

A Binarization Approach for Wafer ID Based on Star-Shape Filter

Wei-Chih Hsu¹, Tsan-Ying Yu^{2,*}, and Kuan-Liang Chen¹

¹Department of Computer and Communication, National Kaohsiung First University of Science and Technology, Kaohsiung, Taiwan, ROC.

²Institute of Mechanical and Electro-Mechanical Engineering, National Formosa University, Yunlin, Taiwan, ROC.

Received 15 April 2012; received in revised form 17 May 2012; accepted 22 June 2012

Abstract

The binarization of wafer ID image is one of the key techniques of wafer ID recognition system and its results influence the accuracy of the segmentation of characters and their identification directly. The process of binarization of wafer ID is similar to that of the car license plate characters. However, due to some unique characteristics, such as non-uniform illumination, the unsuccessive strokes of wafer ID, it is more difficult to make of binarization of wafer ID than the car license plate characters. In this paper, a wafer ID recognition scheme based on Star-shape filter is proposed to cope with the serious influence of uneven luminance. The testing results show that our proposed approach is efficient even in situations of overexposure and underexposure the wafer ID with high performance.

Keywords: binarization, wafer ID, star-shape filter, car license plate

1. Introduction

It is important for Assembly / Test process of semiconductor manufacturing to link to a wafer database by the wafer identification (wafer ID), which is inscribed by a laser scribe system. In order to trace wafer IDs easily, the wafer ID should be recognized automatically. The binarization stage plays an important role in wafer ID recognition.

There are many binarization techniques which can be categorized as global thresholding and local thresholding. Global thresholding [1, 2] selects a single threshold value from the histogram of the entire image. Local thresholding [3, 4] uses localized gray-level information to choose multiple threshold values; each is optimized for a small region in the image. Global thresholding is simpler and easier to implement but its result relies on uniform illumination. It is effective for images with an obviously bimodal histogram and not suitable for images with low contrast. Otsu's algorithm [5], a conventional global thresholding method, reflects the intensity distribution of an image, but its property of a single threshold results in poor robustness. Local thresholding methods can deal with non-uniform illumination but they are slow. Bernsen algorithm [6] is a typical local threshold technique. It utilizes the extreme of the neighbor to calculate every pixel's threshold. There are some surveys of various binarization methods reported in the literature [7-9]. More recent studies on this subject can be found [10, 11].

In this paper, a new image binarization method for wafer ID that we present is a local thresholding method. The binarization of wafer ID recognition is very similar to that of car license plate recognition (LPR). However, because of some

* Corresponding author. E-mail address: yotnyg@gmail.com
Tel.: +886-7-6077139; Fax: +886-7-6077009

wafer ID images' special features, the binarization scheme must be modified in order to enhance recognition performance. Some special features are described as follows.

Usually, both the brightness and contrast of the wafer ID images are not as sharp as those of the license plate images. There are other disturbances which are caused by non-uniform illumination or poor wafer processes. The wafer scribes are often unsuccessful. Each character is composed of unsuccessful horizontal scribes caused by laser scribe. The unsuccessful wafer scribes will mislead character recognition. Hence, the image binarization stage should take care of it specially.

- (1) The spaces between adjacent characters are irregular.
- (2) Though the Semiconductor Equipment and Materials International (SEMI) standard is available, the styles of characters will vary as equipments or processes change.

The remainder of this paper is organized as follows. In Section 2, Otsu's and Bernsen's methods are briefly reviewed. Section 3 presents the algorithm in detail; Section 4 shows the experimental results. Finally, conclusion and the future work will be addressed.

2. Two Conventional Binarization Methods

In this section, we introduce two conventional binarization methods, Otsu's and Bernsen's methods, and analyze their limitations when they are applied in the binarization of wafer ID.

2.1. Otsu's Method

An image is a 2D grayscale intensity function, and contains N pixels with gray levels from 1 to L . The number of pixels with gray level i is denoted f_i , giving a probability of gray level i in an image of

$$p_i = \frac{f_i}{N} \quad (1)$$

In the case of bi-level thresholding of an image, the pixels are divided into two classes, C_1 with gray levels $[1, \dots, t]$ and C_2 with gray levels $[t+1, \dots, L]$. Then, the gray level probability distributions for the two classes are

$$C_1: \frac{p_1}{\omega_1(t)}, \dots, \frac{p_t}{\omega_1(t)} \text{ and } C_2: \frac{p_{t+1}}{\omega_2(t)}, \frac{p_{t+2}}{\omega_2(t)} \dots \frac{p_L}{\omega_2(t)}, \text{ where } \omega_1(t) = \sum_{i=1}^t p_i \quad (2)$$

