IET Journals

The Institution of Engineering and Technology

Accurate image registration method for PCB defects detection

Guoliang Hua[1], Weipeng Huang[2] ✉, Hong Liu[2]

[1]*College of Electronic Information and Optical Engineering, Nankai University, Tianjin, People's Republic of China*
[2]*Key Laboratory of Machine Perception, Peking University, Shenzhen Graduate School, Shenzhen, People's Republic of China*
✉ *E-mail: wepon@pku.edu.cn*

Abstract: Image registration technology has been widely used in many parts of the computer vision system such as the automatic optical inspection system which is used to detect the printed circuit board (PCB) defects. The accuracy of the image registration will deeply influence the system's performance, so this study proposed an accurate image registration algorithm and applied it to the PCB defect detection. Good features to track feature detector and speeded up robust feature descriptor are combined to extract efficient features to achieve the first accurate image registration. Afterwards, cross-correlation functions were used to compute the shift between the reference image and the first-registered image for further accurate registration. Experimental results show that the authors' algorithm performs a much better registration, with a lower root-mean-square error value between the reference image and transformed image. What is more, they applied it to detect the defects of PCB with a high accuracy.

1 Introduction

Image registration technology, which is a difficult and hot topic in the field of image processing, has been widely applied to target detection, pattern registration, medical image analysis etc. In different applications, image recognition has different meaning [1]. When the images have the same content, but there existed some geometric transformation between these images, image registration means the accurate match between these images or correction of the transformed images to the reference image [2, 3]. When some parts of the images have the same contents, image registration means the seamless splicing between images [4]. Moreover, if images are photographed in different sensors, image registration means the fusion of different image information [5].

According to different image information used in the image registration process, image registration methods can be divided into three categories: methods based on image grey-scale information, methods based on image transformation domain information and methods based on image feature information. The method of image registration based on image grey-scale information is the earliest and simplest implementation of a registration method. It does not require complex pre-processing of the image in advance. According to the selected similarity measurement function, this method gets the registration parameters when the max similarity value is found in the searching process. The accuracy of this method can be of high-precision, but greatly influenced by the similarity measure function. Commonly used similarity measure functions include sequential similarity detection algorithm [6], cross-correlation (C-C) [7] mutual information [8] etc. At the same time, this method may take a lot of calculation and time in finding the maximum similarity value between images. The method of image registration based on image transformation domain information transforms the image information into corresponding transform domain information through Fourier transform (or discrete Fourier transform) [9], wavelet transform [10], Walsh transform [11] and other transform methods. According to the specific performance in the transformation domain of rotation, shift and other image transformation types, we can compute the transformation parameters to register the images. This kind of method can achieve image registration in some cases, but with great limitations. The image registration method based on the image feature information extracts and match the features of the reference image and the transformed image first, and then estimates the transformation parameters to register the image from these matched feature points' coordinates. Since the salient feature information of the image can be preserved in the geometric transformation such as corner points, straight lines [12] and so on, this method is robust and adaptive. At the same time, because this algorithm only needs to perform the matching calculation between the feature points extracted, the computational complexity is greatly reduced, and the efficiency of image registration is improved. Some of the commonly used features include Harris corners [13], scale-invariant feature transform (SIFT) [14], speeded up robust feature (SURF) [15] and so on.

With the computer vision technology developing so fast and widely applying to the products' defect detection, automatic optic inspection system has been used to detect the defects of printed circuit board (PCB). Some algorithms were proposed to achieve the accurate and fast detection of PCB defects, normally divided into three categories, the referential method [16, 17], non-referential inspection method [18] and the hybrid inspection method [19]. Since the referential method is the simplest method with a combination of fast speed and high accuracy, it is the most commonly used method to detect the PCB defects. One of the most difficult processes is to achieve the accurate registration between the reference image and the detecting image, which refers to the first meaning we talked before. Here, we proposed an accurate image registration algorithm for PCB defects detection.

2 Proposed algorithm

The image registration method proposed in this paper's process is shown in Fig. 1.

