Center detection algorithm for printed circuit board circular marks based on image space and parameter space

Min Qi,[a,*] Yanan Wang,[a] Yanshuo Chen,[a] Hongjuan Xin,[a] Yuelei Xu,[b] Hongying Meng,[c] and Aili Wang[a]

[a]Northwestern Polytechnical University, School of Electronics and Information, Xi'an, China
[b]Northwestern Polytechnical University, Unmanned System Research Institute, Xi'an, China
[c]Brunel University London, Department of Electronic and Computer Engineering, Uxbridge, United Kingdom

Abstract. A highly efficient circle positioning algorithm, called the two-step optimization Hough transform (TSHT), based on multi-resolution segmentation is proposed to solve the problems of the offset Hough transform, namely, its large memory overhead, long time consumption, and low recognition accuracy. First, using the image feature of the printed circuit board (PCB) circular identifier, the target circle is obtained using adaptive image preprocessing, and then, images of an acceptable quality are separated by shape quality inspection to improve their robustness. Second, using effective interval sampling strategies and gradually controlling the accumulative interval of parameters, the TSHT algorithm reduces the memory overhead and quickly locates the center at the pixel level. Finally, the center at the sub-pixel level is found by the least-squares method for circle fitting. The experiments prove that TSHT, as a result of its high robustness, strong anti-noise capability, fast recognition speed, and accuracy, can be successfully applied to a vision positioning system of a solder paste printing machine. © *2022 SPIE and IS&T* [DOI: 10.1117/1.JEI.32.1.011002]

Keywords: two-step optimization Hough transform; adaptive image preprocessing; shape quality inspection; least-squares method.

Paper 210315SS received Jun. 9, 2021; accepted for publication Oct. 19, 2021; published online Mar. 22, 2022.

1 Introduction

The decrease in device packaging sizes and increase in PCB patch density have led to the high precision detection of the PCB circular identifier becoming the core of the machine vision positioning system.[1,2] At present, the center of the circle is most commonly detected with the Hough transform (HT).[3] Briefly, the image space is transformed into the parameter space, and some parameters are used to describe the edge curve. This method shows good robustness and produces favourable detection results, even under the conditions of noise, defects, and deformation. However, when the parameter space exceeds two dimensions, the computing time and storage requirement increase dramatically, making the method inefficient in circle detection. The HT is a method for the detection clean lines in its most basic form, although it can also be used it to recognize circles and ellipses. The approach is based on the assumption that the edge has been spotted, and it is resistant to interference and lacking points. By selecting in the Hough dimensional space and afterward choosing local maxima in an accumulation matrix, the circular contenders are created.[4]

Randomized Hough transform (RHT)[5,6] solves this problem via sampling points randomly in the edge image and establishing a link-list data structure for the parameters but also has difficulty detecting multi-circle complex images. Improvements can be made by reducing sample points or invalid accumulations in the parameter space. Scientists promote randomized circle detection (RCD) algorithm,[7] which randomly samples four edge pixels—three are used to construct a

*Address all correspondence to Min Qi, drqimin@nwpu.edu.cn

1017-9909/2022/$28.00 © 2022 SPIE and IS&T

possible circle and the remaining to confirm whether the possible circle can be promoted to a candidate circle, which effects cases involving multiple circles but only modestly affect single circles. The RHT differs from of the HT because it uses the geometrical features of analysis curvatures to avert to conduct the computational cost election system for each and every non-negative pixel of an image, improving the period efficiency and reducing the storage overhead of the existing algorithm.[8]

Xia et al.[9] put forward a new algorithm, called the offset Hough transform (OHT). This obtains the mapping coordinate set of all edge points by translating the mapping coordinate set of the datum point in the parameter space only using HT,[10] which greatly improves the speed of single target detection but not the recognition accuracy. Objects that are coherent in the Euclidean geometry input images are clearly obvious when examined in Hough parameter space because they give curves that meet at a single point. The Hough parameter field is quantized into limited periods or accumulation units to apply the transform.[11]

In addition, by combining image preprocessing and the final voting process, relevant scholars have proposed numerous improved algorithms based on specific application backgrounds. For example, Liang et al.[12] proposed a novel angle-aided circle detection algorithm based on the randomized HT, which ameliorated the sampling method of random sampling points to reduce the invalid accumulation. Xie et al.[13] fully considers the eight-neighborhood pixel information and keeps the Roberts operator's advantages of high location accuracy and fast speed to improve sub-pixel edge detection algorithm. Kumar et al.[14] presents a circle HT structure for accurate iris localisation. De Melo et al.[15] processes images through the circle HT (CHT) method to count arbuscular mycorrhizal fungi automatically with neural networks. The CHT is a fundamental extracting features approach for detecting circle in defective pictures in image processing. Through voting in the Hough dimensional space and afterward choosing local maxima in an aggregate matrix, the circular contenders are created.[16]

This paper will present TSHT, an algorithm based on multi-resolution segmentation and combining image preprocessing with the OHT algorithm. After image preprocessing and two-step optimization HT (TSHT), we will find the center of the PCB circular identifier.

2 Methodology

2.1 *OHT Algorithm*

2.1.1 *Principle of the OHT algorithm*

For a circle, one edge point in the image space can be mapped onto a circle in the parameter space by the Hough transformation, and the corresponding circles of all edge points in the parameter space must intersect at one point, which is the center of the circle in the image space. While implementing the OHT algorithm, all corresponding circles can be obtained by translating the circle of the datum point in the parameter space to reduce the time consumption of the HT point by point. A datum point is a documented reference point that can be used to do measurements or analyses. The mark could be based on current place's speed or a reference. The steps can be summarized as follows:

Step 1: Take point $(x_{\text{base}}, y_{\text{base}})$ as the datum point and R as the radius of the original image. The mapping coordinate set of the datum point in the parameter space is F_{base} for the traditional HT.