and

$$\omega_2(t) = \sum_{i=t+1}^L p_i \quad (3)$$

Also, the means for classes C_1 and C_2 are

$$\mu_1 = \sum_{i=1}^t \frac{ip_i}{\omega_1(t)} \quad (4)$$

and

$$\mu_2 = \sum_{i=1}^L \frac{ip_i}{\omega_2(t)} \quad (5)$$

Let μ_T be the mean intensity for the whole image. It is easy to show that

$$\omega_1\mu_1 + \omega_2\mu_2 \quad (6)$$

$$\omega_1 + \omega_2 = 1 \quad (7)$$

Using discriminant analysis, Otsu defined the between-class variance of the thresholded image as

$$\sigma_B^2 = \omega_1(\mu_1 - \mu_T)^2 + \omega_2(\mu_2 - \mu_T)^2 \quad (8)$$

For bi-level thresholding, Otsu verified that the optimal threshold t^* is chosen so that the between-class variance σ_B^2 is maximized; that is,

$$t^* = \max_{0 \leq t \leq L-1} \{ \sigma_B^2(t) \} \quad (9)$$

When the histogram of wafer ID image has a deep valley, the bottom of the valley is regarded as the threshold to separate the characters from background. But when the image is polluted by noisy, the illumination is poor or the background is complex, the histogram will have more than one deep valley. Then we cannot separate the characters from background correctly by this method.

2.2. Bernsen's Method

The above original designs are transformed individually into their corresponding generalized chains (kinematic chains). The generalized chain will be involved in various types of members (edges) and joints (vertices, or said kinematic pairs) for all possible assembly in the following steps.

Bernsen's method is a typical adaptive thresholding method. For each pixel in the image, a threshold has to be calculated. If the pixel value is below the threshold, it is set to the background value, otherwise to the foreground value. An alternative approach to find the local threshold is to calculate the mean of the minimum and maximum values of the local regions. The assumption of this method is that smaller image regions are more likely to have approximately uniform illumination.

The drawback of this method is that it is computational expensive and bring "ghost" phenomena sometimes. The size of the neighborhood must be large enough to cover sufficient foreground and background pixels, otherwise a poor threshold is chosen. Oppositely, choosing too large regions can violate the assumption of approximately uniform illumination.

The Bernsen's arithmetic can be accomplished by the adaptive threshold operator which maps a given grayscale image $F(x,y)$ into a binary image $B(x,y)$ using a thresholding curved surface $T(x,y)$. We can operate as the following steps:

1. For each pixel in the image, a threshold is calculated.

$$T(x,y) = \frac{[\max_{(x,y) \in P_{xy}} F(x,y) + \min_{(x,y) \in P_{xy}} F(x,y)]}{2} \quad (10)$$

2. Threshold the image according to the thresholding curved surface $T(x,y)$.

$$B(x,y) = \begin{cases} 255 & F(x,y) \geq T(x,y) \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

where $F(x,y)$ is the grayscale image, $B(x,y)$ is the binary image, $T(x,y)$ is the thresholding curved surface, P_{xy} is the template whose center is (x,y) and size is $(2W+1) \times (2W+1)$, and W is the maximum of estimated stroke width.

3. The Algorithm of Binarization Approach

The binarization stage plays an important role in wafer ID recognition. The binarization stage affects the accuracy of character recognition. If a poor binarization algorithm is used, accurate character regions cannot be obtained and the characters cannot be extracted from the wafer ID correctly. While a highly accurate binary image is produced, the character segmentation process will go smoothly and the characters can be obtained from the wafer ID efficiently. The flowchart of our binarization approach is shown in Fig. 1. In the remainder of this paper, the detail of the binarization algorithm will be explored.