Instead of using the common SURF detector, we used the good features to track (GFTT) corner detector to extract the feature points from the reference image and transformed image, which can detect the abundant corner features precisely and speed up the detection process. Then, we used the description part of the SURF algorithm to describe each feature points and found the matched feature points by computing their description distance. Later, these matched feature points were used to compute the transforming parameters through the random sample consensus (RANSAC) algorithm. Thus, the first registration finished already. Finally, the C-C method was used to compute the shifts between the first-registered image and the reference image, in order to get the accurate registered image.
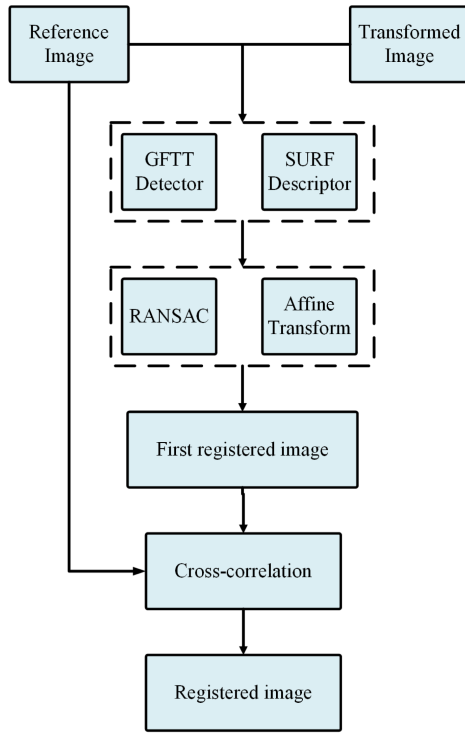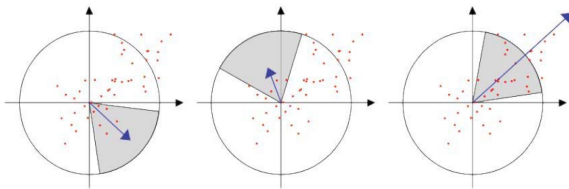
Fig. 1 *Process of the proposed algorithm*

Fig. 2 *Determine the main orientation*

2.1 Good FTT

GFTT is proposed by Shi and Tomasi [20], where the strongest corner detector was proposed. It developed from the Harris corner detector and solved some disadvantages of the Harris corner detector such as corners clustering, corner information loss and the position offset.

In the earliest corner detection method, the corner was defined as a point with low self-similarity. Harris and Stephens proposed an algorithm named Harris corner detector to detect corners existed in the image, which only considered the similarity between the pixels in one direction. The process is as follows.

Assume that detecting the corners of the area around the point (u, v) on a grey-scale image, setting the direction vector as (x, y) and computing the similarity $S(x,y)$ in the equation below:

$$S(x, y) = \sum_{u} \sum_{v} w(u, v)(I(u + x, v + y) - I(u, v))^2 \quad (1)$$

I_x and I_y are the partial differential of the image I in the x-direction and the y-direction, respectively. We can get the Taylor expansion of $I(u + x, v + y)$ in the equation below:

$$I(u + x, v + y) \simeq I(u, v) + I_x(u, v) x + I_y(u, v) y \quad (2)$$

Then, we can turn (1) into (3) or (4)

$$S(x, y) = \sum_{u} \sum_{v} w(u, v)(I_x(u, v) x + I_y(u, v) y)^2 \quad (3)$$

or

$$S(x, y) = (x \quad y) A \begin{pmatrix} x \\ y \end{pmatrix} \quad (4)$$

where

$$A = \sum_{u} \sum_{v} w(u, v) \begin{pmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{pmatrix}$$

By analysing matrix A's two eigenvalues, λ_1, λ_2, determine whether there are corner points existed:

i. If $\lambda_1 \not\subset 0$, $\lambda_2 \not\subset 0$, point (u,v) is a normal point.
ii. If there is one eigenvalue approximating zero, and another is a big positive number, point (u,v) is on a line.
iii. If both eigenvalues are big positive numbers, point (u,v) is a corner point.

However, it needs a lot of computation to compute the matrix A's eigenvalue, thus an approximate value M_c is defined in the Harris corner detection algorithm as given in the equation below:

$$\begin{aligned} M_c &= \lambda_1 \lambda_2 - \kappa(\lambda_1 + \lambda_2)^2 \\ &= \det^2 (A) - \kappa \text{trace}^2(A) \end{aligned} \quad (5)$$

Then, it only needs to calculate the value of M_c. When M_c is bigger than the threshold, there exists one corner point.

Furthermore, GFTT corner detector computes the minimum value of the two eigenvalues, finding one corner existed when the minimum value is bigger than the threshold.

2.2 Speeded up RF

SURF is an improved feature detecting and describing algorithm based on SIFT, which is more efficient and more robust against different image transformations than SIFT [15].