Step 2: Find the first edge point (x_1, y_1). Then, obtain the mapping coordinate set of (x_1, y_1), named F_1, by adding the offset $(x_1 - x_{\text{base}})$ in the x direction and $(y_1 - y_{\text{base}})$ in the y direction, respectively). At last, all corresponding accumulative values of the elements of F_1 are accumulated in the parameter space.

Step 3: Repeat Step 2 and calculate F_n $(n \geq 2)$. Then, accumulate all corresponding accumulative values of the elements of F_n. It is important to ensure that all of the edge points are involved in the calculation.

Step 4: Obtain the accumulative value of the maximum in the parameter space, which is the corresponding center coordinate of the circle in the image space.

2.1.2 *Performance analysis of OHT*

1. It is assumed that the circle image is of N row and M column, and the OHT algorithm needs $M \times N$ scan cycles to determine the number of edge points that participate in the transformation. For convenience, the total number of edge points is recorded as Q. To obtain the mapping coordinate set of some points, F_{base} is translated and the translational coordinates are accumulated. At this point, the calculation area for every point in the parameter space is $2R \times 2R$, that is to say, it is a triple loop of $2R \times 2R \times Q$ for Q edge points. Obviously, this consumes a very large amount of space and time and seriously affects the efficiency of the algorithm.
2. Due to the discretization of coordinates in digital image, the translational mapping coordinates are not completely the same as the calculational coordinates. Moreover, the edge of the circle may appear burred, dislocated, defected, or polluted. Taking this into consideration, the accumulative value in the parameter space will generate peak diffusion, and the real peak will disappear in the noise peak. All of this can lead to a deviation in the calculation results from the true value.

2.2 *Basic Idea of TSHT*

The large amount of edge point transformations involved in Hough algorithms is quite time consuming. Although OHT accelerates the speed of calculation greatly through the use of a single Hough transformation of the datum point, there are still two limitations. The first is that quite a lot of times of accumulative calculations are needed in the parameter space. Second, most of existing improved algorithms are based directly on the edge points, whereas the raw data are images with noise in practical engineering applications. Thus, the question how to quickly acquire edge points from the original image data has real significance.

In this paper, the TSHT algorithm is proposed to reduce the limitations. First, the target circle is obtained using adaptive image preprocessing to locate the effective search area. The method employs image identification as a fundamental component to allow the implementation of customizable picture pre-processing algorithms, boost the system's versatility, and allow self-adapting behaviors. To validate and verify the study's validity, real photos from a washer manufacturing line are shown. Preprocessing has the potential to break down current industrial inflexible supervisory approaches and produce systems that can cope to problem continues and time-varying factors.[17] Then, high quality images are separated by shape quality inspection to be used as accurate data sources. In the process of center positioning, the first-level HT quickly obtains the center interval by heavily reducing the number of points involved in the transformation in the image space and the number of invalid accumulations in the parameter space. It is worth mentioning that this measure can guarantee that the number of calculations is far $<2R \times 2R \times Q$. The second-level HT rapidly determines the center position at the pixel level by further restricting the accumulative range. Using the two-level HT, the center of the circle is detected from an approximative level to an accurate level. At last, least-squares fitting for the circle is adopted to upgrade the positioning accuracy to the sub-pixel level and, at the same time, reduce the influences of burr, dislocation, defects, and pollution. The least-squares method is a scientific procedure for deciding the best fit for a group of data locations by reducing the total of the spots' deviations or royalties from of the plotted curve. To forecast the movement of response variable, least-squares analysis is employed.

2.3 *Image Preprocessing of TSHT*

It is easy to extract edge points of an acceptable quality with the image feature of the PCB circular identifier in only three steps: adaptive binarization processing, searching for the largest area, and shape quality detection. The adaptive binarization system is preferable for real-time computing and can be used on portable devices. It is also the process that turns a grayscale to black-and-white, effectively lowering the amount of knowledge image content from varying shades. When attempting to identify an item from such a picture, this is a frequent operation.[18]

Step 1: In adaptive binarization processing, it is assumed that the image (I_{org}) is in the N rows and M columns, and the gray value ranges from 0 to 255. The calculation is as follows: let $I_{org} = g(x, y)$, where $0 \le x \le M - 1$ and $0 \le y \le N - 1$. The whole image is scanned from left to right and bottom to top, and the average gray value ave is calculated with

$$ave = \frac{1}{MN} \sum_{y=1}^{N} \sum_{x=1}^{M} g(x, y). \tag{1}$$

By taking ave as the reference value, the number of pixels whose gray value is greater than ave is computed, and then the pixel-value difference $det\,ave$ between the acceptable pixel gray value and ave is determined. $meanave$ denotes the average value of $det\,ave$ of all acceptable pixels. Next, the average gray value ave_0 in the range of ave and 255, which can be explained as $ave_0 = ave + meanave$, is determined, so at this point, the threshold range is from ave_0 to 255. The circular identifier gray peak is set to $index$, considering the image boundary transition information, and the final threshold selected is T_b, which represents the weighted sum of $index$ and ave_0.

$$T_b = \frac{m_1}{m_2} ave_0 + \left(1 - \frac{m_1}{m_2}\right) index, \tag{2}$$

where m_1 is the number of pixels whose gray value is ave_0, and m_2 is the number of pixels whose gray value is index.

Step 2: To search for the largest area, the binary image I_{bi} is obtained using threshold T_b from Step 1. Image binarization is used to turn the target pixels black and the background pixels white. The black area contains both the target circle and the noise, with an irregular small black connected area. The target circle has the largest connected area in the actual industrial image, which means it can be easily found by searching the connected area in the binary image I_{bi}. Finally, the gray values of all pixels included in the target area are set to 0 and the background pixels are set to 255, to eliminate the internal small holes for the purpose of extracting the target area outline easily and accurately by edge detection method.

