Int J Adv Manuf Technol (2017) 93:55–63
DOI 10.1007/s00170-015-7638-5

ORIGINAL ARTICLE

Blob analyzation-based template matching algorithm for LED chip localization

Fuqiang Zhong[1] · Songping He[1] · Bin Li[1]

Received: 7 March 2015 / Accepted: 21 July 2015 / Published online: 18 August 2015

Abstract During the testing and sorting of LED chips, traditional methods do not exclude the polycrystalline and fragmentary LED chips from the normal chips well. The purpose of this paper is to propose a new algorithm to solve this problem. The algorithm consists of three steps. Firstly, present a simple but efficient image segmentation method to get blobs. Secondly, analyze the blobs to exclude abnormal blobs and predict the pose (position and orientation) of the potential object based on the pose of the minimum enclosing rectangle (MER) of each remained blob. Finally, according to the predicted poses, locate the LED chips precisely in the originally captured image based on gradient orientation features. Experiments show that the algorithm is not only robust to illumination variation but also can locate the LED chips and exclude the polycrystalline and fragmentary chips efficiently.

Keywords Blob · Gradient orientation · Image segmentation · Minimum enclosing rectangle

1 Introduction

LED is a kind of semiconductor light source component, regarded as the fourth generation of green lighting source [1]. Machine vision, as an advanced manufacturing technology, is widely used in LED manufacturing equipment. Inspection and sorting of LED chips are two imperative procedures in LED manufacturing process. Before LED chip inspection and sorting, it is inevitable to use machine vision to scan the LED chips to locate the LED chips precisely [2]. There are some common defects in LED chips, such as polycrystalline and fragmentary defects, shown in Fig. 1. Polycrystalline defect means a chip contains a part of another chip and is bigger than a normal chip. In addition, if the distance between two chips is too small, the chips should also be excluded in case they are polycrystalline. Fragmentary defect means a chip lacks a part and is smaller than a normal chip. Polycrystalline and fragmentary chips are mainly caused by improper cutting. The electrical and optical properties of the polycrystalline and fragmentary LED chips are very abnormal; therefore, the manufacturers hope that polycrystalline and fragmentary LED chips are excluded from the normal chips during the procedures of inspection and sorting of LED chips. However, the polycrystalline LED chips have the part of a complete chip and most of the fragmentary LED chips are similar to a complete chip very much; so the similarity between the template chip and polycrystalline or fragmentary chip is relatively high and the traditional algorithms [3–9] are universal template matching methods. They do not take the size of the object into consideration. So these methods may not distinguish polycrystalline or fragmentary chips from the normal chips well, as shown in Fig. 2. In order to make sure that the LED chips picked to Bins are normal, it is necessary to develop an algorithm to locate the LED chips on wafers and exclude the polycrystalline and fragmentary chips efficiently.

Template matching methods [10–12] are always used to detect a given object with high speed, high pose accuracy. There are three main kinds of template matching methods, namely template matching based on gray value [3], geometric features [4–7], and gradient orientation features [8–10]. The first method can get relatively high pose accuracy when the

✉ Songping He
hesongping@mail.hust.edu.cn

[1] State Key Laboratory of Digital Manufacturing Equipment and Technology, School of Mechanical Science and Engineering, HUST, Wuhan 430074, China

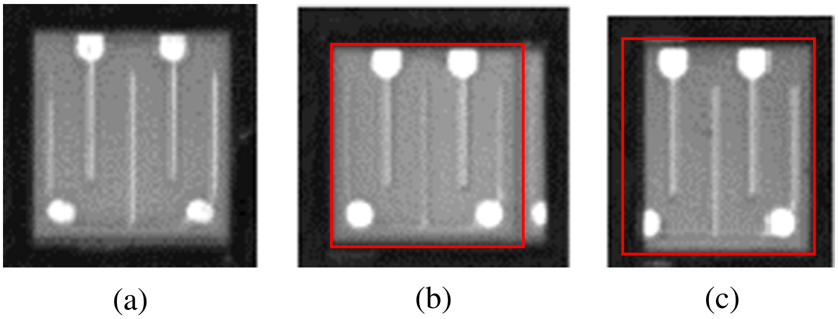

Fig. 1 A normal chip and typical defective chips. **a** Normal chip, **b** polycrystalline chip, **c** fragmentary chip. The *red rectangles* in **b** and **c** denote the normal size of the chip

image edges are fuzzy, but this method requires high illumination stability and uniformity. Template matching based on geometric features is very efficient, but it requires clear image edges and it is easily disturbed by image noise. Template matching based on gradient orientation features is robust to illumination variation and nonuniformity and can obtain high accuracy even when image edges are not clear. Therefore, it is more suitable to use gradient orientation features to locate LED chips precisely in the last step of the proposed algorithm.

Reference [13] proposed a new algorithm for object detection. Before object detection, Alwin Anbu et al. firstly made use of the motion information to obtain regions of interest in video surveillance and then used the segmentation methods to get the regions of interest further, which reduced the detection area greatly. In addition, references [14, 15] presented the similar ideas that, before object detection or matching, image segmentation, blob features, and some other feature analyzation can be used to obtain the potential regions, called regions of interest, which would reduce computation complexity greatly. In this paper, image segmentation and blob analyzation are used to exclude abnormal blobs which may represent the defective chips and to predict the poses of the chips based on the poses of the MERs of the remained blobs, before the LED chips on wafers are located precisely.

2 Proposed algorithm

Figure 3 shows the overview of the proposed algorithm.