The part of this description of SURF has two parts: determine the main orientation of feature points and generate SURF descriptors.

2.2.1 Determine the main orientation of the feature points: For the purpose of keeping the invariance to rotation, it is necessary to set an orientation for every feature points. In the SURF algorithm, a circle with a radius of 6s is selected around the feature point, with s as the scale at which the interest point was detected. Then, calculate the Haar wavelet responses in the x-direction and the y-direction for every pixel in the circle area. It is obvious that the closer to the feature point, the bigger influence of the sampled pixels is to the feature point. So the responses need to be weighted, where the Gaussian weights are usually used.

Finally, as is shown in Fig. 2, a sliding sector window with an angle of 60° rotates around the feature point with the step of 0.2 rad. In each step, the horizontal and vertical responses of the sampled pixels in the window are summed and yield a vector v_w in the equation below:

$$v_w = A_w \angle \theta_w \quad (6)$$

where

$$A_w = \sum dx + \sum dy$$

$$\theta_w = \arctan\left(\frac{\sum dy}{\sum dx}\right)$$

A_w is the amplitude of the vector, θ_w is the angle of the vector.

After the sliding sector window finished a revolution, the authors compare all the vectors' amplitude yielded before, and choose the vector's angle as the feature point's orientation, whose amplitude is the max.
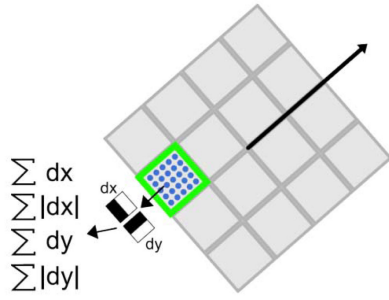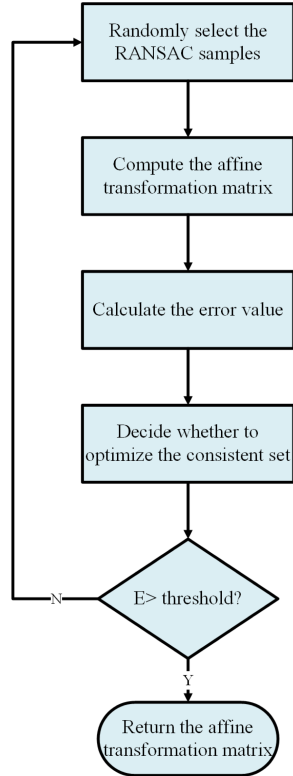
Fig. 3 *Descriptor vector*

Fig. 4 *Process to estimate the affine transformation matrix*

2.2.2 Generate the descriptor: To get the description that can distinguish these feature points, SURF algorithm constructs a square area around the feature point, with the size of $20s \times 20s$. What is more, the square area's orientation should be the same as the feature point's to keep invariance to the rotation. As is shown in Fig. 3, the square area is split up to 16 sub-areas and each sub-area's size is $5s \times 5s$.

In each area, the authors also calculated the x-direction and the y-direction's Haar wavelet responses of every sampled pixel. What is needed to emphasise is that the x-direction and the y-direction should correspond to the feature point's orientation, but not exactly the horizontal and vertical directions. Define the two directions' Haar wavelet responses as dx and dy, respectively, then, weigh the sampled pixels' distance to the feature point with the Gaussian weights and sum all the dx and dy in the sub-area to be a part of the descriptor. What is more, in order to include the change in intensity, the absolute values of dx and dy are summed to be part of the descriptor too. Then, each sub-area can be described by a vector as below:

$$v = \left(\sum dx, \ \sum d|x|, \ \sum dy, \ \sum d|y| \right) \quad (7)$$

Finally, the authors combine all the 16 sub-areas' vectors together, and yield the feature point's descriptor.

After computing the descriptors, the Euclidean distance among all the feature points' descriptors can be calculated and the points sharing the closest distance as a pair of matched points can be selected.

2.3 Affine transform

After getting the matched feature points, we need to estimate the transformation model. For the real production line of the PCB, the captured image of the cameras exists some kinds of transformation such as shift, flip, rotation, scale and so on. The affine transformation model with six parameters can perfectly solve these transformations, thus selected in our algorithm to register the image.