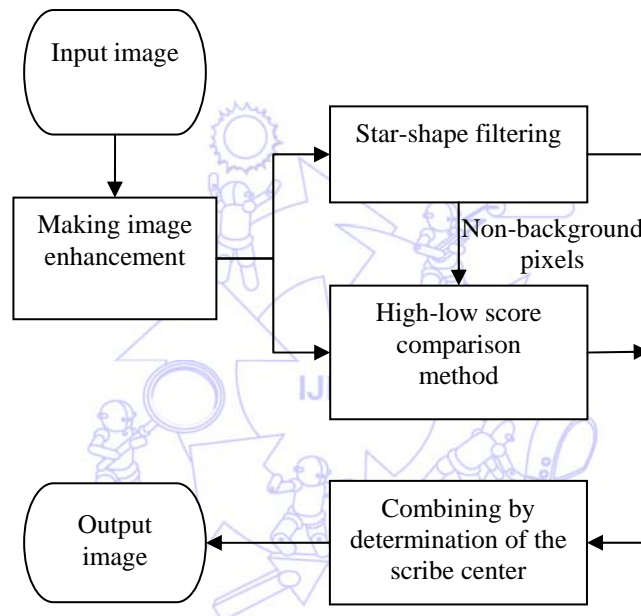


Fig. 1 Flowchart of the image binarization algorithm

3.1. Image Enhancement

Insufficient exposure, overexposure or uneven luminance will lead to low contrast between the wafer ID string and background. Therefore, we have to examine the contrast of wafer ID image to determine whether to enable image enhancement or not. First, we analyze the gray level histogram of the wafer ID sub-image. Then, in the gray level histogram, a curve Q denoting the ratio of the accumulated number of pixels of some gray level to total pixels is calculated. We define a gray level br as background reference in which the number of the pixels with gray level less or equal to br is 40% of total pixels. We also define a level sr as stroke reference in which the number of the pixels with gray level greater or equal to sr is 90% of total pixels. One example is depicted in Fig. 2(a)(b). If the difference of the gray level between stroke reference(sr) and background reference(br) is less than δ_l , it means that the wafer ID image is low contrast. In other words, we have to enhance image by enlarging the difference to δ_l . The enhancement equation is described in the following:

$$r = \frac{sr + br}{2} \quad (12)$$

$$f(x) = r + \frac{\delta_1}{sr - br}(x - r) \quad (13)$$

Note that the difference is not allowed to be more than δ_1 , otherwise the noises will increase after enhancement. The threshold $\delta_1=25$ is determined by experiments. An example of an original and enhanced images are shown in Fig. 3(a)(b). The low contrast wafer ID images are enhanced obviously.

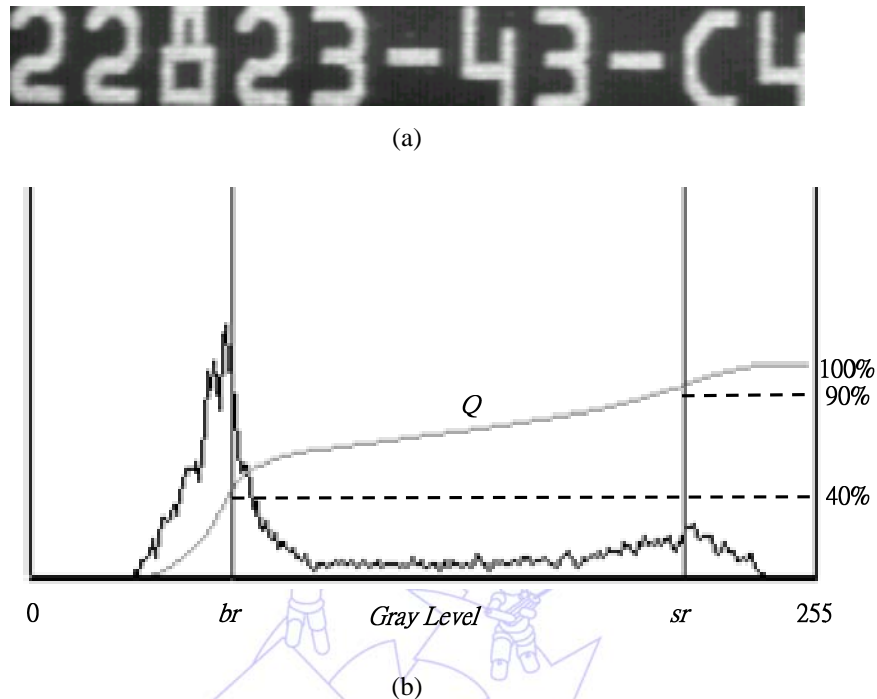


Fig. 2 (a) The cut wafer ID sub-image after locating procedure (b) The gray level histogram and the accumulated ratio curve Q

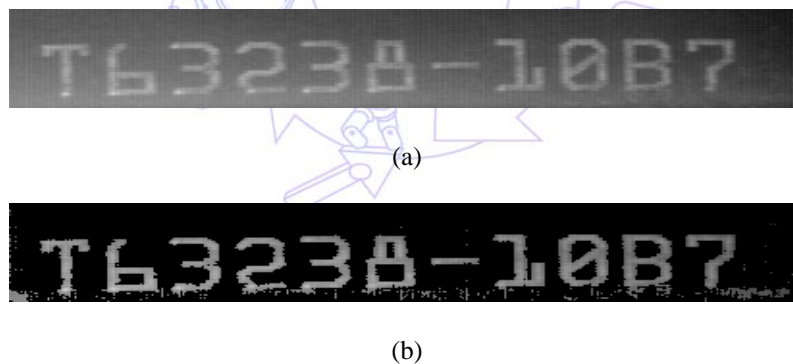


Fig. 3 (a) The original image before enhancement (b) Resulted image after enhancement.

