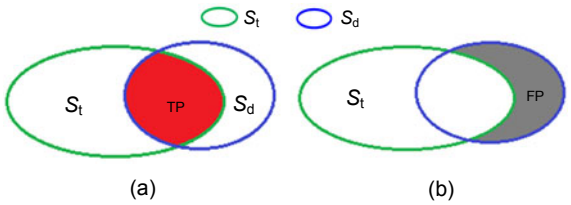


Fig. 12 True positive (TP) (a) and false positive (FP) (b)

Due to the image noise, the detected regions whose sizes are greater than 3×3 pixels are considered as void candidate regions. For the datasets of Sets 1 to 5, the ball diameters are ranged from 150 to 230 pixels, and the maximum void size in these datasets is 12.1% of the ball size. Therefore, in the following experiments regarding the proposed method, S_{min} and S_{max} are set to 3 and 33 (for Sets 1, 2, 4, and 5), respectively. For Set 3, S_{max} is set to 81 since the maximum void size of this dataset is up to 12.1% of the ball size.

To show the detection ability of the proposed method with respect to the void sizes, Set 3 was used to test the detection accuracy of various void sizes. Fig. 13 shows the detection accuracy of the proposed method with respect to the void size. In Fig. 13, the accuracy of '5×5' denotes the detection accuracy of the void sizes ranging from '3×3' to '5×5'. Similarly, the accuracy of '9×9' denotes the detection accuracy of the void sizes ranging from '5×5' to '9×9'. As shown in Fig. 13, the detection accuracy of the void sizes less than '5×5' is below 80%. This is because of the serious image noise, which makes it difficult to distinguish small voids and noise. For those void sizes greater than '17×17', the proposed method shows good performance—the detection accuracy is greater than 90%.

The threshold used to binarize the contrast enhanced region (Section 4) was set to 80; R (Eq. (12)) was set to 0.6. Note that, in Said *et al.* (2010)'s algorithm, instead of using the Laplacian of Gaussian (LoG) (Canny, 1986), the Canny edge detector (Maini and Aggarwal, 2009) was employed to detect the void contour, since Maini and Aggarwal (2009) have shown that Canny's edge detection algorithm (Canny, 1986) outperformed the other operators (LoG, Robert, Prewitt, and Sobel) under almost every scenario.

Table 3 shows the detection results of the proposed method and Said *et al.* (2010)'s algorithm. The proposed method achieved much higher detection accuracy and lower false ratios than Said *et al.* (2010)'s algorithm for all of the datasets. The proposed method outperformed Said *et al.* (2010)'s algorithm by 40.69% with respect to the average detection accuracy, and by 16.91% with respect to the average false ratio. Since the images in Set 1 were captured by a low voltage (95 kV), which results in low contrast, both of the algorithms showed the lowest performance compared to the other datasets. The proposed method obtained its best detection accuracy in Set 3. In Table 3, the false ratio of Set 1 is the highest, whereas Sets 2–5 show similar false ratios. From Table 1 we can see that Sets 2 and 3 have the highest IR. Therefore, the performance of the proposed method is robust to the IR. Since Said *et al.* (2010)'s algorithm detects the voids using an edge detector, it strongly responds to the interference lines. As a result, the false ratios of Sets 2 and 3 for Said *et al.* (2010)'s algorithm are higher than those of the other datasets because of their high IRs. Fig. 14 shows the detection result achieved by applying the proposed method, which can not only mark the voids, but also show the void ratio.

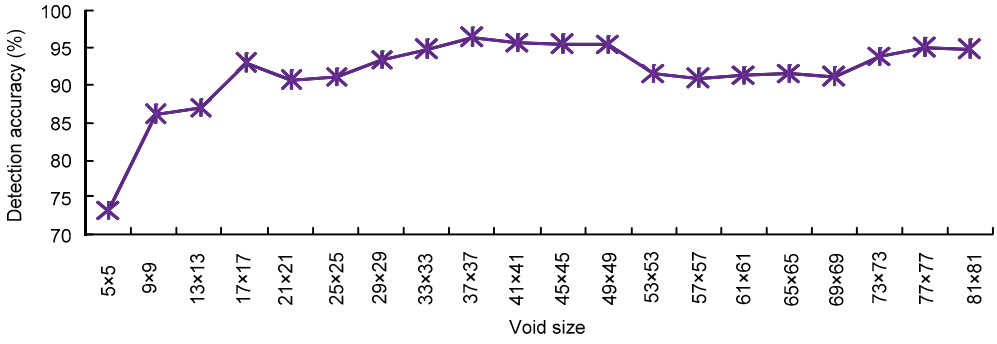


Fig. 13 Performance of the proposed method with respect to the void size

Table 3 Void detection results from the proposed method and Said *et al*. (2010)'s algorithm

