PCB Large Color Variation Image Registration with Local Optimization LoFTR

Yingyan Hou[1,2,3(✉)], Yidan Zhang[1,2,3], Xiaoxuan Liu[1,2,3], Hui Wu[1,4,5], Jie Jia[1,2,3], Xiaohe Li[1,3], Shixiong Liu[1,3], Lei Wang[1,2,3], and Xinyu Zhao[1,2,3]

[1] Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100190, China
houyy@aircas.ac.cn
[2] Key Laboratory of Target Cognition and Application Technology (TCAT), Aerospace Information Research Institute, Beijing 100190, China
[3] Key Laboratory of Network Information System Technology (NIST), Aerospace Information Research Institute, Beijing 100190, China
[4] University of Chinese Academy of Sciences, Beijing 100190, China
[5] School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing 100190, China

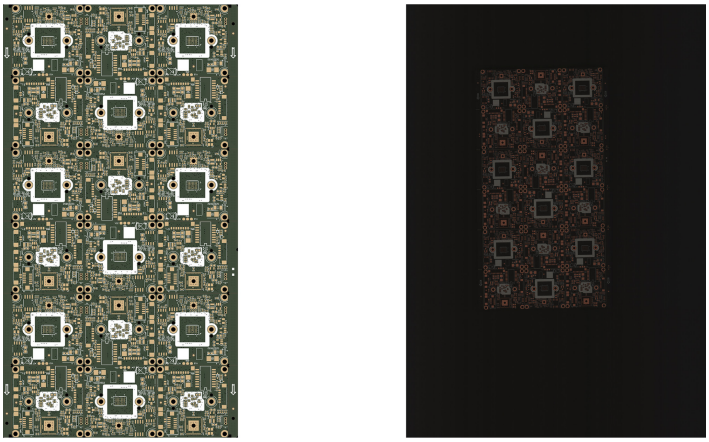
Abstract. For the detection of Printed Circuit Board (PCB) defects with large color variations and large sizes, the traditional PCB image registration algorithm suffers from the problems of long time consumption and low accuracy, and the large color variation also interferes with the registration algorithm. We propose an image registration method for PCB with large color differences and large sizes to solve the problem that PCB image registration is easy to misalign. First, the large color variation problem of PCB images is corrected by the region-based color correction algorithm. Then, a local optimization feature matching algorithm is proposed for PCB image feature matching in response to the problem that the LoFTR algorithm loses the first window vector information. Finally, the MAGSAC++ algorithm is used to match PCB images. Experimental results show that the traditional feature matching algorithm is difficult to use for large-size PCB image registration, while the PCB registration method proposed in this paper has higher accuracy compared with the LoFTR algorithm, and further improves the registration accuracy by the region-based color correction algorithm.

Keywords: Printed circuit board · Image registration · Feature matching

1 Introduction

Printed Circuit Board (PCB), as crucial electronic components, is indispensable in various fields such as civilian communication electronics, consumer electronics, new energy, aerospace, and so on. PCB is virtually present in all electronic

W. Yu and L. Xuan (Eds.): MLICOM 2023, LNICST 518, pp. 38–52, 2024.
https://doi.org/10.1007/978-3-031-71716-1_4



(a) Reference image (b) PCB to be inspected

Fig. 1. PCB Image

devices. With the explosive growth of electronic devices, the demand for PCBs continues to rise, and factories' production lines churn out thousands of printed circuit boards daily. However, such large-scale production poses significant challenges for the quality inspection of PCBs.

PCB, being a precision electronic component at the micron level, is susceptible to potentially fatal defects. Therefore, the accuracy of defect detection is crucial for the quality inspection of PCBs. However, manual inspection of PCB defects entails significant labor costs and is subject to a certain level of subjectivity. Automated defect detection for PCBs can reduce costs while enhancing the precision and robustness of defect detection. In automated defect detection, manufacturers provide a standard board (reference image) as a ground truth value for supervised training. Figure 1a depicts an image of a standard board, while Fig. 1b shows a PCB image to be inspected. It is evident that the untreated PCB image to be inspected contains a considerable amount of background noise and repetitive textures, with a significant color discrepancy compared to the reference image. Utilizing untreated PCB images for defect detection may adversely affect the detection results and lead to detection failures. Therefore, achieving high-precision registration of the PCB image to be inspected based on the reference image is a critical foundation for automated PCB defect detection.