Step 3: Shape quality inspection is performed with the following calculation: let V denote the set of all edge points in I_{bi}, which can be obtained by contour extraction,

$$V = \{(x_{vi}, y_{vi}) | i = 1,2, \cdots Q\}. \tag{3}$$

The width range W and height range H of the circle circumscribed rectangle can be easily inferred. W is also named Δx_v, which is equal to $(x_{v\_\max} - x_{v\_\min})$, and H is also named Δy_v, which is equal to $(y_{v\_\max} - y_{v\_\min})$. Among the aforementioned parameters, $x_{v\_\max}$ and $x_{v\_\min}$ represent the maximum and minimum coordinates in the X direction, respectively. The same goes for $y_{v\_\max}, y_{v\_\min}$, which represent the maximum and minimum coordinates in the Y direction, respectively.

1. Out of W and H, the larger value is named $d_{\max}$, and the smaller, $d_{\min}$. Then, Δd is set as $(d_{\max} - d_{\min})$. For convenience, $D = \Delta d / d_{\min}$ is taken as the difference index for vertical or horizontal difference of the circle, and T_d is the threshold of the difference index. If $D > T_d$, the calculation is terminated due to excessive defects or protrusions, and the result is shown in Fig. 1(a). Otherwise, it does not affect the subsequent center location because the circle identifier area has an acceptable level of defecting or protruding. While, the difference index will fail to exclude the deformations with the same amount of differences both in vertical and horizontal directions, caused by defects or pollution as shown in Fig. 1(b), a matter that will be further discussed in the next step.
2. All elements (x_{vi}, y_{vi}) of the set V are traversed, and then the center of the circular identifier contour is calculated as follows:

$$V_{avex} = \frac{1}{Q} \sum_{i=1}^{Q} x_{vi}, \quad V_{avey} = \frac{1}{Q} \sum_{i=1}^{Q} y_{vi}. \tag{4}$$

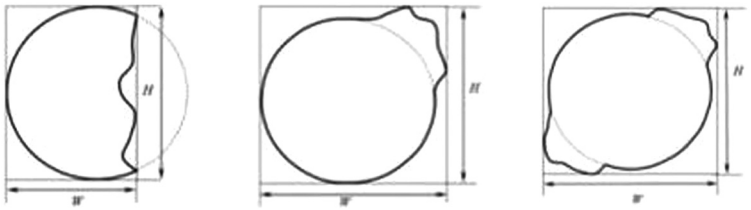

Fig. 1 Unqualified circle contour shape. (a) Image with $D > T_d$; (b) Image with $D \leq T_d$; (c) Image with $G \leq T_g$.

The center of the circle circumscribed rectangle is calculated as follows:

$$F_{avex} = \frac{(x_{v\_\min} + x_{v\_\max})}{2}, \quad F_{avey} = \frac{(y_{v\_\min} + y_{v\_\max})}{2}. \tag{5}$$

Thus, the difference between the contour center position and circumscribed rectangle center position can be expressed as

$$\Delta g = \min\{|V_{avex} - F_{avex}|, |V_{avey} - F_{avey}|\}. \tag{6}$$

$G = \Delta g / d_{\min}$ is taken to be the one-way equivalent deformation index, and T_g is the one-way equivalent deformation index threshold. If $G > T_g$, this indicates that the circle has severe deformation in the one-way vertical or horizontal direction. Therefore, the calculation must be terminated. Otherwise, the subsequent detection of the center cannot be affected by the equivalent deformation in the one-way vertical or horizontal direction. However, horizontal and vertical equivalent deformation can occur simultaneously in two opposite directions, as shown in Fig. 1(c). This cannot be ruled out by merely using G, so a further step is required.

3. The roundness parameter C[19] is calculated using equation $C = 4\pi \cdot s/L^2$, where S is the number of pixels within the circular area, and L is the number of pixels of the circular perimeter in eight neighborhoods. Ideally, C of the standard circle is 1, but it is usually about 0.9 due to the discretization of the image. Furthermore, the circle becomes closer to the standard as C increases, and the maximum is still 1. After calculating the D and G parameters, the roundness parameters can be used to effectively detect whether horizontal and vertical equivalent deformation are occurring simultaneously in two opposite directions. T_c is the roundness threshold; if $C < T_c$, then the process should move to the next new image. Otherwise, the situation is acceptable for the subsequent center location.

In Step 3, the vertical and horizontal difference index D, the one-way equivalent deformation index G, and the roundness C are used to detect the quality of the circle contour to acquire accurate data for the subsequent TSHT algorithm.

2.4 *TSHT Algorithm for Detecting the Circle Center*

Combined with image preprocessing, the THSH algorithm can effectively calculate the center of the circle. The specific steps are as follows:

Step 1: To describe the symbols conveniently, one point $(x_{\text{base}}, y_{\text{base}})$ is defined randomly in set V as the datum point, and r is defined as the radius. In fact, the inner and outer diameters are represented by r_{in} and r_{out}, respectively, where $r_{\text{in}} = d_{\min}/2$, $r_{\text{out}} = d_{\max}/2$. The mapping coordinate set of datum points in the parameter space is set as F_{base}. Finally, a single HT is conducted on the datum point under the conditions of r_{in} and r_{out}, and F_{base}, as follows:

$$F_{\text{base}} = \{(a,b)|(a - x_{\text{base}})^2 + (b - y_{\text{base}})^2 = r, r_{\text{in}} < r < r_{\text{out}}\}. \tag{7}$$

where x_{base} ranges in $[r_{\text{out}}, M - r_{\text{out}}]$ and y_{base} ranges in $[r_{\text{out}}, N - r_{\text{out}}]$. (a, b) is the mapping coordinate of the transformed circle in the parameter space.

Step 2: Set V is sampled with a 3-pixels interval, and the set of sampling points is defined as V', where $V' = \{(x'_{vj}, y'_{vj})|j = 1,2, \cdots Q'\}$. Q' is the number of sampling points. It is necessary to add that only sampling points are involved in the OHT.