2.1 Image segmentation

LED chips are often produced on blue membrane. When a wafer is under suitable coaxial light source and ring light source, LED chips are brighter than background, as shown in Fig. 5a, b. So it can use the difference to segment the input image to get blobs. Image segmentation is the first step of the proposed algorithm, and the quality of image segmentation determines the effectiveness of the algorithm. The difficulty in image segmentation is how to choose the threshold, especially when the illumination is nonuniform and varies. The widespread approaches for image segmentation are Otsu method, histogram-based methods, edge-based methods, model-based methods, and watershed methods. Especially the Otsu method [16], almost the most popular algorithm for image segmentation can compute the threshold for image segmentation automatically and is used well in many occasions. But if the foreground is just a small part, compared to the background, it would not segment the image well, just like histogram-based methods. For example, when there is only a

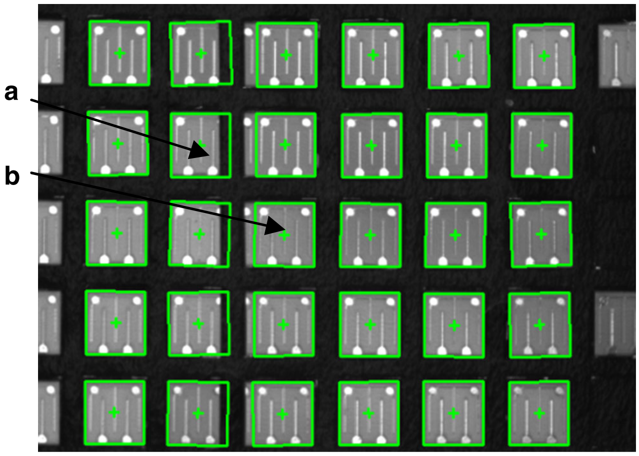

Fig. 2 A result of LED chips location of traditional template matching algorithms. *Green frame* denotes the led chip is located. **a** Fragmentary chip, **b** polycrystalline chip

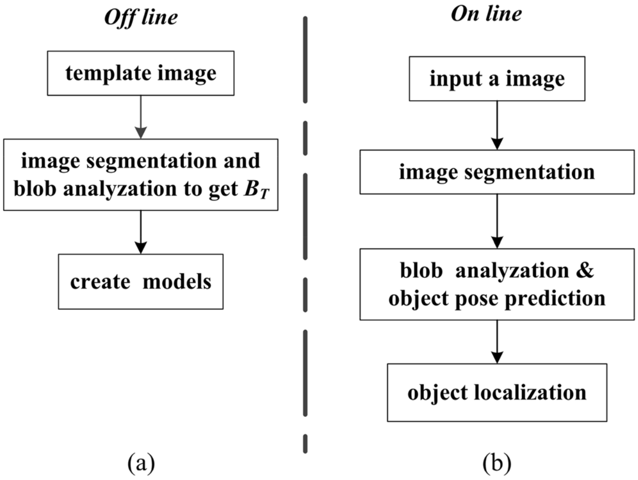

Fig. 3 Overview of the proposed algorithm. **a** Off-line process, **b** on-line process

chip in an image, Otsu method would not segment the image well, as shown in Fig. 4. In addition, edge-based methods and model-based methods are time-consuming and need prior knowledge of the objects. These methods need relatively complex models. There are many kinds of LED chips and different chips need different models. Watershed methods are likely to cause over-segmentation. Therefore, these popular methods may not be suitable for LED.

Because the gray value of the chip is larger than that of the background, even though the differences between the chip and background are not invariant, Eq. (1) can still be used to enhance the contrast between the chip and the background near the chip as much as possible.

$$s(i,j) = \left[p(i,j) - \overline{p}(i,j)\right] \times C + p(i,j) \tag{1}$$

$p(i,j)$ is the pixel gray value. $\overline{p}(i,j)$ is the mean pixel gray value, and the mask size is $K \times K$. C is a constant. $s(i,j)$ is the result. Because p of the LED chips is larger than that of the background, Eq. (1) makes p of the LED chips larger and p of the background near the LED chips smaller. However, p of electrodes and gold threads in the LED chip is at least 230; Eq. (1) may cause p, of the regions near the electrodes and gold threads, to become smaller. So the paper makes p, which is larger than 230, equal to p of the regions near the electrodes and gold threads, which can be easily determined during the off-line models creating process. The effect is shown in Fig. 5c, d; Eq. (1) apparently enhances the contrast between the background near the LED chips and the foreground. The most time-consuming part is to compute $\overline{p}(i,j)$, but it can reduce computation complexity to compute $\overline{p}(i,j)$ based on $\overline{p}(i-1,j)$ or $\overline{p}(i,j-1)$. Therefore, when the size of the image is $M \times N$, the computation complexity of Eq. (1) is about $O((2K+3) \times M \times N)$.

After image preprocessing through Eq. (1), it is easy to select a threshold T_h for Eq. (2) to get blobs. Then use some basically morphological operations like open operations and region fill-up operations to debur and fill some hollow regions. The computation complexity of Eq. (2) and the basically morphological operations is very low. The effect of image segmentation is shown in Fig. 5e, f.

$$b(i,j) = \begin{cases} 1 & if \quad s(i,j) \geq T_h \\ 0 & if \quad s(i,j) < T_h \end{cases} \tag{2}$$