Performing image registration on large-size PCBs presents several challenges. Firstly, there is a noticeable color disparity between reference images and images to be registered, which is detrimental to feature matching. Secondly, PCB images contain a substantial amount of repetitive textures, posing a significant challenge for feature matching. Lastly, PCB images are highly precise, requiring high-precision registration. Commonly employed image registration algorithms in the industry typically rely on techniques such as Scale-Invariant Feature Transform (SIFT) for feature matching [1]. However, feature operators like SIFT intro-

duce high computational costs and slow processing speeds due to the pursuit of increased affine invariance. Consequently, a practical and high-precision image registration method for PCB images is of paramount importance in the PCB production environment.

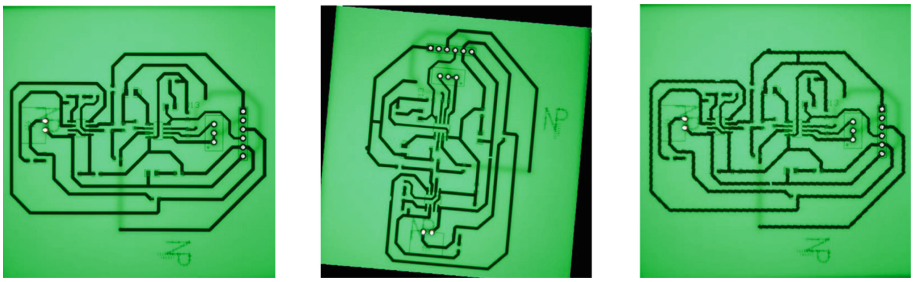
The main contributions of this paper are as follows:

1. We propose an effective method for PCB image registration, which achieves high-precision registration of large-size PCB images.
2. We introduce a region-based color correction algorithm for PCB image preprocessing. Due to environmental influences during image acquisition, color disparities may exist between images. We achieve image color consistency with color correction.
3. LoFTR misses the remaining vector information in the first window. In this study, we propose a local optimization LoFTR feature matching algorithm for PCB image registration. We employ the local optimization LoFTR algorithm to calculate the homography matrix. The registered image is obtained based on the homography estimation results.

2 Related Work

As early as 2006, Mashohor utilized genetic algorithms to compute the rotation angle and displacement of images, achieving registration of PCB images on a conveyor belt [2]. In the early stages of image registration, the primary idea was to minimize the differences in image information, such as the mutual information method [3], normalized cross-correlation method [4], Phase Correlation [5,6], etc. However, merely calculating rotation and displacement does not meet the demand for high-precision registration of PCB images in complex environments. In recent years, academic research on PCB image registration has mainly focused on improving feature matching algorithms.

Scale-Invariant Feature Transform (SIFT) is a classic algorithm for feature matching that establishes a scale space, detects extrema, and assigns feature point orientations within it [1]. Speeded-Up Robust Features (SURF) is a detector and descriptor inspired by SIFT, known for its lower computational cost compared to SIFT [7]. Oriented Fast and Rotated BRIEF Features (ORB) algorithm employs the FAST algorithm [8] for feature point detection and the BRIEF algorithm [9] for descriptor computation. Dai et al. combined Particle Swarm Optimization (PSO) with the SIFT algorithm for PCB image registration, leveraging PSO's powerful search capability to obtain optimal affine transformation parameters and optimize feature matching performance. Their image size of PCB was 400×400 [10]. Hua et al. utilized feature tracking corner detectors and SURF for feature point extraction and description, followed by fine registration using cross-correlation (see Fig. 2). Their PCB image registration dataset is relatively simple [11]. Huang et al. eliminate erroneous match pairs based on prior threshold boundary conditions, leveraging prior information about the mechanical errors of the PCB motion platform. However, this algorithm requires the selection of prior thresholds [12]. Li et al. used the Shi-Tomasi algorithm instead



(a) Reference image (b) Image to be registered (c) Registered image

Fig. 2. The PCB image registration results by Hua et al. [11]

of the SURF algorithm to extract the feature points in the overlapping region of the image to improve the efficiency and accuracy of the PCB image registration process [13].