Step 3: According to OHT algorithm, the mapping coordinate of (x'_{vj}, y'_{vj}) after the HT is named F_j:

$$F_j = \{(a_j, b_j)|a_j = a + (x'_{vj} - x_{\text{base}}), b_j = (y'_{vj} - y_{\text{base}})\}. \tag{8}$$

The term (a_j, b_j) in Eq. (8) is the mapping coordinate of the transformed circle in the parameter space.

Step 4: In fact, the circle center must be located in the circle-inscribed rectangle, so the accumulative space in the parameter space is restricted as follows:

$$x_{v\_\min} + \frac{\sqrt{2} * r_{\text{out}}}{2} < a_j < x_{v\_\max} - \frac{\sqrt{2} * r_{\text{out}}}{2}, \tag{9}$$

$$y_{v\_\min} + \frac{\sqrt{2} * r_{\text{out}}}{2} < b_j < y_{v\_\max} - \frac{\sqrt{2} * r_{\text{out}}}{2}. \tag{10}$$

Step 5: An accumulator array $(A_1(a_j, b_j))$ is established and, initially, all elements are set to 0. Then, the coordinates of F_j can be accumulated in the three-pixel interval. At the same time, to speed up the accumulation, the cumulative range is limited in Eqs. (9) and (10). At last, the center position $(x_{\text{cent}}, y_{\text{cent}})$ is obtained through the first-level HT. Obviously, $(x_{\text{cent}}, y_{\text{cent}})$ represents the center position with several pixels of error, so the real center position can be conclusively determined to be in small areas of m × m. m is assumed to be equal to 5. Hence, the expression of a_j and b_j is as follows:

$$x_{\text{cent}} - 5 \le a_j \le x_{\text{cent}} + 5, \tag{11}$$

$$y_{\text{cent}} - 5 \le b_j \le y_{\text{cent}} + 5. \tag{12}$$

Step 6: Sample set V pixel by pixel. Repeat Step 2 and Step 3. The mapping coordinate of the edge points after the HT in the parameter space is named F_i.

Step 7: The coordinates of F_i are accumulated, pixel-by-pixel. In addition, to improve the speed, the cumulative range is limited in Eqs. (11) and (12). The real center position $(x_{\text{center}}, y_{\text{center}})$ is acquired via the second-level HT.

Step 8: The distance r_{vi} from every point $(x_{\text{vi}}, y_{\text{vi}})$ to the real center $(x_{\text{center}}, y_{\text{center}})$ is calculated. The reliable radius R is the arithmetic average of r_{vi}. This is shown as follows:

$$R = \frac{1}{Q}\sum r_{vi} = \frac{1}{Q}\sum \sqrt{(x_{vi} - x_{\text{cent}})^2 + (y_{vi} - y_{\text{cent}})^2}. \tag{13}$$

Step 9: T_N denotes the denoising threshold for the boundary. If the absolute value of $(r_{vi} - R)$ is $>T_N$, it should be the noise point and can be discarded. Otherwise, the point should be retained.

Step 10: After least-squares fitting using the edge points after denoising processing, the center of the circular identifier at the sub-pixel level is determined.

At this point, a coarse-to-fine circle center detection algorithm is realized. The specific process of the algorithm is shown in Fig. 2.

2.5 *Performance Analysis of the TSHT Algorithm*

On the whole, TSHT is superior to the OHT algorithm in terms of time consumption, space complexity and accuracy.

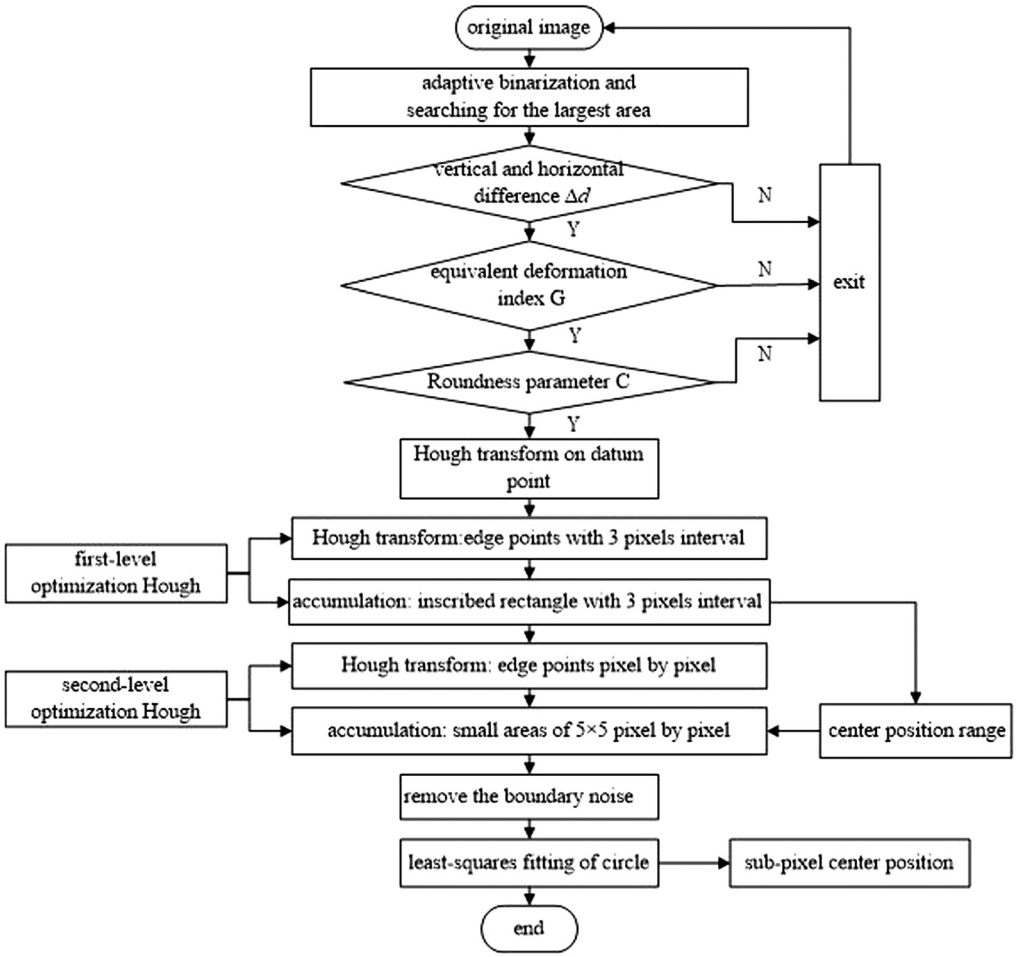

Fig. 2 Implementation of the TSHT algorithm.

