ABSTRACT

Voids in solder balls can cause board failures. The detection and assessment of voids in solder balls can help in reducing board yield issues caused by incorrect scrapping and rework. X-ray imaging machines make voids visible to the operator for manual inspection. Some existing x-ray inspection systems have void detection algorithms that require the use of intensive manual tuning operations that are time consuming, and inaccurate due to the inability to examine balls overshadowed with other components. In this paper, a robust automatic void detection algorithm is proposed. The proposed method is able to detect voids with different sizes inside the solder balls, including the ones that are overshadowed by board components and under different brightness conditions. Results show that the proposed method achieves a correlation squared in the range of 91% to 97% with ground truth data from a 3D x-ray scan. The proposed algorithm is fully automated and benefits the manufacturing process by reducing operator effort and by providing a cost effective solution to improve output quality.

Index Terms- Voids Detection, BGA, segmentation, features extraction.

1. INTRODUCTION

The detection of voids in solder balls has become a critical issue for surface mount production lines. Voids in solder balls are difficult to detect using manual inspection alone. One current solution to this problem involves the set up and programming of a 2D x-ray machine to image either semiconductor devices or printed circuit boards. The x-ray makes the voids inside the solder balls visible to the human eye as shown in Fig. 1. Some of the existing x-ray inspection systems include void detection algorithms. However, the use of the void detection algorithms in these systems requires intensive preprocessing steps and manual fine tuning. Most of the existing void detection systems use global thresholding to segment the balls and voids inside the 2D x-ray image. This is typically done by letting the user set one threshold value to segment the balls in the image. A second threshold value is then set to segment the voids inside the segmented balls. These threshold values are typically determined by the operator using a trial and error method. However, this method is invalidated due to inconsistency of image brightness. Many voids are missed because a single threshold does not give the capability to differentiate between actual voids and false voids. Moreover, some of those algorithms ignore or fail to inspect the balls that are occluded by overshadowing components as shown in Fig. 1.

Another way to detect voids in solder balls is to use existing image enhancement software to examine the x-ray images manually after enhancing the image brightness and contrast. The operator must then detect and measure each observed void. The process is very tedious, time consuming and highly variable.

In this paper, we present a new 2D x-ray image void detection algorithm. The proposed method is fully automated and is capable of detecting voids with different sizes in all solder balls including those that are occluded by overshadowing components. The algorithm also operates under different brightness conditions. The proposed method was successfully tested and results show a high correlation with ground-truth data obtained from 3D x-ray and experienced operators.

This paper is organized as follows. Section 2 presents the proposed automatic void detection algorithm. Performance results and comparison with existing schemes are presented in Section 3. A conclusion is given in Section 4.

2. PROPOSED VOID DETECTION METHOD

There are many challenges that are observed in the acquired 2D x-ray images. These can be summarized as follows: (i) poor image contrast at some balls making void detection difficult even by the human eye, (ii) interference of other components on the unit or board, such as capacitors, that occlude solder balls as shown in Fig. 1, (iii) interference from vias ((plated through hole) that have similar characteristics to the actual voids (see Fig.1(b)), and (iv) voids have a circular shape in most cases, but this is not
consistent due to overlapping and shadowing caused by the use of a single two dimensional image.

Most of the existing void detection systems do not provide a robust solution to tackle all the above challenges in x-ray images. The main goal of the proposed algorithm is to provide a consistent, highly accurate, and robust automated void detection capability. A block diagram of the proposed void detection scheme is shown in Fig. 2. The proposed method consists of several components including individual solder ball segmentation, the extraction of occluded solder balls, and the extraction of candidate void regions inside each segmented solder ball, and the classification of the extracted candidate in order to exclude all non-void areas. Details about the individual solder ball segmentation and the candidate void detection and classification are given in Sections 2.1 and 2.2 below.

2.1. Individual Solder Ball Segmentation

The 2D x-ray images do not exhibit uniform lighting and, thus, using a fixed threshold for solder ball segmentation fails to properly segment the desired balls. In this work, an automatic thresholding method based on histogram analysis is used to segment the solder balls. As shown in Fig. 3, the histogram of the considered 2D x-ray images contains two distinct regions (clusters) represented by two distinct peaks: the solder ball and background regions. The two cluster regions can be represented by a mixture of two Gaussian distributions with two different means and variance parameters. In order to segment the two regions, an automatic threshold is selected midway between the mean values of the two Gaussian distribution functions. First, in order to compute the segmentation threshold, the means of the Gaussian distributions are estimated. For this purpose, we adopt an iterative procedure to calculate the mean values that are then used to compute the segmentation threshold. The steps of the procedure for computing the mean values of the two Gaussian distributions and the segmentation threshold can be summarized as follows:

- Step 1: Compute the probability \( p_i \) from the histogram distribution of the input image \( h(i) \), where \( i \) denotes the 8-bit image intensity, as:
  \[
  p_i = \frac{h(i)}{\sum_{j=0}^{255} h(j)}, \quad i = 0, 1, \ldots, 255. \tag{1}
  \]

- Step 2: At the \( n^{th} \) iteration, the updated mean values of the two Gaussian functions, \( u_1^{n+1} \) and \( u_2^{n+1} \), are given by
  \[
  u_1^{n+1} = \frac{\sum_{k=0}^{T_n} k p_k}{\sum_{k=0}^{255} p_k}, \quad u_2^{n+1} = \frac{\sum_{k=T_n+1}^{255} k p_k}{\sum_{k=T_n+1}^{255} p_k} \tag{2}
  \]

  where \( T_n \) denotes the threshold at the current \( n^{th} \) iteration. For \( n = 0 \), the initial threshold \( T^0 \) is selected to be the mean of the input image.

- Step 3: The new updated threshold is obtained from the mean values \( u_1^{n+1} \) and \( u_2^{n+1} \) as
  \[
  T^{n+1} = \frac{u_1^{n+1} + u_2^{n+1}}{2} \tag{3}
  \]

Steps 2 and 3 of the above process are repeated until the threshold value \( T^{n+1} \) is no longer changing. This procedure converges in an average of 4 to 6 iterations. Fig. 4(b) shows the segmentation results for the image given in Fig. 4(a) after using the above procedure for solder ball segmentation. Only complete balls need to be processed to achieve an accurate computation of voiding area and total voiding percentages. This necessitates the exclusion of incomplete balls that lie on the image border. Fig. 4(c) shows the area of each segmented region in Fig. 4(b). The median area value in Fig. 4(c) represents the approximate solder ball area \( A_b \). The small values (below 0.6 of the median area) in Fig. 4(c) correspond to areas of incomplete balls and undesired background regions, while the high values (above 1.2 of median area) correspond to the areas containing the occluded balls and connected components. The regions corresponding to the segmented complete and non-connected solder balls are obtained by keeping only the segmented regions with an area between 0.6 and 1.2 of the median area.

In order to extract the occluded balls, the proposed scheme exploits the fact that the solder balls are aligned along different directions, including \( 0^\circ, \pm 45^\circ, \pm 90^\circ \), as shown in Fig. 4(d). For this purpose, the centroids of the already segmented non-occluded balls are determined and are used in computing the angles of the lines connecting each of these centroids to pixels in the regions containing the occluded balls (regions with areas above 1.2 of the median area). Each pixel having at least two lines with angles within 2 degrees of the values \( 0^\circ, \pm 45^\circ, \pm 90^\circ \) is taken to be a candidate pixel belonging to one of the occluded balls. The set of all candidate pixels results in \( M \).
isolated connected regions whose centroids are computed to
give the centroids of candidate occluded solder balls. The
candidate occluded balls can then be extracted by drawing,
for each of these centroids, a circle centered at that centroid
and with a radius \( R = \sqrt{\frac{A_r}{\pi}} \), where \( A_r \) is the previously
determined solder ball area. Furthermore, those candidate
occluded balls with a mean intensity that deviates
significantly from the average intensity of non-occluded
balls are considered as outliers and are eliminated. Fig. 4(e)
represents the final segmentation mask including the balls
from the occluded regions. Using this mask, all the
individual complete solder balls shown in Fig. 4(a) are
segmented successfully as shown in Fig. 4(f).

2.2. Void Detection

In order to locate voids inside the individually segmented
ball, each segmented ball is adaptively processed separately.
In the proposed method, for each segmented solder ball,
edge detection is first performed to locate the contours of
the voids inside the considered solder ball. There are many
edge detection methods [1-6] that can be applied here.
However, in our implementation, a simple Laplacian of
Gaussian (LoG) edge detection method was used [5-7].
The edge detection can however result in some open contours.
Closed contours are obtained by applying a morphological
dilation operation [8, 9] using a structuring element of size 1.

Closed contours are obtained by applying a morphological
dilation operation [8, 9] using a structuring element of size 1.

In order to locate voids inside the individually segmented
ball, each segmented ball is adaptively processed separately.
In the proposed method, for each segmented solder ball,
edge detection is first performed to locate the contours of
the voids inside the considered solder ball. There are many
edge detection methods [1-6] that can be applied here.
However, in our implementation, a simple Laplacian of
Gaussian (LoG) edge detection method was used [5-7].
The edge detection can however result in some open contours.
Closed contours are obtained by applying a morphological
dilation operation [8, 9] using a structuring element of size 1.