In recent years, with the continuous progress and development of disciplines such as computer vision [14] and deep learning [15], feature matching is gradually embracing deep learning. Detector-based feature matching algorithms have gained attention. Fischer et al. trained a Convolutional Neural Network (CNN) with unlabeled data and found that features learned by CNN outperformed SIFT [16]. Han et al. introduced MatchNet, composed of CNN and three fully connected layers, to enhance performance based on image blocks using a deep network architecture [17]. Deep Convolutional Feature Point Descriptors, DeepDesc, compares image blocks through CNN to obtain feature descriptors [18]. Choy et al. proposed the Universal Correspondence Network (UCN), which enlarges the information range through a convolutional spatial transformer [19]. Yi et al. introduced Learned Invariant Feature Points (LIFT), in which CNN unifies feature extraction, estimation, and descriptor calculation in a single framework [20]. Daniel et al. proposed SuperPoint, a robust feature point detector that detects feature points of high density in an image [21]. Sarlin et al. proposed SuperGlue, which aims to establish the correspondence of local features in two images and is often used in conjunction with the SuperPoint paired with [22].

Detector-based feature matching may detect only a few feature points in images with repeated textures. Therefore, researchers have considered adopting detector-free methods based on Transformer for feature matching [23]. Rocco et al. introduced the Sparse-NCNet network, which employs the Neighbourhood Consensus Network (NCN) [24]. This method processes sparse matching correlation tensors and submanifold sparse convolutions [25]. Li et al. proposed the Dual-Resolution Correspondence Network (DRC-Net), transforming the matching problem into a regression problem. It utilizes neural networks to learn how to predict the matching relationships between every pixel in two images [26]. Jiang et al. introduced the Correspondence Transformer (COTR) feature matching algorithm, utilizing a Transformer architecture to match image features [27]. Sun et al. presented the Local Feature Transformer (LoFTR), a novel detector-free

local feature matching method [28]. Inspired by SuperGlue, LoFTR utilizes Self-Attention and Cross-Attention mechanisms for matching. The method employs a multi-scale strategy for matching and optimization, ensuring matching points even in sparse texture regions of images. It is noteworthy that the LoFTR feature matching algorithm has become a mainstream framework in recent feature matching research.

Currently, most of the relevant research on PCB image registration utilizes classical feature matching algorithms, and these studies generally involve smaller PCB image sizes with relatively simple PCB layouts. The objective of this study is to achieve high-precision feature matching for large-sized PCB images with significant color variations, thereby accomplishing high-precision registration of PCB images.

3 Method

3.1 Region-Based Color Correction

The reference image and the image to be registered for the Printed Circuit Board exhibit noticeable color discrepancies. In addressing the significant color disparities observed among printed circuit board images, this study proposes a region-based color correction algorithm for image preprocessing. The algorithm clusters and merges color blocks, performing distinct color corrections for different regions.

The Reinhard algorithm is not effective for color correction in content-rich images, where various color blocks mutually influence each other [29]. The region-based color correction algorithm independently applies color correction to different color blocks, ensuring that each color block is unaffected by others.

Region Clustering. Initially, the algorithm clusters regions based on features such as color and texture to obtain preliminary region boundaries. This study employs Simple Linear Iterative Clustering (SLIC) for superpixel segmentation, grouping adjacent pixels with similar texture, color, brightness, and other features into a single region [30].