2.5.1 *Low time and space complexity*

During the process of center positioning, the first-level Hough transformation samples the edge pixels, with a 3-pixels interval in the image space and accumulates the coordinates of the transformed circles with a 3-pixels interval, also in the parameter space. At the same time, the accumulative space is restricted to the inscribed rectangle. From this, it can be inferred that its computation amount is approximately a triple loop of $\frac{\sqrt{2}R}{3} \times \frac{\sqrt{2}R}{3} \times Q$. Compared with the OHT algorithm of $2R \times 2R \times Q$, the number of calculations required is decreased by 53 times, thus, dramatically reducing the memory and time overhead. Based on the first-level Hough transformation, the second-level HT is equal to a triple loop, with $11 \times 11 \times Q$. Clearly, this requires very little computation. Furthermore, the time complexity of the least-squares circle fitting is negligible, so it will not affect the time consumption of the algorithm. Therefore, the TSHT algorithm significantly improves the efficiency of circle recognition.

2.5.2 *Sub-pixel locating accuracy*

The improved algorithm reduces the number of calculations required by 53 times, at the expense of a 3-pixel error in the first-level HT by sampling. Then, the second-level HT achieves an accurate center at the pixel level, with little time consumption. Finally, the least-squares fitting method is adopted to upgrade the positioning accuracy to the sub-pixel level.[20] In general, the TSHT algorithm avoids discrete coordinates and peak diffusion, greatly enhancing the credibility of the test results.

2.5.3 *Strong robustness*

TSHT algorithm excludes the problematic image before center detection by shape quality inspection and target image segmentation, which strongly increases the robustness and makes it applicable to all of the series of PCB images in actual industrial production.

3 Results and Discussion

In this section, the results from five experiments are provided to validate the high speed and resolution of the proposed algorithm in different aspects of the rationality of shape quality inspection, the speed and accuracy of the algorithm, the ratio of robustness to noise, robustness to the non-standard circle, and the performance of the real images. The first four experiments were conducted with synthetic images. Synthetic imaging is the method of creating two-dimensional diffraction patterns through computational mathematics calculations of accumulated data rather than the more tool has the ability approach of focusing light rays through camera or even other optical elements. In the backup system, the physical image is essentially stored digitally. The synthetic image is a two-dimensional array that is mostly applied for picture creation. Depending on various forms of signals in the computer animation, the primary arrays value of the kind of cover image is assessed and calculated. All of these experiments were performed on an Intel CPUAMD Athlon(tm) II X2 250 Processor with 3.0 GHz and 2 GB RAM. The operating system adopted was MS-Windows XP, and the programming environment was C++ Builder 6.0.

3.1 *Rationality of Shape Quality Inspection*

The experiments used synthetic images with dimensions of 1280×960. The center was set as (500, 500). R was used to denote the radius, and ideal circles were generated based on the midpoint circle algorithm. Next, the ideal circles were changed artificially to simulate burring, dislocations, defects, and pollution. To reflect the adaptability of TSHT for different sizes, R values of 100, 150, 200, and 250 pixels were used. For each circle of a certain radius, different shape changes were made according to the different vertical and horizontal difference indexes D, the one-way equivalent deformation index G, and the roundness parameter C. We carried out 50 experiments for each radius, 200 experiments in total. The proposed algorithm was used to detect the center, and the parameters of D, G, and C were measured in every experiment. It was found that, when $T_d = 1/4$, $T_g = 1/5$, $T_c = 0.7$, the center identification error was less than 1 pixel, which meets the demand for high-precision positioning. In total, the qualified data included 153 of the 200 groups.

Due to space limitations, we only provide 12 sets of experimental data and their corresponding images, and it is necessary to explain that the images above were of an acceptable quality and all had a radius of 150 pixels. Empirical data are recorded in Table 1, where Δg_x and Δg_y represent the difference in the center position between the circle contour and the circumscribed rectangle in the x and y directions, respectively, and Δa and Δb are the deviation values between the calculated center and standard center, respectively. All parameters mentioned in this section are in pixels. The images that correspond with the data in Table 1 are shown in Fig. 3; specifically, the first four images correspond to sequences 1–4 in Table 1, and so on.

3.2 *Speed and Accuracy of the TSHT Algorithm*

Four different algorithms were carried out to validate the performance of the TSHT algorithm in terms of speed and accuracy, as compared with the RHT (in Ref. 5), RCD (in Ref. 6), and OHT (in Ref. 9). As previously mentioned, the dimensions of the synthetic images were 1280×960, and the center of each image was at (500, 500). The minimum radius was 50 pixels, and the maximum radius was 400 pixels. It was assumed that the radius R was the interval of 10 pixels, so 36 circles of different sizes could be acquired. Finally, a total of 36 tests were developed. The experimental data are shown in Table 2, where t means the execution time (in seconds),

Table 1 Shape quality detection ($R = 150$).