Assume the reference image $I(X)$ and the pixel point $X(x,y)$ in the image transform into $Y(u,v)$ through the affine transformation model, then X and Y have the relationship of the equation below:

$$\begin{bmatrix} u \\ v \end{bmatrix} = A \begin{bmatrix} x \\ y \end{bmatrix} + B \quad (8)$$

where

$$A = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix}, \quad B = \begin{bmatrix} a_5 \\ a_6 \end{bmatrix}$$

Matrix A explains the rotation, scale and flip transformation and matrix B explains the shift transformation; here, are some examples

$$\text{i}. \ A = \begin{bmatrix} a & 0 \\ 0 & a \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

the image will zoom in or out

$$\text{ii}. \ A = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

the image will rotate θ radian

$$\text{iii}. \ A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix},$$

the image will shift.

This paper combines the RANSAC algorithm and the least-squares method to estimate the affine transformation parameters. The main function of the RANSAC algorithm is to delete the mis-matched points from Section 2.2, and then perform model parameters' estimation based on the least-squares method. The basic process of the RANSAC algorithm is shown in Fig. 4:

(i) Randomly select an RANSAC sample from all the samples, that is, three pairs of matched points.
(ii) Compute the transformation matrix M according to three selected pairs of matched points.
(iii) According to all the samples, the transformation matrix M and the error metric function (9) yield the consistent set and the number of elements

(see (9)) .

(iv) According to the number of the elements in the consistent set decide whether to optimise the consistent set.

$$\text{error} = \sum_i \left\{ [x_i' - (a_1 x_i + a_2 y_i + t_1)] + [y_i' - (a_3 x_i + a_4 y_i + t_2)] \right\} \quad (9)$$

(v) Update the error rate E, if E is bigger than the set error threshold, proceed through iterations (i)–(iv) and continue until it is smaller than the current error rate E.

After the process, we can get a good affine transformation matrix. Then, the transformed image can be registered through the affine transformation module by using the affine transformation matrix gotten from the process in Fig. 4. This is the first registration process in our algorithm.

2.4 Accurate registration based on C-C information

After the first registration, we found that there is still sub-pixel shift to the reference image, so we use the C-C method to make the accurate registration and the algorithm was described in [21].

In this section, the reference image and the first-registered image are assumed to have the same size, $nx \times ny$. We use $I_R(x,y)$ to represent the reference image and $I_F(x,y)$ to represent the first-registered image. The process to shift the first-registered image corresponding to the reference image is shown in Fig. 5.

(i) First of all, we need to flatten the two-dimensional (2D) image array into a 1D array, in order to calculate the C-C conveniently. Assume that $I_R(x,y)$ is flattened into $f(n)$ and $I_F(x,y)$ is flattened into $g(n)$. Then, we calculate the C-C matrix between $f(n)$ and $g(n)$, and the C-C operation is defined as the equation below:

$$(f \times g)[n] = \sum_{m=-\infty}^{\infty} f^*[m]g[m+n] \quad (10)$$

After the C-C operation, we will get the C-C matrix, $ccor[n]$:

(ii) Second, we use the traversal method to find the max value in $ccor[n]$, and return the index mx.
(iii) Third, we will calculate the first and second partial derivations. Before the computation, we need to do some pre-work.

Convert the C-C matrix $ccor[n]$ into 2D matrix with the same size of $I_R(x,y)$, define it as $accor[n]$.

Get the point's coordinates (i_x, i_y) from mx in the equations below:

$$i_x = mx \bmod nx \quad (11)$$

$$i_y = mx/nx \quad (12)$$

Then, we start calculating the first and second partial derivations in the equations below:

$$dx = \frac{accor[i_x + 1, i_y] - accor[i_x - 1, i_y]}{2} \quad (13)$$

$$dy = \frac{accor[i_x, i_y + 1] - accor[i_x, i_y - 1]}{2} \quad (14)$$

$$d^2x = accor[i_x + 1, i_y] + accor[i_x - 1, i_y] - 2accor[i_x, i_y] \quad (15)$$

$$d^2y = accor[i_x, i_y + 1] + accor[i_x, i_y - 1] - 2accor[i_x, i_y] \quad (16)$$

$$dxy = \frac{accor[i_x + 1, i_y + 1] + accor[i_x - 1, i_y - 1]}{4} - \frac{accor[i_x + 1, i_y - 1] - accor[i_x - 1, i_y + 1]}{4} \quad (17)$$