2.2 Blob analyzation and object pose prediction

This process is to analyze the blobs obtained above, exclude the abnormal blobs, and get the poses of the MERs of the normal blobs to predict the poses of the objects. There is also a widespread method using the rectangularity of the blob to exclude the abnormal blob which may represent the fragmentary chip or polycrystalline chip. Although rectangularity can be used to detect many fragmentary and polycrystalline chips, the rectangularity of some fragmentary and polycrystalline chips is still high. As is shown in Fig. 6, it is easy to know that (a) and (b) are fragmentary or polycrystalline chips through rectangularity. However, the rectangularity of (c) and (d) is high and rectangularity is not suitable to be used to exclude these chips, whereas it is easy to know these chips are fragmentary or polycrystalline chips based on the width and length of the blobs' MERs. Another advantage of the second step of the proposed algorithm is that the poses of the blobs' MERs can be used to predict the poses of potential chips.

$$B_k = \{\alpha_k, l_k, w_k, P_k, \theta_k\} \tag{3}$$

Each blob B_k is represented by its area α_k, MER's length l_k, width w_k ($w_k \leq l_k$), central position P_k, and orientation θ_k ($-\pi/2 < \theta \leq \pi/2$), as shown in Eq. (3). Here, orientation θ_k is the angle between the short edge (long edge) of MER and the horizontal direction, as Fig. 7 shows. But if l_k is equal to w_k, the angle, whose absolute value is smaller, is chosen as the orientation θ_k. The method to compute the MER of a blob is shown in the Appendix.

$$B_T = \{\alpha_T, l_T, w_T, P_T, \theta_T\} \tag{4}$$

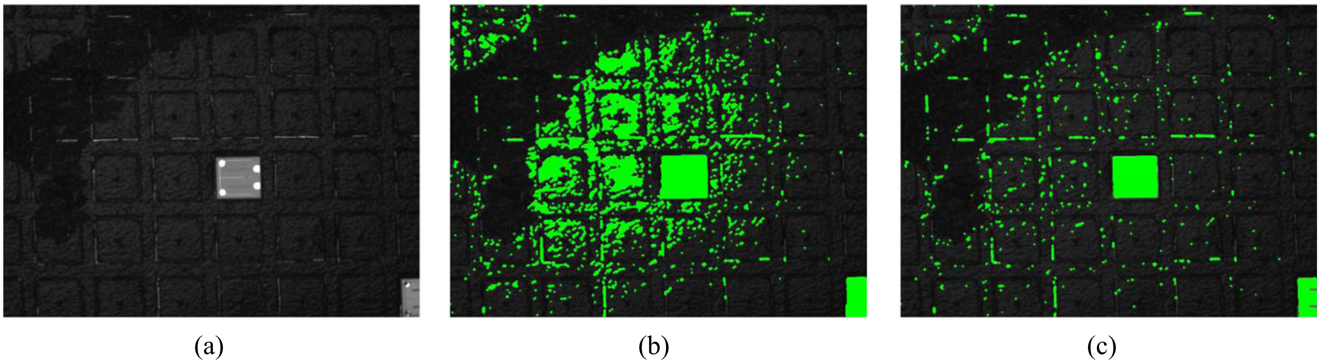

Fig. 4 **a** Original image, and there is only a chip in the image. **b** Segmentation result of Otsu method. **c** Segmentation result of the first step of the proposed algorithm