Each area inside a closed contour is then extracted by using
an image labeling procedure [8]. For example, region \( R_l \)
with a label \( l \) can be extracted from an image \( f(x,y) \) where
\( f(x,y) \) is a labeled image that contains integer numbers
representing the labels for the segmented balls as follows:

\[
R_l = \{(x,y), \text{ where } f(x,y) = l\}
\]

where \( x \) and \( y \) represent the pixel locations inside image
\( f(x,y) \) and \( l \) represents the label number of each segmented
region \( l = 1, 2, \ldots, N \).

The final step in void detection is to filter out the regions
that represent false voids and keep only the regions that
represent the actual voids. This process is performed by
classifying the segmented candidate regions as void or non-
void based on specific feature parameters. The selected
feature parameters should be robust to artifacts such as vias
and other interferences in the image. The classification
process is performed for each region \( R_l \) inside the
segmented solder ball using mainly two feature parameters:
compactness factor and contrast. The compactness factor
\( CF_l \) for region \( R_l \) is defined by

\[
CF_l = \frac{p_l^2}{4\pi A_l}
\]

where \( p_l \) and \( A_l \) represent the perimeter and the area of
region \( R_l \), respectively. The compactness factor \( CF_l \) is
important for eliminating irregular shapes corresponding to
false voids. \( CF_l \) is close to 1 for a circular shape and greater
than 1 for other shapes. Each candidate void \( l \) is also
assessed and classified based on the contrast between inside
and outside of the segmented region \( R_l \). It follows that only
regions \( R_l \) with a mean intensity greater than or equal to a
value \( T_l \) are kept as candidate voids. \( T_l \) is adaptively
computed for each region \( R_l \) based on the gray level values
\( V_l \) from pixels in the vicinity surrounding the contour
outside \( R_l \) as follows:

\[
T_l = \text{mean} \left( V_l \geq 0.8 \times \text{median} \left( V_l \right) \right)
\]

In (6), the median intensity value \( \text{median}(V_l) \) helps in
eliminating false voids with black borders in their vicinity
(e.g., black borders due to vias).

In order to account for the fact that some true voids
contain small gaps causing small shape irregularities and
resulting in relatively higher compaction factors, a
morphological dilation operation is applied separately to
each candidate region to close these small gaps. A circular
structuring element with a radius of 2 pixels is used for the
dilation operation, except when the dilation operation causes
two nearby regions to be merged. In this latter case, a
structuring element of radius 1 pixel is used. After the
dilation operation, only regions with a compactness factor
less than or equal to 1.15 are classified as void regions. Fig.
4(g) shows all possible visible voids inside the segmented
solder ball, while Fig. 4(h) represents the void detection
result of Fig. 4(g) using the proposed method.

The proposed void detection scheme is robust to various
artifacts inside the solder ball such as inconsistent lighting
and interference from vias and other components.

3. PERFORMANCE RESULTS

The proposed method was applied to two image sets, with a
total of 510 images, corresponding, respectively, to the Intel
Diamondville and Penryn product lines. The results of the
proposed algorithm were compared to the existing latest 2D
x-ray void detection algorithms that are provided as part of
the x-ray imaging machine. Comparison and performance
results are shown in Fig. 5. From Fig. 5, it can be seen that,
while existing 2D x-ray void detection algorithms are not
able to detect small voids and voids in solder balls that are
occluded or connected by board components, the proposed
algorithm is able to successfully detect all voids including
small ones and the ones present in occluded balls. In
addition, the proposed algorithm is able to more accurately
detect voids and their areas while avoiding false void-like
structures that are caused by void-like shadows in the visible
viases.

In order to further assess the performance of the proposed
method, the obtained results were compared to both the
results obtained by an expert operator manually measuring
voids using an image processing program, and the visual
inspection results obtained by a 3D x-ray scan that is
considered ground truth data. The proposed method results
in a correlation squared with the ground truth data of 91% and
97% for Penryn and Diamondville, respectively, with no
significant bias. The obtained void detection results also
correlate well with the operator measurements (88% for
Penryn and 91% for Diamondville (correlation squared with
no significant bias). It should be noted that a correlation
squared value greater than or equal to 75% correspond to well correlated data.

4. CONCLUSION
A robust automatic void detection scheme is presented in order to allow automated inspection and automated manufacturing quality assessment. The proposed method is fully automated and can benefit the manufacturing process by reducing operator effort and process variability.

5. REFERENCES