3.2. Image Binarization

Once we have separated the wafer ID string from background, we employ two stages to carry out binarization. At first stage, two methods are applied. The first one called the star-shape filtering will remove luminance disturbance. The second method called the high-low score comparison method which primarily aims to eliminate the noise near wafer scribe edges. At the second stage, we compare and combine the two output images resulted from the star-shape filter and the high-low score comparison method, respectively. At the final stage, the unsuccessful vertical edges are compensated.

3.2.1. The Star-Shape Filtering

Because of background and luminance disturbance effect, the closer to wafer ID scribe center, the higher the gray level will be. Furthermore, the orientation of wafer ID scribe is a very important feature. The orientation of standard wafer ID

scribes usually can be classified into vertical, horizontal, -45 degree, and 45 degree orientation. For the measured pixel, judging from the gray level of the pixels distributing in its corresponding four directions, we can determine whether the measured pixel is foreground pixel or not. Since the scribe width is between 4 and 7 pixels, we set the window size as 9×9 . Let the measured pixel as the window center. We divide the 81 samples of 9×9 window into upper, lower, left and right regions. For each region, we define its flags whose values are true or false are determined by the following equations:

$$flag_{right} = \begin{cases} true & \text{if } \sum_{k=1}^4 f(x+k, y) - f(x, y) < \delta_2 \text{ or } \sum_{k=1}^4 f(x+k, y) = 0 \\ false & \text{otherwise} \end{cases} \quad (14)$$

$$flag_{upper_right} = \begin{cases} true & \text{if } \sum_{k=1}^4 f(x+k, y+k) - f(x, y) < \delta_2 \text{ or } \sum_{k=1}^4 f(x+k, y+k) = 0 \\ false & \text{otherwise} \end{cases} \quad (15)$$

$$flag_{upper} = \begin{cases} true & \text{if } \sum_{k=1}^4 f(x, y+k) - f(x, y) < \delta_2 \text{ or } \sum_{k=1}^4 f(x, y+k) = 0 \\ false & \text{otherwise} \end{cases} \quad (16)$$

$$flag_{upper_left} = \begin{cases} true & \text{if } \sum_{k=1}^4 f(x-k, y+k) - f(x, y) < \delta_2 \text{ or } \sum_{k=1}^4 f(x-k, y+k) = 0 \\ false & \text{otherwise} \end{cases} \quad (17)$$

$$flag_{left} = \begin{cases} true & \text{if } \sum_{k=1}^4 f(x-k, y) - f(x, y) < \delta_2 \text{ or } \sum_{k=1}^4 f(x-k, y) = 0 \\ false & \text{otherwise} \end{cases} \quad (18)$$

$$flag_{lower_left} = \begin{cases} true & \text{if } \sum_{k=1}^4 f(x-k, y-k) - f(x, y) < \delta_2 \text{ or } \sum_{k=1}^4 f(x-k, y-k) = 0 \\ false & \text{otherwise} \end{cases} \quad (19)$$

$$flag_{lower} = \begin{cases} true & \text{if } \sum_{k=1}^4 f(x, y-k) - f(x, y) < \delta_2 \text{ or } \sum_{k=1}^4 f(x, y-k) = 0 \\ false & \text{otherwise} \end{cases} \quad (20)$$

$$flag_{lower_right} = \begin{cases} true & \text{if } \sum_{k=1}^4 f(x+k, y-k) - f(x, y) < \delta_2 \text{ or } \sum_{k=1}^4 f(x+k, y-k) = 0 \\ false & \text{otherwise} \end{cases} \quad (21)$$

where f is the image after enhancement, $f(x, y)$ is the measured pixels and δ_2 is a threshold, determined as follows.

$$\delta_2 = \begin{cases} \delta_4 & \text{if } sr - br \geq \delta_1 \text{ and } sr - br \leq \delta_3 \\ \delta_5 & \text{otherwise} \end{cases} \quad (22)$$

where sr and br are the stroke and background reference obtained in previous section, and the threshold δ_3 , δ_4 and δ_5 are 50, 20 and 15 determined by experiments.