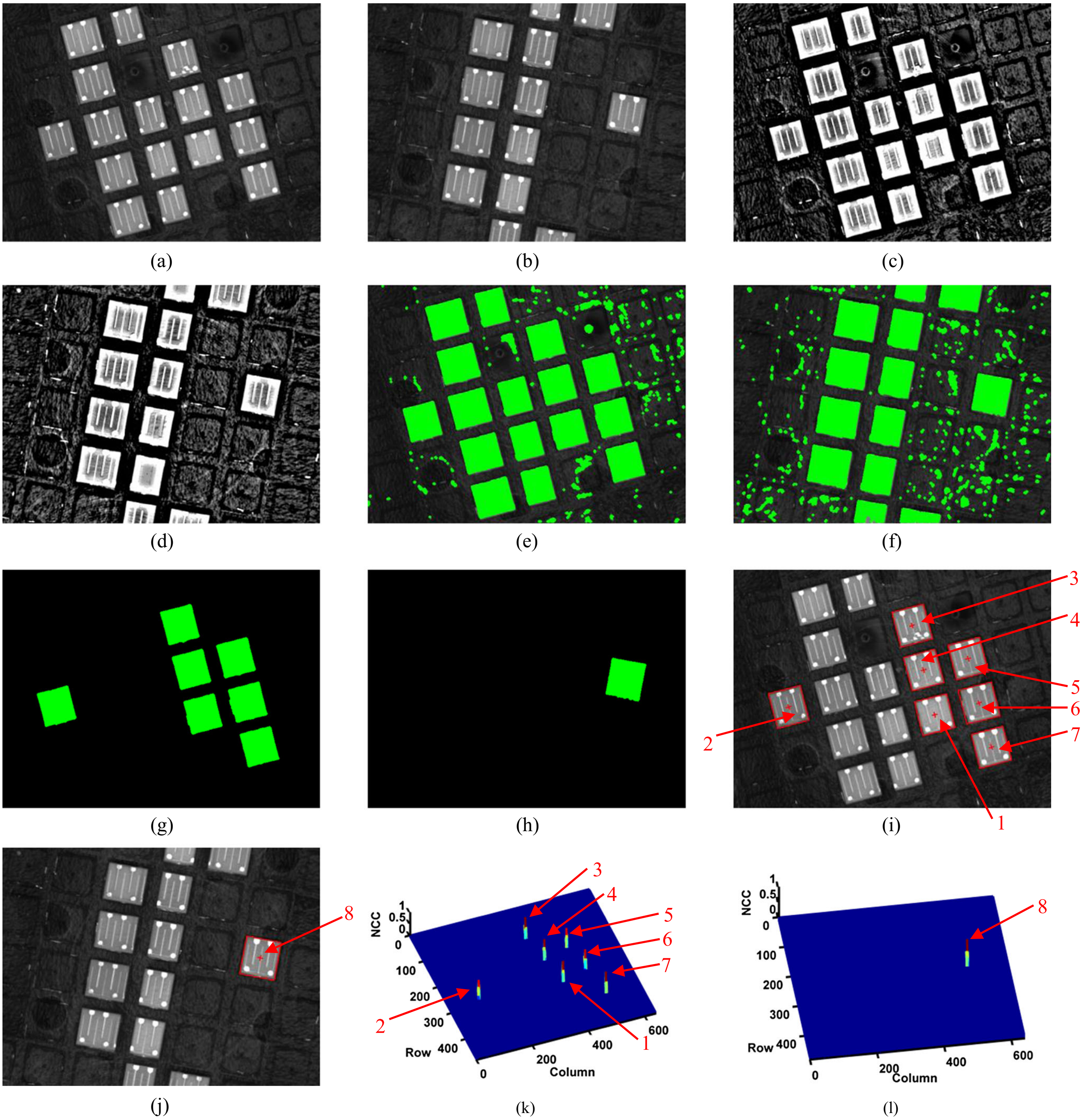

Fig. 5 Results of different processes. **a**, **b** Originally captured images. **c**, **d** Images after processing through Eq. (1). **e**, **f** Results of image segmentation. **g**, **h** Results after blob analyzation to select the normal blobs and exclude the abnormal blobs. **i**, **j** MERs (*red frames*) of normal blobs in the originally captured images. **k**, **l** NCC of every potential region around the center of MER. *1*,*2*,*3*,*4*,*5*,*6*,*7* in **k** correspond to the regions around the central positions of *1*,*2*,*3*,*4*,*5*,*6*,*7* in **i**. *8* in **l** corresponds to the regions around the central position of *8* in **j**

During the off-line process, when a template image is obtained, image segmentation method and blob analyzation are also used to get the template's blob B_T in Eq. (4). During the on-line process, comparing each blob B_i obtained after the segmentation of an input image to the B_T, it is easy to select the normal blobs and exclude the abnormal blobs. δ, ε, *and* ζ are three thresholds to decide whether blob B_i is normal and they are set by the user, as the flow shown in Fig. 8. As Fig. 5g, h show, the abnormal blobs are all excluded and the remained blobs are the normal blobs. As Fig. 5i, j show, the red frames are MERs of normal blobs. Obviously, the central position P_i of the MER is close to the central position of its corresponding object.

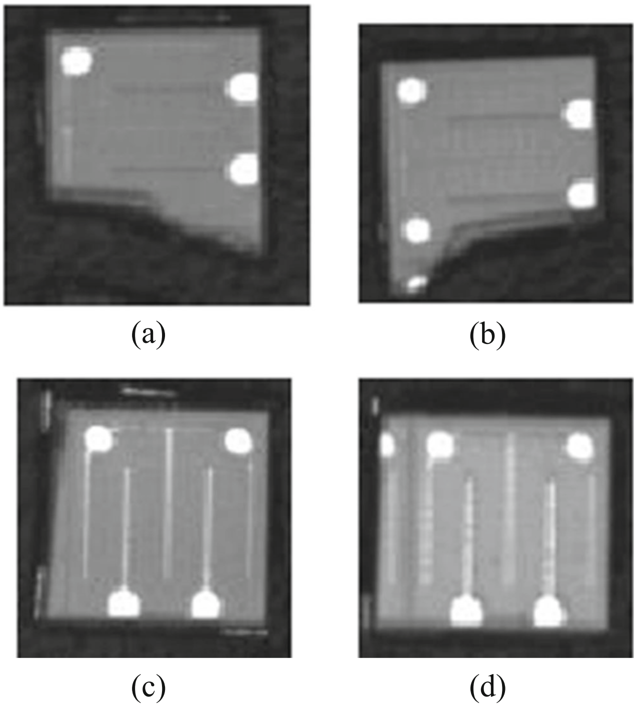

Fig. 6 **a**–**d** Four fragmentary or polycrystalline chips. The rectangularity of **a**–**d** is 0.77, 0.67, 0.95, and 0.98, respectively

So the paper uses P_i to predict the object's position. The orientation θ_i of the MER is close to the real angle of the object. However, the angle obtained through template matching is relative to the angle of template image. So the paper uses $\Delta\theta=\theta_i-\theta_T$ to predict the angle of the object. This process presents low computation complexity and reduces the area for object detection in the next process greatly.

2.3 Localization based on gradient orientation

This process here is used to discriminate object and determine the pose of the object precisely, around the position P_k and orientation θ_k of each remained blob's MER. The gradient orientation angle Θ ($0\leq\Theta<2\pi$) of every pixel can be computed through the ratio of G_x and G_y, which denote horizontal and vertical gradient magnitude respectively and can be obtained through some differential operators like Sobel Operator. Then, encode the gradient orientation angle through Eq. (5). $c(i,j)$ is the code of pixel, and $\Delta\Theta$ can be determined by the grades *Num* through Eq. (6).

$$c(i,j)=\left[\frac{\Theta(i,j)}{\Delta\Theta}\right] \tag{5}$$

where [•] is the Gaussian operation.

$$\Delta\Theta=\frac{2\pi}{Num} \tag{6}$$

During the off-line process, it is to extract gradient orientation features to create models. As shown in Fig. 9a, a LED chip mainly consists of luminous zone, electrodes, and gold threads. In order to improve the accuracy of localization and reduce the computation complexity, the gradient orientation features of the pixels on the edges between the chip and background, on the edges between electrodes and luminous zone, and on the edges between gold threads and luminous zone are chosen for matching. It is easy to choose a threshold T_m for gradient magnitude G in Eq. (7) to get those gradient orientation features, as shown in Fig. 9b. In fact, the angles of LED chips on wafers are always within ±15° and the angle discrimination accuracy is 0.5°. So the paper creates models every 0.25° from −15° to 15° based on template image.