Region Merging. After obtaining the initial regions through SLIC clustering, the next step involves merging the regions to reduce their quantity. Ideally, the goal is to correspond regions between images, meaning that the closer the colors of similar objects in the images, the better the result. However, due to the presence of background noise between images, achieving a perfect one-to-one correspondence for each region is not possible. To address this issue, SIFT feature matching is employed to obtain rough matching pairs. Based on the results of SLIC clustering and SIFT feature matching, the regions of the reference image I_a and the image to be corrected I_b are merged.

Region Color Correction. After the merging of regions, color correction is performed on a region-by-region basis. The image is transformed from the RGB color space to the $l\alpha\beta$ color space [29]. The $l\alpha\beta$ color space is based on human visual perception and provides a representation where the l component corresponds to brightness, aligning with human perception of luminance, while α and β represent chromaticity information. The $l\alpha\beta$ color space exhibits uniformity, meaning that similar color differences correspond to similar distances in space. Therefore, $l\alpha\beta$ is particularly suitable for computing distances and similarities between colors.

The direct conversion between the RGB space and the $l\alpha\beta$ color space is unfeasible. The conversion involves a multistep process: the first step transforms RGB values into Tristimulus color components. Subsequently, the color space transits from the Tristimulus components to the LMS space in the second step. Finally, the color space is further transformed from LMS space to $l\alpha\beta$ space.

In $l\alpha\beta$ space, the mean and standard deviation of each region are computed. For a specific region $R_b^{i'}$ in the target correction image and its corresponding region $R_a^{i'}$ in the reference image, color correction is performed according to Eq. 1.

$$C_b^{\mathrm{i}\prime}(m,n) = \mu_a^{\mathrm{i}} + \frac{\sigma_a^{\mathrm{i}}}{\sigma_b^{\mathrm{i}}}\left(C_b^{\mathrm{i}}(m,n) - \mu_b^{\mathrm{i}}\right) \tag{1}$$

where μ_b^i and μ_a^i represent the color mean values of a specific region $R_b^{i'}$ in the target correction image and its corresponding region $R_a^{i'}$ in the reference image, respectively. Similarly, σ_b^i and σ_a^i denote the standard deviations of regions $R_b^{i'}$ and $R_a^{i'}$, respectively. $C_b^{i'}(m,n)$ signifies the color-corrected pixel value at a specific pixel location (m,n) within the region $R_b^{i'}$, while $C_b^i(m,n)$ represents the pixel value before color correction at the same pixel location (m,n).

Ultimately, the image I_b' undergoes the reverse transformation from the $l\alpha\beta$ color space back to the RGB space, completing the color correction process.

3.2 Feature Matching Based on Local Optimization LoFTR

A typical algorithm for detector-free feature matching, such as LoFTR, uses 1/8 of the feature map for coarse-grained global matching [28]. After obtaining the coarse-grained confidence matrix, it directly maps the coarse-grained matches to the fine-grained feature map through filtering. Optimization is then performed in various windows of the fine-grained feature map. However, LoFTR local matching calculates the feature similarity between the center vector of the first image window and all the features in the corresponding window of the second image only in one direction, which misses the remaining vector information in the first window. In this study, we propose a local optimization LoFTR feature matching algorithm for PCB image registration.

Global Feature Matching Module. Feature Pyramid Networks (FPN) can accumulate low-level and high-level features, obtaining multi-scale features fused

with information from multiple layers. The local optimization LoFTR algorithm utilizes a fine-tuned ResNet-18 as the backbone network for feature extraction, with a stem set to 128, using the first three levels $L = \{128, 256\}$, resulting in feature maps containing rich information at different levels. For the first-level feature map, corresponding features are obtained after three layers; the second-level feature map is obtained from L_1. The local optimization LoFTR algorithm extracts two types of feature maps with resolutions of 1/8 and 1/2 of the original image, denoted as $F_g^A, F_g^B \in R^{C_g \times \frac{H}{8} \times \frac{W}{8}}, F_l^A, F_l^B \in R^{C_l \times \frac{H}{2} \times \frac{W}{2}}$, where C_g, C_l represent the feature dimensions, and H, W are the original image's height and width.