For the measured pixel, each flag is to indicate which direction's pixels are continuously black. The upper region is

related to flags of $flag_{right}$, $flag_{upper_right}$, $flag_{upper}$, $flag_{upper_left}$ and $flag_{left}$ as depicted in Fig. 4(b), the lower region $flag_{left}$, $flag_{lower_left}$, $flag_{lower_right}$ and $flag_{right}$ as depicted in Fig. 4(c); the left region $flag_{upper}$, $flag_{upper_left}$, $flag_{left}$, $flag_{lower_left}$ and $flag_{lower}$ as depicted in Fig. 4(d); the right region $flag_{upper}$, $flag_{upper_right}$, $flag_{right}$, $flag_{lower_right}$ and $flag_{lower}$ as depicted in Fig. 4(e). If any one of these regions whose related flags are all “true”, then the measured pixel will be set to 0, else the measured pixel will remain the original value. Sometimes, the variance of the gray level near wafer scribe edges is too large, noise can not be removed. The high-low score comparison method is to solve this problem.

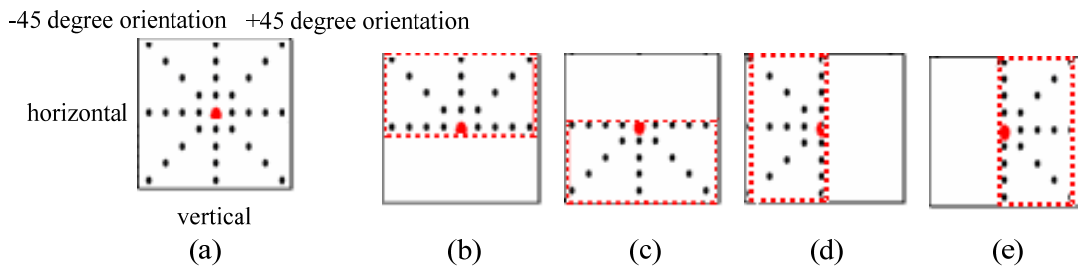


Fig. 4 (a) The four distribution orientations of the star-shape filter window (b) The upper region (c) The lower region (d) The left region (e) The right region

3.2.2. The High-Low Score Comparison Method

Assume $G(x,y)$ is the image after the star-shape filtering. Let the $HighScore_{avg}$ be the average of the gray level of the pixels which are higher than that of $G(x,y)$ in the window. And let the $LowScore_{avg}$ be the average of the gray level of the pixels in the window which are lower than that of $G(x,y)$ in the window. The closer the measured pixels to wafer scribe centers, the higher the values of $G(x,y) - LowScore_{avg}$ and the lower the values of $HighScore_{avg} - G(x,y)$ are. In order to separate out the pixels of the higher gray level from the background, we evaluate and check whether $G(x,y)$ is greater than $LowScore_{avg} + \delta_6$ or not. Utilizing this criteria, the measured pixels near wafer scribe edges will be classified into wafer ID or background by this method. The method is implemented as follows.

$$H(x,y) = 0 \quad \text{if} \quad G(x,y) = 0 \quad (23)$$

$$HighScore_{avg} = Average \left(\sum_{i=-4}^4 \sum_{j=-4}^4 f(x+i, y+j) \right) \quad \text{if} \quad G(x,y) < \sum_{i=-4}^4 \sum_{j=-4}^4 f(x+i, y+j) \quad \text{and} \quad G(x,y) \neq 0 \quad (24)$$

$$LowScore_{avg} = Average \left(\sum_{i=-4}^4 \sum_{j=-4}^4 f(x+i, y+j) \right) \quad \text{if} \quad G(x,y) > \sum_{i=-4}^4 \sum_{j=-4}^4 f(x+i, y+j) \quad \text{and} \quad G(x,y) \neq 0 \quad (25)$$

where f is the image after enhancement, G is the image after the star-shape filtering, H is the output image, δ_6 the threshold, (x,y) the measured pixel index. The high-low score comparison method is described as follows:

$$H(x,y) = \begin{cases} 255 & \text{if } G(x,y) > LowScore_{avg} + \delta_6 \wedge G(x,y) > (LowScore_{avg} + HighScore_{avg})/2, \\ 0 & \text{otherwise} \end{cases} \quad (26)$$

If $G(x,y) > LowScore_{avg} + \delta_6$ and $G(x,y) > (LowScore_{avg} + HighScore_{avg})/2$, then the measured pixels are set to 255, otherwise they are set to 0. The threshold δ_6 is set as 15 by experiments. Apply the high-low score comparison method to Fig. 5(b). Although the variance of the gray level is large, most noise near wafer scribe edges can be removed, processing result of

the high-low score comparison method is illustrated in Fig. 5(c).