$$G=\sqrt{{G_x}^2+{G_y}^2} \tag{7}$$

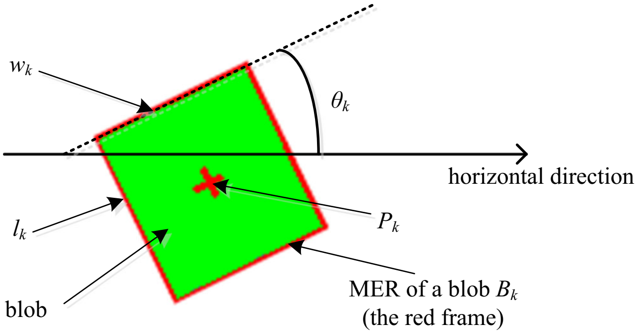

Fig. 7 MER of a blob B_k and its parameters

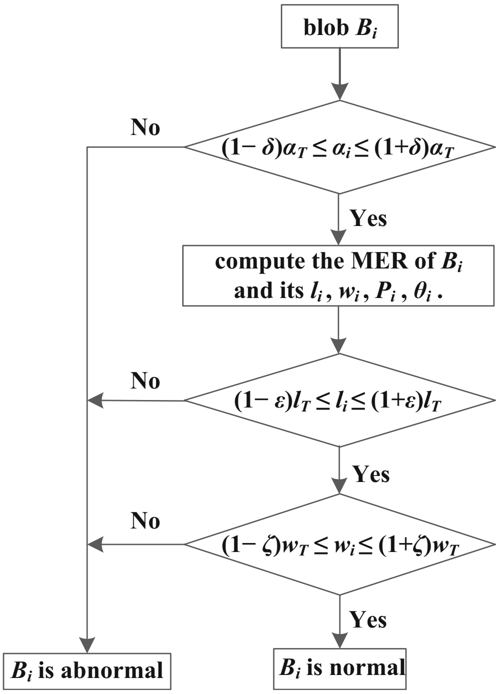
blob B_i
No
$(1-\delta)\alpha_T\leq\alpha_i\leq(1+\delta)\alpha_T$
Yes
compute the MER of B_i and its l_i, w_i, P_i, θ_i.
No
$(1-\varepsilon)l_T\leq l_i\leq(1+\varepsilon)l_T$
Yes
No
$(1-\zeta)w_T\leq w_i\leq(1+\zeta)w_T$
Yes
B_i is abnormal
B_i is normal

Fig. 8 Flow of blob analyzation to select normal blobs and exclude abnormal blobs

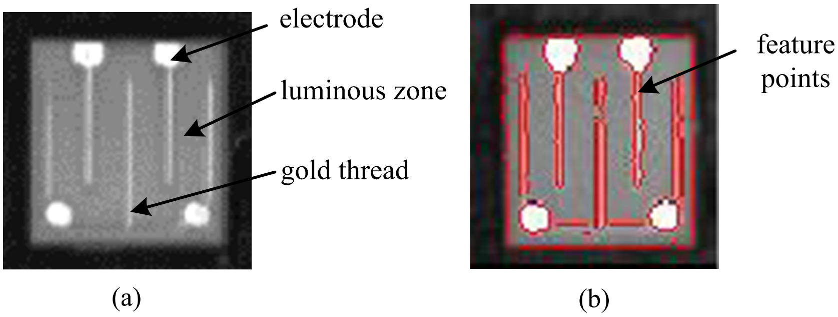

Fig. 9 LED chips. **a** Components of a chip. **b** Template for matching Fig. 5a and b. Gradient orientation features of the red points, which denote the pixels on the edges, are chosen for matching. T_m=27

On-line process is to locate the objects precisely, and it consists of two steps: matching at pixel accuracy level and matching at subpixel accuracy level. Normalized cross correlation (NCC) [17] in Eq. (8) is a classic similarity measure.

$$NCC(i,j) = \frac{1}{n}\sum_{(u,v)\in T}\frac{t(u,v)-m_t}{\sqrt{s_t{}^2}}\bullet\frac{C(i+u,j+v)-m_I(i,j)}{\sqrt{s_I{}^2(i,j)}} \tag{8}$$

$$m_t = \frac{1}{n}\sum_{(u,v)\in T} t(u,v) \tag{9}$$

$$m_I = \frac{1}{n}\sum_{(u,v)\in T} C(i+u,j+v) \tag{10}$$

$$s_t^2 = \frac{1}{n}\sum_{(u,v)\in T} (t(u,v)-m_t)^2 \tag{11}$$

$$s_I^2 = \frac{1}{n}\sum_{(u,v)\in T} (C(i+u,j+v)-m_I(i,j))^2 \tag{12}$$

Here, $t(\mu, \nu)$ is the selected pixel's code of the template. m_t is the mean value and s_t^2 is the variance. $C(i+\mu, j+\nu)$ is the corresponding pixel's code of the input image at a shifted position of the template region. m_I is the mean value and s_I^2 is the variance. The bigger the absolute value of NCC is, the more similar the template is to the detected region in the input image. Firstly, according to the poses of the remained blobs' MERs, use NCC to locate the LED chips at pixel accuracy level. Then, according to the poses at pixel accuracy level, locate the LED chips at subpixel accuracy level based on the demand of accuracy. Figure 5k, 1 shows the NCC of every potential region around the central position of the blob's MER. Because the paper only chooses the gradient orientation codes of some certain pixels to match the potential chips in some certain regions, the computation complexity of this process is low.