With these derivations, we can estimate the optimal shifts in the equations below:

$$\text{shift } x = \frac{d^2y\,dx - dy\,dxy}{dxy\,dxy - d^2x\,d^2y} \quad (18)$$
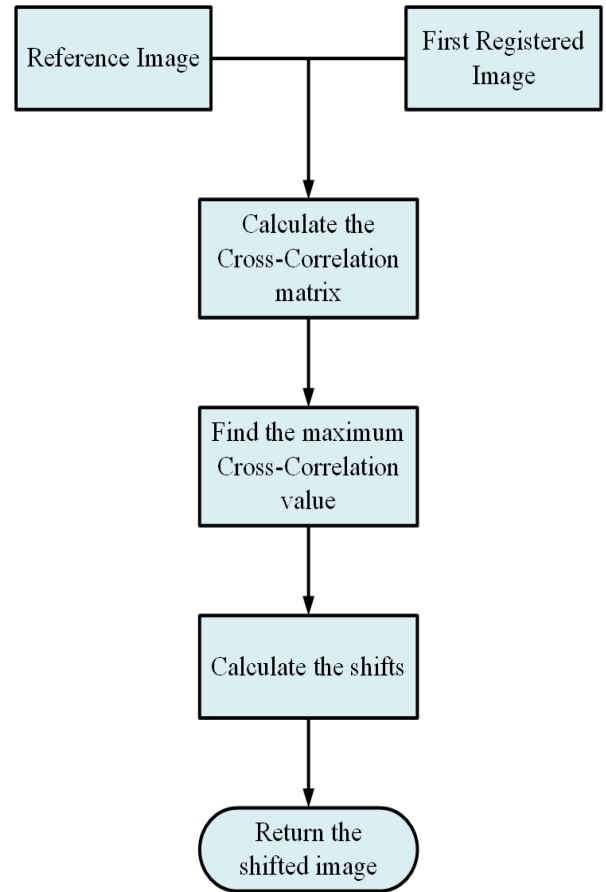

Fig. 5 *Process to shift the first-registered image*

$$\text{shift } y = \frac{d^2x\,dy - dx\,dxy}{dxy\,dxy - d^2x\,d^2y} \quad (19)$$

Finally, according to the shifts, we can translate the first-registered image corresponding to the reference image. Thus, we make a further image registration between the first-registered image and the reference image.

3 Experiments and discussion

In this section, we randomly selected some PCB images with different geometric transformations including rotation, scale and 3D transformation, as the experimental test dataset. To show the advantages of the proposed method in image registration, we used root-mean-square error (RMSE) (20) as the evaluation standard for registration accuracy. $T(p_i, \theta)$ represents the registered image and p_i represent the transformed image and we calculate the RMSE value between them and compare it with other image registration algorithm: registration method based on discrete fourier transform domain and image registration algorithm based on SURF features

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\| T(p'_i, \theta) - p_i \|)} \quad (20)$$