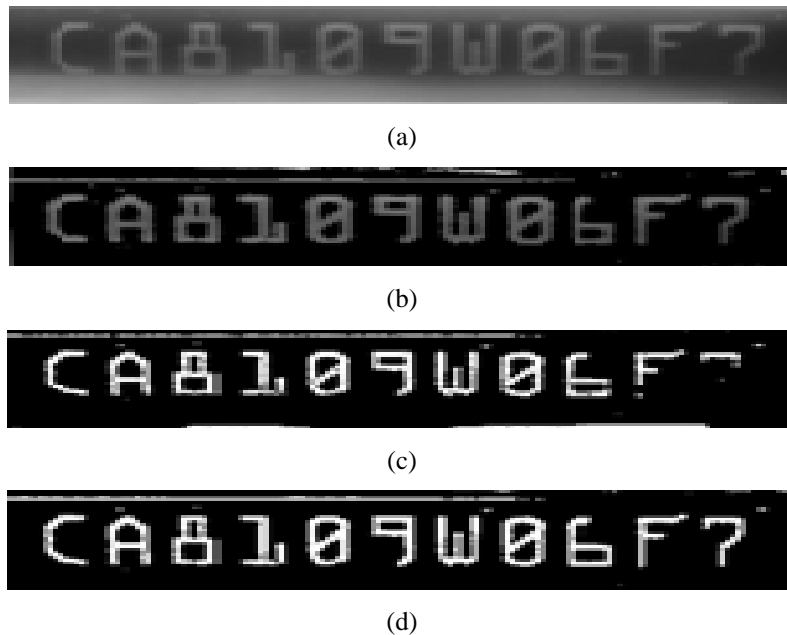


Fig. 5 (a)Original image (b)Resulted image after star-shape filtering (c) Processing result of the high-low score comparison method (d) Processing result that combined by (b) and (c) and determined by the characteristic of wafer scribe

3.2.3. Combining by Determination of the Scribe Center

The high-low score comparison method aims to deal with those noises which near to wafer scribe edges but not eliminated out by the star-shape filter. As before, the window size is set to 9×9 , and the measured pixel is set at the center.

Since the two images G and H resulting from the star-shape filter and the high-low score comparison method respectively are not the same for all pixels, we have to combine them in some way. These processing steps are described as follows.

Step 1: If both of them are background pixels in the same position, then the measured pixel is set to 0. If both of them are bright pixels, then the measured pixel is set to 255. Otherwise, its gray level has to be determined in next step by some rules.

Step 2: While we zoom in to observe the cross section of a wafer ID character image, the pixels near the scribe center are brighter than those far away from the scribe center. A typical example of character 'A' is shown in Fig. 6(a). Its histogram shown in Fig. 6(b) also reveals the phenomenon. Based on this fact and the phenomenon, the directions of standard wafer scribes can be classified into four types: vertical, horizontal, slope +1 line, slope -1 line. By the star-shape filtering method, reserved pixels have indicated the direction for wafer scribes of G .

Step 3: The direction for wafer scribes at the measured pixel is obtained in step 2. The cross direction of wafer scribes at the measured pixel can also be acquired. The original gray image is checked in this step. Base on the fact that the width of a scribe is always between 4 and 7 pixels; along with cross direction of wafer scribes at the measured pixel and within 5 pixels distance from the measured pixel, the pixels with the maximum gray level value are regarded as the center of the wafer scribes. If the distance between the measured pixel and the center of the wafer scribe is less than or equal to 2 pixels, then the gray level value of the measured pixel is set to 255, otherwise 0. Fig. 5(d) is the combination result of Fig. 5(b) and Fig.5 (c).

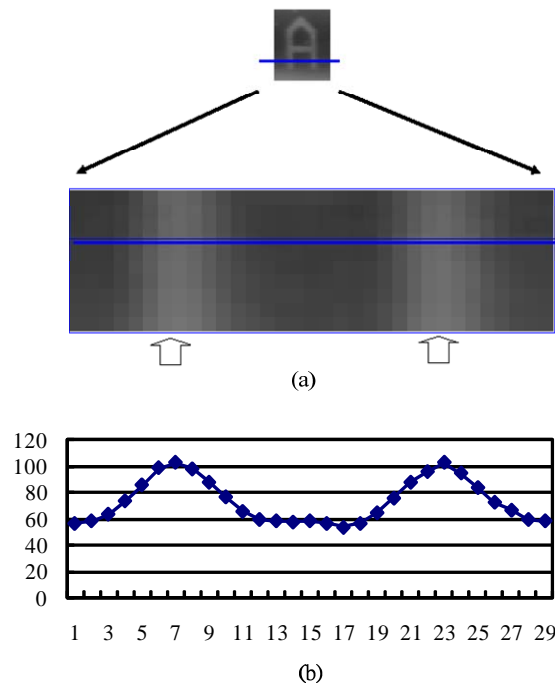


Fig. 6 (a) The closer the scribe center; the more outstanding the gray levels. The arrows indicate the scribe center (b) The histogram of (a) The locations near to the scribe center have peaks