3 Experiment

This section is to verify the proposed algorithm from the following perspectives.

- Whether it is robust to illumination variation to a certain degree.
- How are the effect and efficiency of the algorithm?

The algorithm is tested in a computer with an Intel Core E6500 CPU of 2.94 GHz and RAM of 4 GB. Parameters are set for the paper: C=5. K=15 for $\overline{p}$ in Eq. (1). p=170 if p>230 in image segmentation process. T_h=90 for image segmentation in Eq. (2). δ=0.15. ε, ζ=0.1. T_m=27 for choosing the gradient orientation features of the pixels on the edges during the model creating process. The threshold for NCC during on-line process is 0.65.

The size of input image is 640×480.

3.1 Test whether the proposed algorithm is robust to illumination variation to a certain degree

In Fig. 11, panels a and g are two captured images under a normal illumination condition. Panels b and h in Fig. 11 are obviously darker while panels c and i (Fig. 11) are obviously brighter. Figure 10a is the template image for locating the LED chips in Fig. 11a–c. Figure 10b is the template image for locating the LED chips in Fig. 11g–i. The results of localization show that even if the illumination varies to a certain degree, only the normal chips are located and polycrystalline, fragmentary chips are excluded.

Because even if the illumination varies to a certain degree, p of the pixels of the chips is still larger than that of the pixels of the background, Eq. (1) can still enhance the contrast

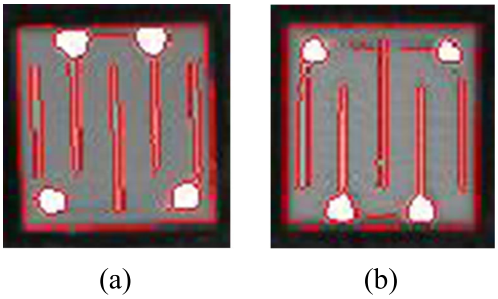

Fig. 10 Template images. **a** Template for Fig. 11a–c. **b** Template for Fig. 11g–i. *Gradient orientation features of the red points*, which denote the pixels on the edges, are chosen for matching