Firstly, the model takes the flattened feature maps F_g^A, F_g^B as input into the Interlaced Attention Module, utilizing linear attention to aggregate global information. The Interlaced Attention Module consists of a positional encoding block and N interlaced attention blocks, as illustrated in Fig. 3.

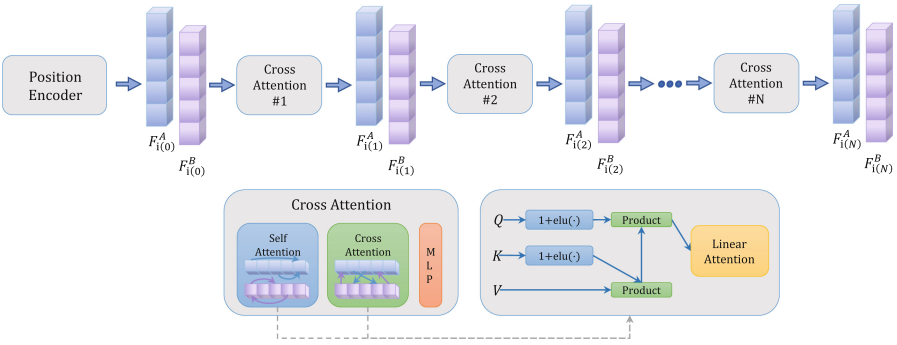


Fig. 3. Interlaced Attention Module

This study utilizes linear attention to reduce the complexity of the Interlaced Attention Module to $O(N)$ (similar to LoFTR [28]), where N is the length of the flattened input features. Each attention block includes a self-attention and a cross-attention. Three vectors $query(Q)$, $key(K)$, and $value(V)$ are defined as inputs to the attention layer. The linear attention layer transforms the input features F^i, F^j. The input features F^i, F^j are consistent for self-attention, while they are different for cross-attention.

The model computes four iterations of interleaved linear attention ($N_g = 4$), ultimately obtaining $F^A_{f(N_g)}, F^B_{f(N_g)}$. Following LoFTR, the score matrix $\mathcal{R}_g$ for each position is calculated using the Einstein summation convention.

$$\mathcal{R}_g(i,j) = \left\langle F^A_{f(N_g)}(i), F^B_{f(N_g)}(j) \right\rangle \tag{2}$$

As shown in Eq. 3, the bidirectional Softmax is employed to ensure differentiability. The matching probability matrix $\mathcal{C}_g$ is computed for the score matrix $\mathcal{R}_g$ (see Fig. 4).

$$\mathcal{C}_g(i,j) = \text{softmax}\left(\mathcal{R}_g(i,\cdot)\right)_j \cdot \text{softmax}\left(\mathcal{R}_g(\cdot,j)\right)_i \tag{3}$$

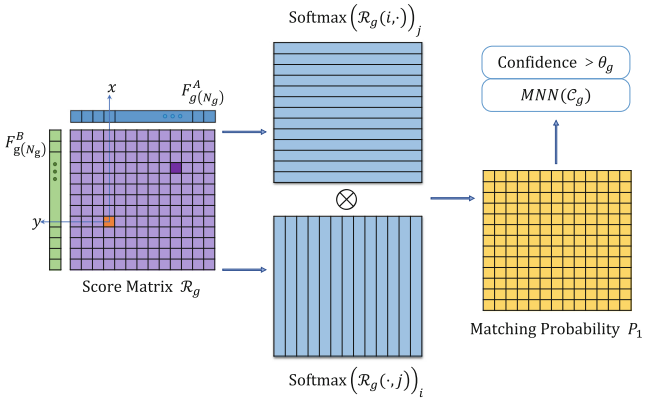


Fig. 4. Matching Probability Matrix

Based on the matching probability matrix $\mathcal{C}_g$, the model's focus area is obtained by filtering and selecting matching relationships. The matching relationships are filtered based on the following two conditions:

1. Confidence exceeding the threshold θ_g;
2. Mutual Nearest Neighbors (MNN) filtering.

By employing the mutual nearest neighbor algorithm to filter outlying matching pairs, the impact of outliers on the range of the focal area can be eliminated. After filtering, all matches K_g satisfying the above two conditions are obtained. Based on the positions corresponding to K_g in both images, the matching relationship $\mathcal{H}_g = \{(D_g^A, D_g^B)\}$ is determined.

$$\mathcal{H}_g = \left\{\left(D_g^A(i), D_g^B(j)\right) \mid \forall (i,j) \in K_g\right\} \tag{4}$$

Local Optimization Module. LoFTR similarity is computed only in one direction, from the center vector of the first window to all features in the corresponding window of the second image. This results in the loss of additional vector information within the first window. To address this limitation, our study proposes a strategy of local pre-positioning followed by optimization, preserving the vector information within the first window. After obtaining the global matching relations $\mathcal{H}_g$ in the previous step, these matches are optimized on $F_l^A, F_l^B \in R^{C_l \times \frac{H}{2} \times \frac{W}{2}}$, ultimately achieving sub-pixel-level feature matching results.

For each matching pair (i,j) in $\mathcal{H}_g$, we position them on the high-resolution feature maps F_l^A, F_l^B to obtain two small windows for each global match on the

high-resolution feature maps, denoted as $F_l^A(w), F_l^B(w) \in R^{(w\times w\times C_l)}$. There are a total of $2K$ windows, where K is the number of matches in the global matching results $\mathcal{H}_g$, and C_l is the number of channels.

We flatten $F_l^A(w), F_l^B(w)$, and classical dot-product attention has been experimentally proven to perform better than non-linear attention in this context. Since the sequence is short, the time cost difference is negligible, so we adopt a single interleaved dot-product attention ($N_l = 1$). Finally, we obtain $F_{l(N_l)}^A, F_{l(N_l)}^B$.

Using only the center vector i from the first image's window is not accurate enough. We compute the score matrix $\mathcal{R}_l^w$ for the window as follows:

$$\mathcal{R}_l^w(i,j) = \left\langle F_{l(N_l)}^A(i), F_{l(N_l)}^B(j) \right\rangle \tag{5}$$

As shown in Eq. 6, we calculate the matching probability matrix $\mathcal{C}_l^w$ for the score matrix $\mathcal{R}_l^w$.

$$\mathcal{C}_l^w(i,j) = \text{softmax}\left(\mathcal{R}_l^w(i,\cdot)\right)_j \cdot \text{softmax}\left(\mathcal{R}_l^w(\cdot,j)\right)_i \tag{6}$$

The position corresponding to the maximum value in the matching probability matrix $\mathcal{C}_l^w$ is denoted as $(\widetilde{i},\widetilde{j})$, accurately locating the position $\widetilde{i}$ in the window of the first image rather than directly defining the center position. All the matching pairs $(\widetilde{i},\widetilde{j})$ in $K_{lc} = \left\{(\widetilde{i},\widetilde{j})\right\}$ form a set.

Based on the positions corresponding to K_{lc} in the two images, all matching pairs $(\widetilde{i},\widetilde{j}) \in K_{lc}$ are used to create pixel-level matching relationships $\mathcal{H}_{lc} = (D_{lc}^A, D_{lc}^B)$.

$$\mathcal{H}_{lc} = \left\{\left(D_{lc}^A(\widetilde{i}), D_{lc}^B(\widetilde{j})\right) \mid \forall(\widetilde{i},\widetilde{j}) \in K_{lc}\right\} \tag{7}$$

Similarly to LoFTR fine-grained matching, the correlation matrix between $\widetilde{i}$ and $F_{l(N_l)}^B$ is computed next. Softmax normalization is applied, and the probability distribution expectation is calculated to obtain sub-pixel-level matching pairs $(\widetilde{i},\widetilde{j}')$. All matching pairs in the windows form a set $K_l = \left(\widetilde{i},\widetilde{j}'\right)$.