Image	Δd	D	$(\Delta g_x, \Delta g_y)$	P	C	$(\Delta a, \Delta b)$
1	0	0.000	(0, 0)	0.000	0.92	(0.000, 0.000)
2	4	0.013	(7, 62)	0.195	0.78	(0.002, 0.023)
3	44	0.020	(31, 0)	0.105	0.72	(0.06, 0.078)
4	38	0.086	(2, 37)	0.134	0.70	(0.686, 0.045)
5	24	0.086	(3, 27)	0.090	0.82	(0.003, 0.105)
6	0	0.000	(21, 9)	0.070	0.83	(0.023, 0.006)
7	10	0.034	(26, 17)	0.090	0.80	(0.028, 0.037)
8	6	0.020	(47, 0)	0.159	0.71	(0.016, 0.000)
9	58	0.239	(4, 27)	0.115	0.77	(0.000, 0.406)
10	40	0.153	(5, 30)	0.115	0.72	(0.148, 0.176)
11	2	0.006	(48, 44)	0.161	0.70	(0.094, 0.842)
12	0	0.000	(8, 4)	0.027	0.83	(0.416, 0.038)

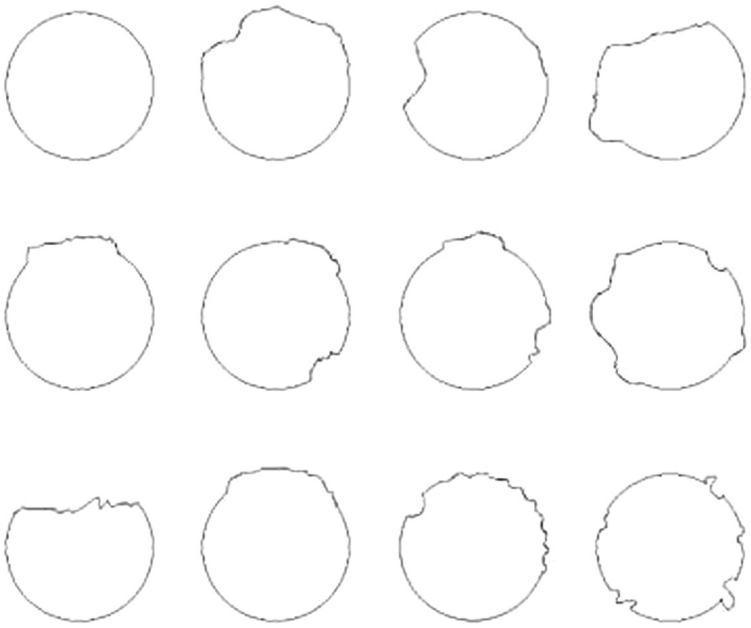

Fig. 3 Non-standard circle with a radius of 150 pixels.

and Δa and Δb are the deviation values of the calculated center and standard center, respectively. Because of the limited space, eight groups of experimental data are presented in Table 2.

As can be seen in Table 2: (1) considering time consumption only, the detection time of the improved TSHT algorithm was stable, at around 200 ms., which significantly reduced the time consumption compared to other algorithms. (2) Considering the identification accuracy only, the results of the RHT algorithm were limited by the point-to-point distance and the number of sampling points. The OHT and TSHT algorithms both showed zero deviation in the process of the standard circle. Overall, the TSHT algorithm significantly reduced the time consumption to an acceptable range and maintained an extremely high level of accuracy.

Table 2 Time and accuracy performance of the four algorithms.

Image sequence	Radius R	RHT		RCD		OHT		TSHT	
		$(\Delta a, \Delta b)$	t	$(\Delta a, \Delta b)$	t	$(\Delta a, \Delta b)$	t	$(\Delta a, \Delta b)$	t
1	50	(1, 2)	0.032	(2,1)	0.022	(0, 0)	0.140	(0, 0)	0.171
2	100	(0, 0)	0.031	(0,1)	0.021	(0, 0)	0.218	(0, 0)	0.172
3	150	(2, 1)	0.016	(0,2)	0.018	(0, 0)	0.250	(0, 0)	0.172
4	200	(1, 1)	0.031	(2,2)	0.015	(0, 0)	0.562	(0, 0)	0.187
5	250	(0, 1)	0.031	(0,3)	0.015	(0, 0)	0.812	(0, 0)	0.187
6	300	(1, 2)	0.032	(2,1)	0.016	(0, 0)	0.656	(0, 0)	0.187
7	350	(2, 1)	0.031	(0,1)	0.015	(0, 0)	0.984	(0, 0)	0.188
8	400	(0, 3)	0.031	(4,1)	0.016	(0, 0)	1.375	(0, 0)	0.203

3.3 *Robustness to Noise Ratio*

Considering disturbance, different levels of Gauss noise, expressed by the signal-to-noise ratio (SNR), were added into the synthetic images. The experimental results are shown in Fig. 4. Δs represents the deviation between the standard value and calculated value, which is expressed as $\Delta s = \sqrt{\Delta a^2 + \Delta b^2}$. By fully considering the diversity of samples to ensure the credibility of the data, each datum in Fig. 4 uses the average value of Δs of 36 different circle sizes from Experiment 2.

We can draw four conclusions from Fig. 4: (1) With an increase in noise density (decrease in SNR), the recognition accuracy of the four algorithms decreased to varying degrees. (2) The noise had a small effect on the accuracy, compared with the results shown in Table 2, mainly due to the edge accuracy, which indicates that the precision of the algorithm is barely affected by noise. (3) Most of the Δs shown to be 0 proved OHT good ability of resisting noise, but it didn't mean OHT is better than TSHT for not reflecting the tiny changes at the sub-pixel level. (4) The TSHT algorithm produces a maximum error of <0.3 pixels, even in a strong noise environment (SNR = 4), which is sufficient to prove a good robustness to noise ratio.