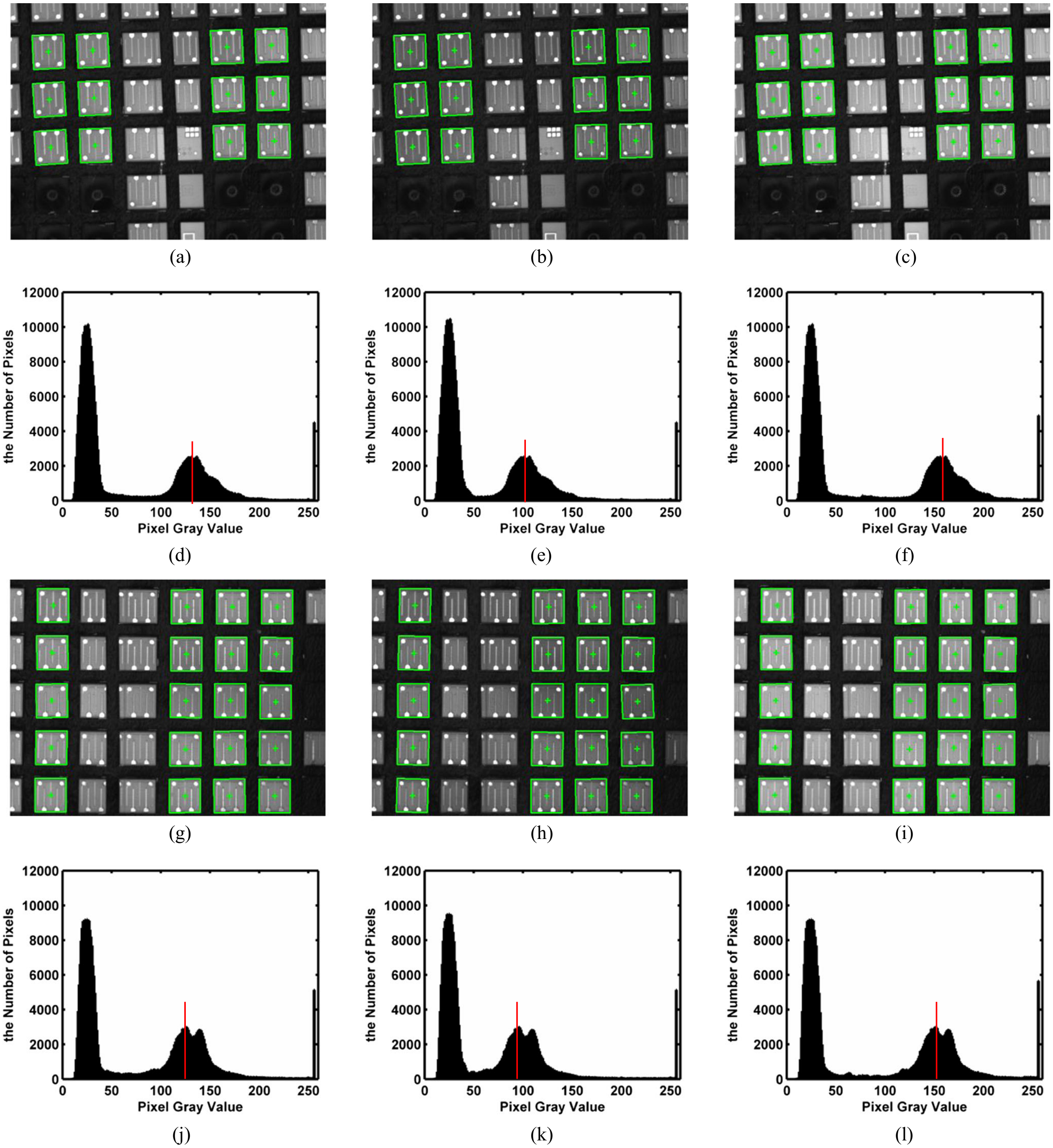

Fig. 11 Results of matching of images under different illumination conditions and the *gray histograms* of these images. **a**, **g** Images captured under normal illumination condition, **d**, **j** *gray histograms*. Panels **b** and **h** are darker and **e** and **k** are their *gray histograms*. Panels **c** and **i** are brighter and **f** and **l** are their *gray histograms*. The *green frames* denote the corresponding chips were located. The *green crosses in the center of the green frames* denote the central positions of chips obtained from NCC localization

between the chips and background near the chips. Then the image segmentation and blob analyzation will not be affected. Plus, gradient orientation codes are almost invariant to illumination variation. Therefore, it is shown that the proposed algorithm is robust to illumination variation to a certain degree.

3.2 Test the effect and efficiency of the proposed algorithm

Here, the paper compares the proposed algorithm to the recently published template matching algorithm [10] based on gradient orientation features to test the effect and efficiency of the algorithm. Two indices are used to evaluate the effect of the proposed method, misjudgment rate β, and leakage rate γ.

Misjudgment means failing to exclude the polycrystalline and fragmentary chips. Leakage means detecting normal chips as defective chips. Here, the paper divides the number of defective chips of misjudgment by the total number of defective chips to obtain misjudgment rate β, and it divides the number of normal chips of leakage by the total number of normal chips to obtain leakage rate γ. For each of the two indices, the smaller the value is, the better the algorithm is. Additionally, efficiency is an important factor for the proposed algorithm.

Five experiments were conducted on lots of LED chips provided by a local manufacturing company. Table 1 suggests that the template matching algorithm [10] based on gradient orientation is almost unable to exclude the defective chips. The misjudgment rate is at least 72.79 %. However, the proposed algorithm locates the normal chips quite well, with a leakage rate about 0.0 %, and it excludes most of the defective chips, with a misjudgment rate less than 2.7 %. Now, there is no unified standard for the leakage rate among LED manufacturers. A leakage rate less than 0.2 % is good and a leakage rate about 0.5 % can be accepted by the manufacturers. The leakage rate of the proposed algorithm is affected by the threshold for NCC. The larger the threshold for NCC, the higher the leakage rate of the proposed algorithm is. In fact, in a normal wafer, the possibility of appearing polycrystalline or fragmentary chips is less than 2 %. Therefore, if the proposed algorithm is used, the possibility of locating the polycrystalline or fragmentary chips is less than $2\% \times 3\% = 6 \times 10^{-4}$, which can satisfy the requirements about 0.1 % of the LED manufacturers. The misjudgment of the proposed algorithm is determined by the parameters ε and ζ, and we often set them to be 0.1. They can also be set by the users based on different kinds of LED chips. Additionally, the efficiency of the proposed algorithm is very high. The average time is about 0.90 ms for locating every chip on wafers which is much less than the time 2.10 ms of template matching algorithm [10] based on gradient orientation.

4 Conclusion

The paper proposes an algorithm which locates the LED chips on wafers and excludes the polycrystalline and fragmentary chips efficiently. The algorithm has three highlights: one is that it uses a simple but efficient method to enhance the contrast between the LED chips and the background near the chips. One is that it uses blob analyzation to exclude the abnormal blobs which may represent the polycrystalline and fragmentary chips. The last is that it uses poses of the MERs of normal blobs to predict the poses of potential chips, which reduce the area for object detection greatly. The experiments demonstrate that the algorithm is robust to illumination variation to a certain degree. The effect, with a low misjudgment rate and leakage rate, is very good, and the efficiency is very high. In addition, the algorithm can also be applied to locate other kinds of chips if the pixel gray value of the chips is larger than that of the background.

Acknowledgments This work was supported in part by Key New Products and New Technology Research and Development projects in Hubei Province, under Grant No.2014BAA007, and in part by Youth Science and Technology in Wuhan Morning Program under Grant No.201307230410828.

Appendix

The section shows the method to compute the MER of the blob. The angles of the chips on wafers are almost within ±15°, so Φ is within ±15°. We make Φ from −15° to 15° one by one. At each Φ, shown in Fig. 12, it is easy to find the points A and C, which represent the top and bottom points of the blob in orientation R. Similarly, it is easy to find points B

Table 1 Result of the five experiments on lots of led chips

ID	Normal chips 1	Defective chips 1	Proposed algorithm					Template matching algorithm based on gradient orientation				
			Normal chips 2	γ_1 (%)	Defective chips 2	β_1 (%)	t_1 (ms)	Normal chips 3	γ_2 (%)	Defective chips 3	β_2 (%)	t_2 (ms)
1	406	39	406	0.0	1	2.6	0.90	406	0.0	32	82.05	2.10
2	622	57	622	0.0	0	0	0.90	622	0.0	50	87.72	2.11
3	923	98	923	0.0	2	2.0	0.90	922	0.108	91	92.86	2.10
4	1130	125	1129	0.088	2	1.6	0.89	1130	0.0	111	88.8	2.09
5	1507	147	1507	0.0	4	2.7	0.90	1507	0.0	107	72.79	2.09