3.2.4. Unsuccessful Vertical Edge Compensation

Almost all wafer scribes have unsuccessful edges in vertical direction as depicted in Fig. 7(a). We have to compensate the dark pixel and eliminate the noises to make the strokes continuous. After the first stage of binarization, unsuccessful edges in vertical direction stand out clearly, as shown in Fig. 7(b). Unsuccessful vertical edge compensation can be done according to the following equations.

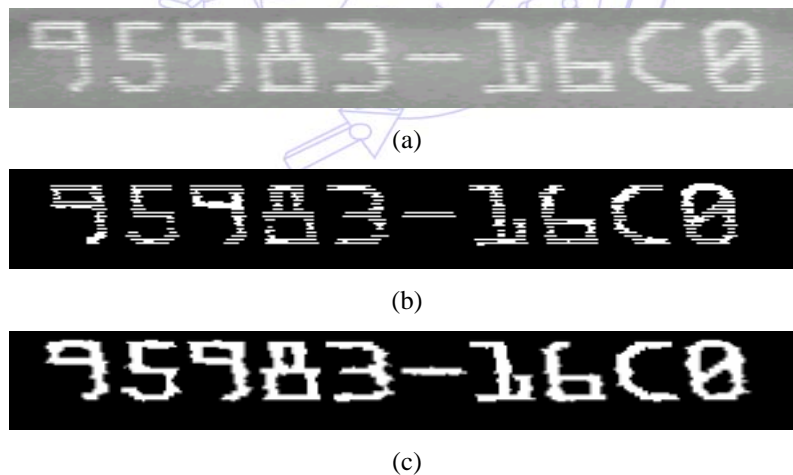


Fig. 7 (a) The wafer scribe edges are unsuccessful in the original image. (b) In the processed image, the wafer scribe unsuccessful edges come out clearly. (c) The processing result shows the vertical edges are compensated in the processed image.

where g_5 is the image after the scribe center distance determination and 765 comes from 255 times 3. The above equation is to check whether three of any five adjacent pixels in vertical direction are all bright pixels or not. If true, then the gray level values of measured pixels are set to 255, else 0. The processing result is depicted in Fig. 7(c). Obviously, the unsuccessful edges are compensated.

$$g_6(i, j) = \begin{cases} 255 & \text{if } \sum_{k=-2}^2 g_5(i, j+k) \geq 765 \\ 0 & \text{otherwise} \end{cases} \quad (27)$$

4. Experimental Results And Analysis



Fig. 8 (a) Original wafer ID image (b) The images after enhancement and star-shape filtering The images after high-low score comparing (d) The final binarized images after combining by determination of the scribe center (e) Binarization results by Otsu's method (f) Binarization results by BernSon's method

There are several thresholds and parameters in their image processing steps for segmenting out the characters of wafer ID. These parameters may be regarded as system parameters. They are optimized parameters that are obtained from training image data by design of experiment. For example, the masking operation is repeated three times. While training the image data, the first operation is repeated only once, and the result is not sufficient. So, the second operation is repeated twice. The improvement is better than previous step. As the same previous steps, the very little improvement is made after the fifth time operation. Hence, the optimized times for of masking operations is obtained. As the same steps, the other parameters can be optimized.

We present several tests images in this section. The representative experimental results are shown in Fig. 8. Experiments involving three types of original wafer ID image as shown in Fig. 8(a), which are arranged in order from left to right, wafer ID image with normal, wafer ID image with overexposure, wafer ID image with underexposure. Fig. 8(b) shows the images after enhancement and star-shape filtering from Fig. 8(a). Fig. 8(c) depicts the images after high-low score comparing from Fig. 8(a). Fig. 8(d) illustrates the final binarized images after combining by determination of the scribe center from Fig. 8(c). Fig. 8(e)(f) shows the binarization results by Otsu's method and Bernsen's methods, respectively.