Here are the experiment results shown in Fig. 6, using the algorithm proposed in this paper, the RMSE value is lower than the others. Figs. 6*a* and *b* show the RMSE value of registration of the images with different rotation and scale using a different algorithm, respectively. The RMSE value when using the proposed algorithm is lower than 0.1, having a big improvement than other existed algorithms. What is more, when there are various transformations between the reference image and the transformed image, the algorithm proposed in this paper still has a big improvement in registration accuracy. What is more, Fig. 6 also shows that using the features extracted from the combination of GFTT feature detector and SURF feature descriptor to achieve the image
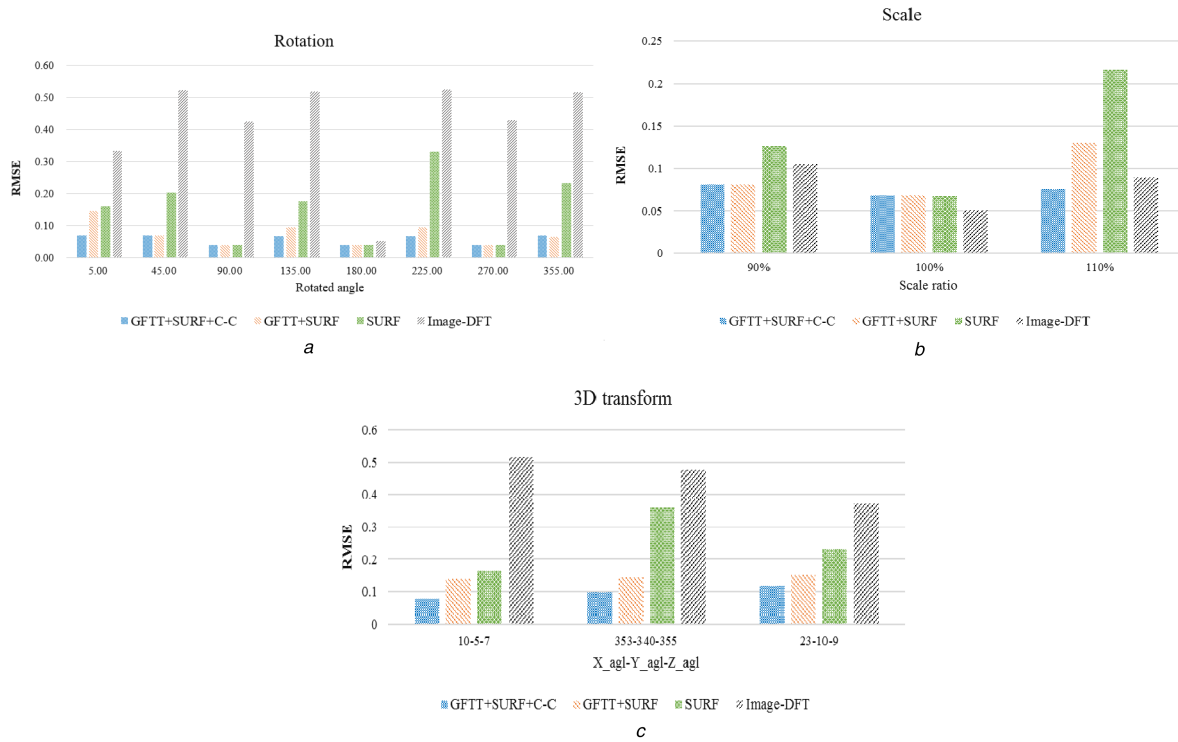
Fig. 6 *RMSE value*
(a) RMSE value under rotation, *(b)* RMSE value under scale,
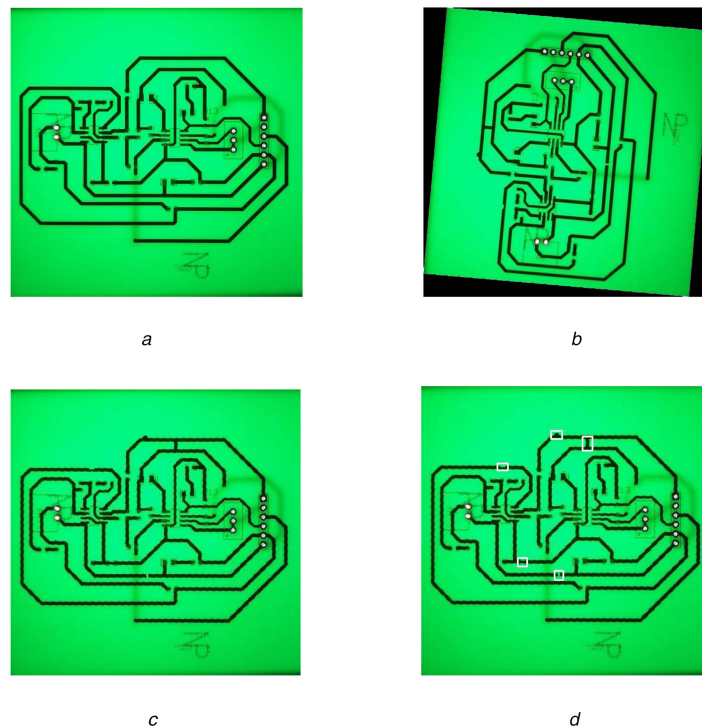*(c)* RMSE value under 3D transform

Fig. 7 *Proposed algorithm used in the PCB defect detection*
(a) Reference image, *(b)* Detecting image,
(c) Registered image, *(d)* Registered image with boxing the defects' position

registration has already performed better than another algorithm. Furthermore, the C-C method is added to get a more accurate registration result, which can satisfy the need in the defects detection of PCB. The algorithm was applied to the PCB defects detection and it was found performing splendidly. The results are shown in Fig. 7. The detecting PCB image was well registered and all the defects were detected. It shows that the algorithm proposed in this paper can be applied to the PCB defects detection.