Dataset	VN	TP	FP	Accuracy (%)	False ratio (%)
Proposed method					
Set 1	84 526	75 035	13 658	88.77	16.16
Set 2	454 429	406 984	38 441	89.56	8.46
Set 3	646 905	604 644	59 205	93.47	9.15
Set 4	197 764	180 852	21 963	91.45	11.11
Set 5	247 759	223 540	16 931	90.22	6.83
Total	1 631 383	1 491 055	150 198		
Average				91.40	9.21
Said *et al*. (2010)'s algorithm					
Set 1	84 526	36 948	9318	43.71	11.02
Set 2	454 429	225 148	168 023	49.55	36.97
Set 3	646 905	334 295	200 654	51.68	31.02
Set 4	197 764	101 740	19 477	51.45	9.85
Set 5	247 759	129 200	28 713	52.15	11.59
Total	1 631 383	827 331	426 185		
Average				50.71	26.12

VN: pixel number of true voids; TP (true positive): pixel number of voids that are correctly detected; FP (false positive): pixel number of normal regions that are incorrectly detected as void pixels

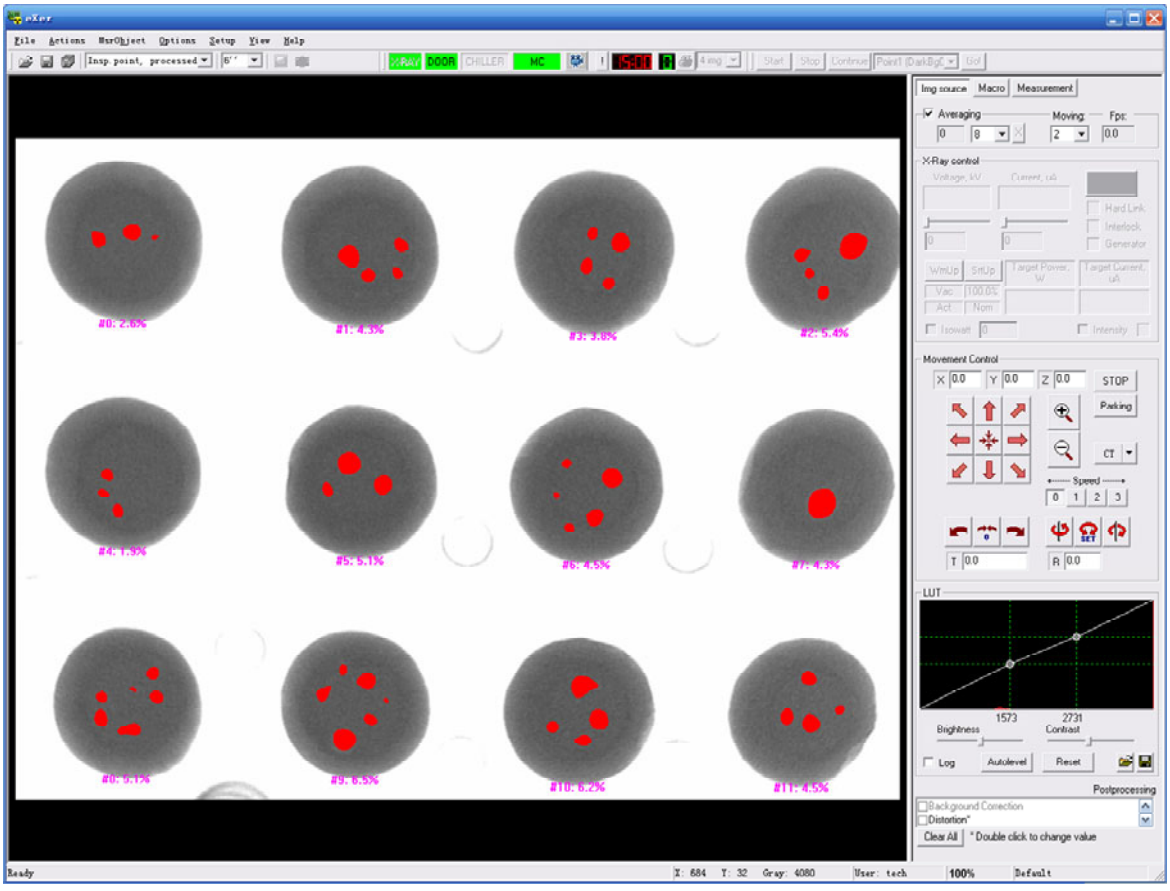


Fig. 14 Detection result example

6 Conclusions

Automatic void detection is critical in the manufacturing of BGA soldered/assembled boards. In this paper, an automatic void detection system is presented based on the proposed blob filter. First, the system locates the BGA balls using an automatic segmentation method. A novel blob filter is then used to enhance the BGA voids by taking advantage of the image local gradient magnitude. Since the output of the blob filter is defined by a comparison to the average grey value of the neighboring regions, it is not affected by the image brightness, void position, or component interference. To overcome the difficulty found from various sizes of voids, different sized average box filters are employed for a multi-scale analysis. Due to the advantages of the proposed blob filter, the void detection system can achieve high detection accuracy and maintain a low false ratio when component interference, non-uniform brightness, etc., are present. In our future work, the proposed method will be tested in other applications requiring blob-like region detection, such as stain detection in wafer images and nodule detection in CT lung images.