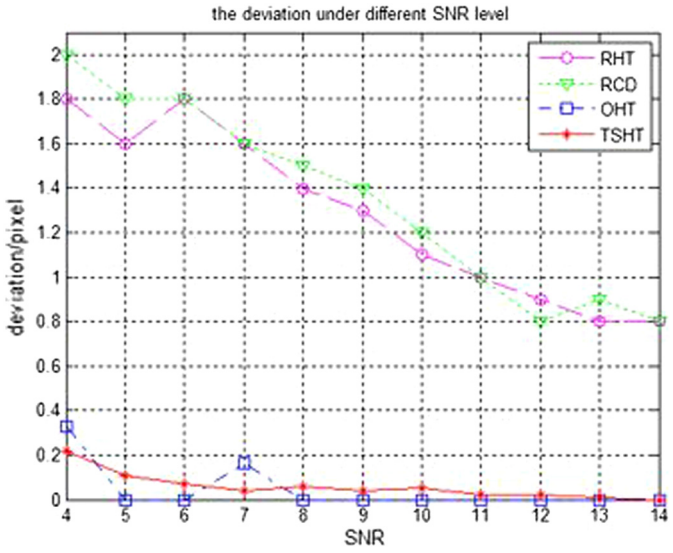

Fig. 4 The deviation under different SNR levels.

Table 3 Non-standard circle recognition ($R = 150$).

Figure sequence	RHT		RCD		OHT		TSHT	
	$(\Delta a, \Delta b)$	t	$(\Delta a, \Delta b)$	t	$(\Delta a, \Delta b)$	t	$(\Delta a, \Delta b)$	t
1	(0, 1)	0.031	(0, 1)	0.022	(0, 0)	0.250	(0.000, 0.000)	0.172
2	(3, 4)	0.032	(3, 4)	0.020	(0, 2)	0.656	(0.002, 0.023)	0.170
3	(0, 1)	0.033	(1, 1)	0.020	(1, 2)	0.828	(0.060, 0.078)	0.172
4	(1, 4)	0.016	(1, 3)	0.020	(0, 2)	2.234	(0.686, 0.045)	0.172
5	(0, 2)	0.032	(0, 1)	0.016	(0, 2)	2.265	(0.003, 0.105)	0.175
6	(0, 0)	0.032	(0, 1)	0.018	(0, 0)	0.265	(0.023, 0.006)	0.172
7	(2, 2)	0.031	(3, 2)	0.017	(5, 5)	1.281	(0.028, 0.037)	0.172
8	(6, 1)	0.032	(5, 2)	0.022	(2, 0)	0.842	(0.016, 0.000)	0.165
9	(0, 1)	0.032	(2, 1)	0.018	(12, 1)	1.242	(0.000, 0.406)	0.168
10	(2, 5)	0.031	(2, 5)	0.019	(2, 2)	0.542	(0.148, 0.176)	0.172
11	(1, 1)	0.031	(2, 1)	0.016	(0, 0)	0.390	(0.094, 0.842)	0.172
12	(4, 1)	0.031	(4, 2)	0.016	(0, 0)	0.250	(0.416, 0.038)	0.178
Average	(1.58, 1.92)	0.030	(1.91, 2.00)	0.018	(1.83, 1.33)	0.920	(0.12, 0.15)	0.172

3.4 *Robustness to Non-Standard Circle Ratio*

The high speed and precision of the TSHT algorithm was verified theoretically in Experiment 2.

Now, the identification results of the non-standard circle are used to prove the practicability of the algorithm. Here, 153 non-standard circles that met the quality requirements were chosen from Experiment 1 to be tested. At the same time, the Gauss noise was set as SNR = 12. There is insufficient space to cover all of them here, but 12 images are provided, which are shown in Fig. 3. The results are shown in Table 3.

The 12 data sets in Table 3 basically reflect the experimental results of the 153 non-standard circles. It can be seen in Table 3 that almost each datum, calculated by the RHT, RCD, and OHT algorithms, produced error, and the maximum average errors were 1.58, 2, and 1.83 pixels, respectively; these values are insufficient for use in practical applications. Nevertheless, the proposed TSHT algorithm achieved an error rate of 0.2 pixels for non-standard circles, which meets the quality requirements with sufficient recognition speed.

3.5 *Actual Circle Identifier Image Test*

To verify the detection effect for actual images rather than simulated ones, we take 100 images from a CCD industrial camera with a resolution of 1280×960 were tested. The proposed TSHT algorithm was adopted to illustrate its performance, as shown in Fig. 5. The proposed algorithm obtained correct results for the 100 randomly captured images. In terms of time consumption,

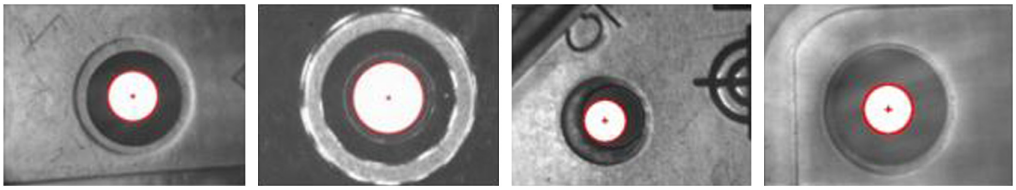

Fig. 5 Actual circle identifier images.

the image processing took <200 ms, and the TSHT also consumed less than 200 ms, so the total execution time was far <0.4 s. Therefore, we have strong reason to believe that the proposed algorithm can satisfy actual production requirements.

4 Conclusions

Through combing image pre-processing with OHT algorithm, this paper presented the new proposed Two-step optimization HT based on multi-resolution segmentation to improve the execution-time performance as well as the detection accuracy of OHT algorithm. Two innovations were developed and explained clearly. The first innovation was adaptive pre-processing and shape quality detection, which can offer an acceptable circle and research area divided from the original image while strongly increasing the robustness. The second innovation was the use of effective interval sampling strategies in the image space and gradually controlling the accumulative interval in the parameter space through the two-level HT to reduce the memory overhead and quickly locate the circle centre at the pixel level. Finally, least-squares fitting was adopted to upgrade the positioning accuracy of the circle centre to the sub-pixel level. Based on the test images, the experimental results shown in this paper demonstrated that the improved algorithm, with its high robustness, strong anti-noise ability, fast recognition speed, and high accuracy, significantly improved circle detection algorithm. Consequently, it is a suitable high-speed and high-precision identification system.