4 Conclusion

In this paper, we combine the GFTT feature detector and the SURF descriptor to extract the efficient features of the images and achieve the first accurate image registration whose accuracy is better than previous algorithms. Furthermore, the C-C method was added to satisfy the high-precision need in the PCB defects detection. Experimental results show that our algorithm is quite usable and can be applied to the PCB defects detection with a more robust and accurate performance.

5 Acknowledgment

6 References

[1] Brown, L.G.: 'A survey of image registration techniques', *ACM Comput. Surv.*, 1999, **24**, (4), pp. 325–376

[2] Mashohor, S, Evans, J.R., Arslan, T.: 'Image registration of printed circuit boards using hybrid genetic algorithm'. IEEE Int. Conf. Evolutionary Computation, Vancouver, Canada, July 2006, pp. 2685–2690

[3] Wang, W., Yan, X., Wang, Z*., et al.*: 'A robust affine invariant point extraction algorithm for image registration'. Int. Symp. Computational Intelligence and Design, Hangzhou, China, December 2017, pp. 79–82

[4] Brown, M., Lowe, D. G.: 'Automatic panoramic image stitching using invariant features', *Int. J. Comput. Vis.*, 2007, **74**, (1), pp. 59–73

[5] Yang, W., Han, C., Sun, H*., et al.*: 'Registration of high resolution SAR and optical images based on multiple features'. Int. Geoscience and Remote Sensing Symp., Seoul, South Korea, July 2005, pp. 3542–3544

[6] Barnea, D.I., Silverman, H.F.: 'A class of algorithms for fast digital image registration', *IEEE Trans. Comput.*, 2009, **21**, (2), pp. 179–186

[7] Dubois, E., Mitiche, A.: 'Digital picture processing', *Proc. IEEE*, 1990, **73**, (1), pp. 169–170

[8] Colerhodes, A., Zavorin, I., Moigne, J.L.: 'Mutual information as a similarity measure for remote sensing image registration', *Proc. SPIE*, 2011, **12**, (2), pp. 179–186

[9] Reddy, B.S., Chatterji, B.N.: 'An FFT-based technique for translation, rotation, and scale-invariant image registration', *IEEE Trans. Image Process.*, 1996, **5**, (8), pp. 1266–1271

[10] Fan, X., Rhody, H., Saber, E.: 'Automatic registration of multisensor airborne imagery', *Diss. theses – Gradworks*, 2011, **12**, (3), pp. 6–86

[11] Lazaridis, G., Petrou, M.: 'Image registration using the Walsh transform', *IEEE Trans. Image Process.*, 2006, **15**, (8), pp. 2343–2357

[12] Larntz, K., Hampel, F.R., Ronchetti, E.M*., et al.*: 'Robust statistics', *J. Am. Stat. Assoc.*, 1989, **84**, (406), p. 624

[13] Harris, C.J.: 'A combined corner and edge detector'. Proc. Alvey Vision Conf., Manchester, March 1988, pp. 147–151

[14] Lowe, D.G.: 'Object recognition from local scale-invariant features'. Int. Conf. Computer Vision, Kerkyra, Corfu, Greece, 2002, pp. 1150

[15] Bay, H., Tuytelaars, T., Gool, L.V.: 'SURF: speeded up robust features'. European Conf. Computer Vision, New York, USA, May 2006, pp. 404–417

[16] Leta, F.R., Feliciano, F.F., Martins, F.P.R.: 'Computer vision system for printed circuit board inspection'. *ABCM Symp. Series in Mechatronics*, 2008, **3**, pp. 623–632

[17] Ren, S., Lu, L., Zhao, L*., et al.*: 'Circuit board defect detection based on image processing'. Int. Congress on Image and Signal Processing (CISP), Shenyang, China, February 2016, pp. 899–903

[18] Ejiri, M., Uno, T., Mese, M*., et al.*: 'A process for detecting defects in complicated patterns', *Comput. Graph. Image Process.*, 1973, **2**, (3), pp. 326–339

[19] Mandeville, J.R.: 'Novel method for analysis of printed circuit images', *IBM J. Res. Dev.*, 2010, **29**, (1), pp. 73–86

[20] Shi, J., Tomasi, C.: 'Good features to track'. IEEE Computer Society Conf., Charlottesville, VA, USA, September 2002, pp. 593–600

[21] 'Cross_cor_taylor.pro'. Available at http://solarmuri.ssl.berkeley.edu/~welsch/public/software/cross_cor_taylor.pro, accessed 9 June 2011