References

Bay, H., Ess, A., Tuytelaars, T., Gool, L.V., 2008. Speeded-up robust features (SURF). *Comput. Vis. Image Understand.*, **110**(3):346-359. [doi:10.1016/j.cviu.2007.09.014]

Canny, J., 1986. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, **8**(6):679-698. [doi:10.1109/TPAMI.1986.4767851]

Lin, W.C., 2007. The Void-Free Reflow Soldering of BGA with Vacuum. 8th Int. Conf. on Electronic Packaging Technology, p.1-5. [doi:10.1109/ICEPT.2007.4441462]

Liu, Y.M., Ye, L.B., Zheng, P.Y., Shi, X.R., Hu, B., Liang, J., 2010. Multiscale classification and its application to process monitoring. *J. Zhejiang Univ.-Sci. C (Comput. & Electron.)*, **11**(6):425-434. [doi:10.1631/jzus.C0910430]

Maini, R., Aggarwal, H., 2009. Study and comparison of various image edge detection techniques. *Int. J. Image Process.*, **3**(1):1-11.

Otsu, N., 1979. A threshold selection method from gray-level histogram. *IEEE Trans. Syst. Man Cybern.*, **9**(1):62-66. [doi:10.1109/TSMC.1979.4310076]

Peng, S.H., Muzzammil, K., Kim, D.H., 2010a. Robust Feature Detection Based on Local Variation for Image Retrieval. 17th IEEE Int. Conf. on Image Processing, p.1033-1036. [doi:10.1109/ICIP.2010.5652973]

Peng, S.H., Kim, D.H., Lee, S.L., Lim, M.K., 2010b. Texture feature extraction based on a uniformity estimation method for local brightness and structure in chest CT images. *Comput. Biol. Med.*, **40**(11-12):931-942. [doi:10.1016/j.compbiomed.2010.10.005]

Rooks, S.M., Benhabib, B., Smith, K.C., 1995. Development of an inspection process for ball-grid-array technology using scanned-beam X-ray laminography. *IEEE Trans. Comp. Pack. Manuf. Technol. Part A*, **18**(4):851-861. [doi:10.1109/95.477473]

Said, A.F., Bennett, B.L., Karam, L.J., Pettinato, J., 2010. Robust Automatic Void Detection in Solder Balls. IEEE Int. Conf. on Acoustics Speech and Signal Processing, p.1650-1653. [doi:10.1109/ICASSP.2010.5495524]

Sa-nguannam, A., Srinonchat, J., 2008. Analysis Ball Grid Array Defects by Using New Image Technique. 9th Int. Conf. on Signal Processing, p.785-788. [doi:10.1109/ICOSP.2008.4697247]

Sankaran, V., Kalukin, A.R., Kraft, R.P., 1998. Improvements to X-ray laminography for automated inspection of solder joints. *IEEE Trans. Comp. Pack. Manuf. Technol. Part C*, **21**(2):148-154. [doi:10.1109/3476.681394]

Seul, M., O'Gorman, L., Sammon, M.J., 2008. Practical Algorithms for Image Analysis: Descriptions, Examples, and Code. Cambridge University Press, UK.

Sumimoto, T., Maruyama, T., Azuma, Y., Goto, S., Mondo, M., Furukawa, N., Okada, S., 2002. Detection of Defects at BGA Solder Joints by Using X-Ray Imaging. IEEE Int. Conf. on Industrial Technology, p.238-241. [doi:10.1109/ICIT.2002.1189898]

Teramoto, A., Murakoshi, T., Tsuzaka, M., Fujita, H., 2007. Automated solder inspection technique for BGA-mounted substrates by means of oblique computed tomography. *IEEE Trans. Electr. Pack. Manuf.*, **30**(4):285-292. [doi:10.1109/TEPM.2007.907574]

Viola, P., Jones, M., 2001. Rapid Object Detection Using a Boost Cascade of Simple Features. Int. Conf. on Computer Vision and Pattern Recognition, **1**:511-518. [doi:10.1109/CVPR.2001.990517]

Xia, N.J., Cao, Q.X., Fu, Z., Lee, J., 2004. A Machine Vision System of Ball Grid Array Inspection on RT-Linux OS. Int. Conf. on the Business of Electronic Product Reliability and Liability, p.81-85. [doi:10.1109/BEPRL.2004.1308154]