Acknowledgments

The authors thank all the members of the research group for the technical support provided during the research activities. **Funding:** This research was sponsored by the Key Project of Shaanxi Province Industrial Innovation Program, China (Grant No. 2017ZDCXL-GY-11-02-02).

References

1. Z. Liu et al., "Positioning of circular mark in PCB based on PCA and segment RHT," *J. Chongqing Univ. Technol. (Nat. Sci.)* **31**(01), 93–99 (2017).
2. C.-S. Liu et al., "PCB clamping positioning deviation correction based on computer vision technology," *Mech. Electr. Eng. Technol.* **48**(7), 107–110 (2019).
3. P. Mukhopadhyay and B. B. Chaudhuri, "A survey of Hough transform," *Pattern Recognit.* **48**(3), 993–1010 (2015).
4. M. Abdel-Basset et al., "2-levels of clustering strategy to detect and locate copy-move forgery in digital images," *Multimedia Tools Appl.* **79**(7-8), 5419–5437 (2018).
5. Y. B. He et al., "Image quality enhanced recognition of laser cavity based on improved random Hough transform," *J. Vis. Commun. Image Represent.* **64**, 102679 (2019).
6. P. Yuan et al., "Random circle detection and location technology based on improved RCD algorithm," *J. Hunan Inst. Eng. (Nat. Sci. Ed.)* **27**(01), 49–53 (2017).
7. J.-K. Liu and Q. Fan, "An improved randomized circle detection algorithm using in printed circuit board locating mark," *Appl. Math.* **10**(10), 848–861 (2019).
8. R. Gupta et al., "Fingerprint image enhancement and reconstruction using the orientation and phase reconstruction," *Inf. Sci.* **530**, 201–218 (2020).
9. L. Xia et al., "New fast algorithm of Hough transform detection of circles," *Appl. Res. Comput.* **24**(10), 197–199, 210 (2007).
10. F. Ye et al., "Fast circle detection algorithm using sequenced Hough transform," *Opt. Precis. Eng.* **22**(4), 1105–1111 (2014).
11. M. Alazab, M. Islam, and S. Venkatraman, "Towards automatic image segmentation using optimised region growing technique," *Lect. Notes Comput. Sci.* **5866**, 131–139 (2009).
12. Q. K. Liang et al., "Angle aided circle detection based on randomized Hough transform and its application in welding spots detection," *Math. Biosci. Eng.* **16**(3), 1244–1257 (2019).

13. X. Xie et al., "An improved industrial sub-pixel edge detection algorithm based on coarse and precise location," *J. Ambient Intell. Hum. Comput.* **11**, 2061–2070 (2019).
14. V. Kumar, A. Asati, and A. Gupta, "Memory-efficient architecture of circle Hough transform and its FPGA implementation for iris localization," *IET Image Process.* **12**(10), 1753–1761 (2018).
15. C. A. O. De Melo et al., "Semi-automated counting model for arbuscular mycorrhizal fungi spores using the Circle Hough Transform and an artificial neural network," *Ann. Brazilian Acad. Sci.* **91**(4), e20180165 (2019).
16. M. S. Pillai et al., "Real-time image enhancement for an automatic automobile accident detection through CCTV using deep learning," *Soft Comput.* **25**, 11929–11940 (2021).
17. T. Zhang et al., "2-D cartoon character detection based on scalable-shape context and Hough voting," *Inf. Technol. J.* **12**(12), 2342–2349 (2013).
18. A. A. El-Latif et al., "A new encryption scheme for color images based on quantum chaotic system in transform domain," *Proc. SPIE* **8878**, 88781S (2013).
19. Y. Takashimizu and M. Iiyoshi, "New parameter of roundness R: circularity corrected by aspect ratio," *Prog. Earth Planet. Sci.* **3**(1), 2 (2016).
20. L. Y. Guo et al., "Sub-pixel level defect detection based on notch filter and image registration," *Int. J. Pattern Recognit. Artif. Intell.* **32**(6), 1854016 (2018).

Min Qi is an associate professor in the School of Electronics and Information, Northwestern Polytechnical University, China. She received her PhD in systems engineering at the same university in 2000. Currently, her research focuses on signal and information processing, industrial vision inspection, pattern recognition, and artificial intelligence.

Yanan Wang received her BE degree from Northwestern Polytechnical University, Xi'an, China, in 2018. She is currently pursuing the ME degree with the school of electronic and information, NPU. Her research interest is signal and information processing and image processing.

Yanshuo Chen received his BE degree from Northwestern Polytechnical University, Xi'an, China, in 2018. He is currently pursuing the ME degree with the School of Electronic and Information, NPU. His research interest is signal and information processing, unsupervised defect detection and machine learning.

Hongjuan Xin received her BE degree in 2012 and the ME degree in 2015, both from Northwestern Polytechnical University, Xi'an, China. Her research interest is signal and information processing and image processing. She is now working as a quality assurance engineer at Fujian Fuqing Nuclear Power Co. Ltd.

Yuelei Xu is a professor at the Unmanned System Research Institute, Northwestern Polytechnical University, China. Currently, his research focuses on intelligent unmanned systems, intellisense enhancement, machine vision, and brain-like computing.

Hongying Meng is a reader at the Department of Electronic and Electrical Engineering, College of Engineering, Design and Physical Sciences, Brunel University London. His research focuses on digital signal processing, artificial intelligence, human computer interaction, computer vision, embedded systems and communications. He is an IEEE senior member and associate editor for IEEE, TCSVT, and TCDS.

Aili Wang received her BE degree from Yanshan University, Qinhuangdao, China in 2014, and received her ME degree from the Northwestern Polytechnical University, Xi'an, China, in 2017. Her research interest is electronics and communication and image processing. Now her work is TV signal processing at TCL Corporation in Shenzhen.