Normal chips 1 represent the total number of normal chips. Defective chips 1 represents the total number of polycrystalline and fragmentary chips provided for the experiment. Normal chips 2 and normal chips 3 represent the total number of normal chips located by the proposed algorithm and template matching algorithm [10] based on gradient orientation, respectively. Defective chips 2 and defective chips 3 represent the total number of polycrystalline and fragmentary chips located by both algorithms, respectively. t_1 and t_2 are the average time both algorithms take to locate every chip

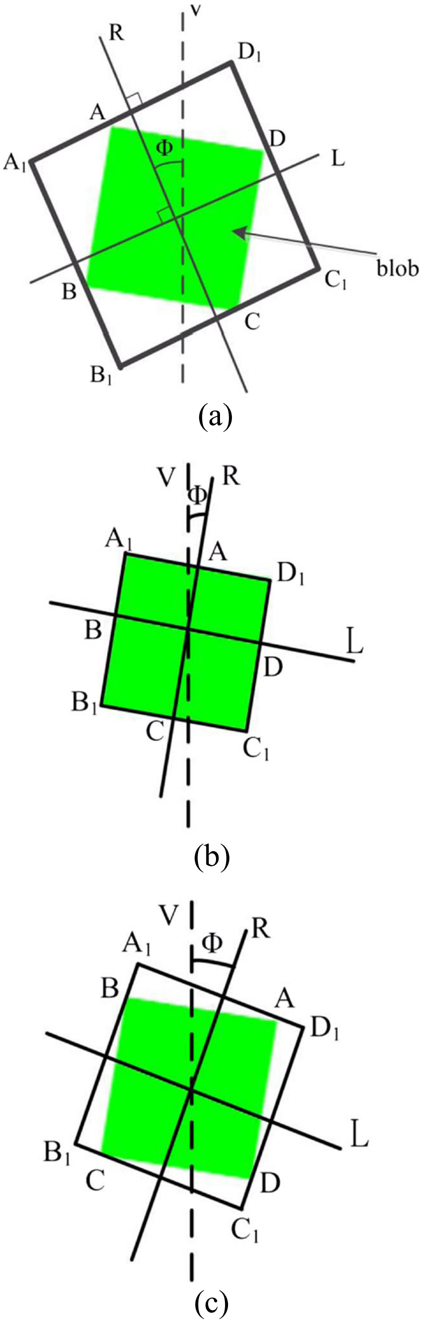

Fig. 12 The method to compute the MER of a blob. *V* denotes the vertical orientation. The *rectangle* A_1 B_1 C_1 D_1 in **b** is the MER

and D in orientation L (L⊥R). Through A, B, C, and D, we can find a rectangle A_1 B_1 C_1 D_1. After $\varPhi$ from −15° to 15° one by one, we can get 31 rectangles A_1 B_1 C_1 D_1. Among all the rectangles A_1 B_1 C_1 D_1, the one which has the smallest area is the MER. Figure 12b is the MER. It is easy to conclude that the computation complexity of this process to get MER is low.

References

1. Steele RV (2010) High-brightness LED market overview. Proc. SPIE 4445,Solid State Lighting and Displays, doi:10.1117/12.450027
2. B. L. L. W. Tao WU (2010) Automatic detect and match of LED dies basing on position. Processings of the Ninth International Conference on Machine Learning and Cybernetics,Qindao
3. Li Q, Zhang B (2006) A fast matching algorithm based on image gray value. J Softw 17:216–222
4. Ayache N, Faugeras OD (1986) A new approach for the recognition and positioning of two-dimensional objects. IEEE Trans Pattern Anal Mach Intell, pp. 44–54
5. Grimson WEL, Lozano-Perez T (1987) Localizing overlapping parts by searching the interpretation tree. IEEE Trans Pattern Anal Mach Intell, pp. 469–482
6. Koch MW, Kashyap RL (1987) Using polygons to recognize and locate partially occluded objects. IEEE Trans Pattern Anal Mach Intell, pp. 483–494
7. Ventura JA, Wan W (1997) Accurate matching of two-dimensional shapes using the minimal tolerance zone error. Image Vis Comput 15:889–899
8. Marimon D, Ebrahimi T (2007) Efficient rotation-discriminative template matching. Progress in Pattern Recognition, Image Analysis and Applications, pp. 221–230
9. Ullah F, Kaneko S (2004) Using orientation codes for rotation-invariant template matching. Pattern Recogn 37:201–209
10. X. Xu, P. van Beek and X. Feng (2014) High-speed object matching and localization using gradient orientation features. Proc. SPIE 9025, Intelligent Robots and Computer Vision XXXI: Algorithms and Techniques, 902507.
11. Cho H–J (2010) Wavelet transform based image template matching for automatic component inspection. Int J Adv Manuf Technol 50: 1033–1039
12. Crispin AJ, Rankov V (2007) Automated inspection of PCB component using a genetic algorithm template-matching approach. Int J Adv Manuf Technol 35:293–300
13. Anbu A, Agarwal G, Srivastava G (2002) A fast object detection algorithm using motion-based region-of-interest determination. 14th Int Conf Digital Signal Process 2:1105–1108
14. Forssen PE, Moe A (2005) View matching with blob features. The 2nd Canadian Conference on Computer and Robot Vision, pp. 228–235
15. Goel S, Dabas S (2013) Vehicle registration plate recognition system using template matching. 2013 International Conference on in Signal Processing and Communication (ICSC), pp. 315–318
16. Otsu N (1975) A threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern 11:23–27
17. Di Stefano L, Mattoccia S, Mola M (2003) An efficient algorithm for exhaustive template matching based on normalized cross correlation. 12th International Conference on Image Analysis and Processing, pp. 322–327