By analyzing the above results, we can see that the Otsu's method has serious drawbacks. The left image and the middle image in Fig. 8(a) are normal and overexposure, respectively. Non-uniform illumination will cause the conflict between the object loss and the noise increase. The wafer ID image can not be thresholded effectively. The right image in Fig. 8(a) is underexposure, but uniform illumination will get better performance than the former one. No matter poor or non-uniform illumination, our method can still get more details than Otsu's method. The object boundaries are much delicate for our method. From Fig. 8, we can directly see that our approach allows to differentiate the characters from the background efficiently.

The result image of Ostu method is quite fuzzy and the characters aren't separated from each other. The one of Bernsen's method has obvious ghosts and doesn't favor the character recognition. In both of them, character outlines are broken and not clear. By our proposed method, the character stroke is more exquisite.

This is a psycho-visual experiment that is performed to ask a group of people to compare the results of the proposed binarization method with the results obtained by independent binarization techniques. This test is based on ten test Wafer ID images captured by CCD. For these images we obtain and print the binary images produced by the proposed method and the comparison methods. The printed images were given to 20 persons and asked them to detect visually the best binary image of each image. A percentage of 94% (188/200) of the answers, state that the proposed technique, using the weighting values, result to the best binary image.

With the proposed binarization method, 92% of 125 sample vehicle wafer ID images are correctly recognized. There are 1316 characters in 121 wafer ID images which step in recognition stage successfully. Total 1308 ones are successfully segmented out. The ratio of character recognition is high to 99.39%. The results show that the approach is promising and the binarization rate is close to 100%.

5. Conclusions

In this paper, a binarization approach based on star-shape filter for wafer ID images is proposed. The proposed approach can effectively enhance the performance of wafer ID recognition. It enhances the wafer ID image at first and original binarization based on star-shape filter and high-low score comparison method are applied in the following. Finally, final binarization result is obtained by combining the determination of the scribe center. Experimental results presented in the paper show that the algorithm provides good results even in situations of uneven luminance, overexposure and underexposure. The experiment results show that the proposed method surpasses Otsu's method and Bernsen's method obviously. As a result, we believe that the proposed technique is an attractive alternative to currently available methods for binarizing wafer ID images. We may apply this technique to other applications, such as car license-plate recognition and the container ID numbers recognition.

References

- [1] S. H. Shaikh, A. Maiti, and N. Chaki, "Image binarization using iterative partitioning: A global thresholding approach," in *Recent Trends in Information Systems (ReTIS)*, 2011 International Conference, 2011, pp. 281-286.
- [2] V. Sokratis, E. Kavallieratou, R. Paredes, and K. Sotiropoulos, "A hybrid binarization technique for document images," *Learning Structure and Schemas from Documents*, vol. 3163, pp. 165-179, 2011.
- [3] S. Cunzhao, X. Baihua, W. Chunheng, and Z. Yang, "Adaptive Graph Cut Based Binarization of Video Text Images," in *Document Analysis Systems (DAS)*, 2012 10th IAPR International Workshop, 2012, pp. 58-62.
- [4] J. Shi, N. Ray, and H. Zhang, "Shape based local thresholding for binarization of document images," *Pattern Recognition Letters*, vol. 33, pp. 24-32, 2012.

- [5] H. Chen and R. Gururajan, "Otsu's Threshold Selection Method Applied in De-noising Heart Sound of the Digital Stethoscope Record," in *Advances in Information Technology and Industry Applications*, 2012, pp. 239-244.
- [6] B. Singh, V. Chand, A. Mittal, and D. Ghosh, "A Comparative Study of Different Approaches of Noise Removal for Document Images," in *Proceedings of the International Conference on Soft Computing for Problem Solving (SOCPROS 2011)*, 2012, pp. 847-854.
- [7] P. Stathis, E. Kavallieratou, and N. Papamarkos, "An evaluation survey of binarization algorithms on historical documents," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference, 2008*, pp. 1-4.
- [8] F. Chun Che and R. Chamchong, "A Review of Evaluation of Optimal Binarization Technique for Character Segmentation in Historical Manuscripts," in *Knowledge Discovery and Data Mining, 2010. WKDD '10. Third International Conference on*, 2010, pp. 236-240.
- [9] M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *Journal of Electronic Imaging*, vol. 13, pp. 146-168, 2004.
- [10] T. R. Singh, S. Roy, O. I. Singh, T. Sinam, and K. Singh, "A New Local Adaptive Thresholding Technique in Binarization," *International Journal of Computer Science Issues*, vol. 8, pp. 271-277, 2011.
- [11] X. Chen, K. Wang, and Q. Wang, "An adaptive binarization method for camera based document image," *Software Engineering and Knowledge Engineering: Theory and Practice*, vol. 162, pp. 677-684, 2012.