Based on the positions corresponding to K_l in the two images, the final sub-pixel-level accurate matching relationships $\mathcal{H}_l = (D_l^A, D_l^B)$ are obtained.

$$\mathcal{H}_l = \left\{\left(D_l^A(\widetilde{i}), D_l^B(\widetilde{j}')\right) \mid \forall(\widetilde{i},\widetilde{j}') \in K_l\right\} \tag{8}$$

Loss Function. The loss function is divided into the global feature matching module $\mathcal{L}g$ and the local refinement optimization module, consisting of $\mathcal{L}_{lc}$ and $\mathcal{L}_l$. The overall loss function is formulated as follows:

$$\mathcal{L} = \mathcal{L}_g + \mathcal{L}_{lc} + \mathcal{L}_l \tag{9}$$

For the global feature matching part, the cross-entropy loss function is utilized to supervise the matching probability matrix $\mathcal{C}_g$, as shown in Eq. 10. Here, $\mathcal{H}_g^{gt}$ represents the ground truth values obtained by reprojecting the focused

region of the image using depth information and camera pose. The computation of $\mathcal{L}_g$ is as follows:

$$\mathcal{L}_g = -\frac{1}{\left|\mathcal{H}_g^{gt}\right|} \sum_{(i,j)\in\mathcal{H}_g^{gt}} \log(\mathcal{C}_g(i,j)) \tag{10}$$

The loss function for local positioning optimization is divided into two parts: $\mathcal{L}_{lc}$ and $\mathcal{L}_l$. $\mathcal{L}_{lc}$ supervises the matching probability matrix $\mathcal{C}_l^w$ to obtain pixel-level matching results. $\mathcal{L}_l$ calculates the L_2 distance, obtaining sub-pixel-level accurate pixel positions. Here, $\widetilde{j}'_{gt}$ represents the point $\widetilde{i}$ from the first image, reprojected into the second image using the depth map and camera pose.

$$\mathcal{L}_{lc} = -\frac{1}{\left|\mathcal{H}_{lc}^{gt}\right|} \sum_{(i,j)\in\mathcal{H}_{lc}^{gt}} \log(\mathcal{C}_{lc}^w(i,j)) \tag{11}$$

$$\mathcal{L}_l = \frac{1}{|\mathcal{H}_l|} \sum_{(\tilde{i},\tilde{j}')\in\mathcal{H}_l} \frac{1}{\sigma^2(\tilde{i})} \left\|\tilde{j}' - \tilde{j}_{gt}{}'\right\|_2 \tag{12}$$

3.3 Image Registration

PCB image registration is accomplished through spatial transformation based on the LoFTR algorithm's feature matching results. A three-dimensional homography matrix is employed to establish a spatial transformation relationship between given images, completing the PCB image registration process.

The most classic method for computing the homography matrix is the RANSAC algorithm [31]. However, RANSAC requires users to set an inlier threshold, which is based on subjective experience. Improper threshold setting may adversely affect the accuracy of model estimation. In contrast to the RANSAC algorithm, MAGSAC++ does not require users to set inlier and outlier thresholds; instead, it marginalizes the inlier threshold through the model evaluation function Q [32,33]. The matching results $\mathcal{H}_{pcb}$ are input into MAGSAC++, which performs local-to-global sampling to estimate the homography matrix.

After obtaining the matching results $\mathcal{H}_{pcb}$, the MAGSAC++ algorithm is employed to estimate the homography matrix H between the Printed Circuit Board (PCB) standard board (reference image) I_a and the image to be registered I_b. Subsequently, the image to be registered I_b undergoes a homography transformation using the estimated matrix H, completing the image registration.

4 Experiments

4.1 Datasets

This study conducted experiments using PCB images provided by a smart information equipment company. The standard board (reference image) comprises 456 images with varying sizes. The resolutions of relatively smaller reference images are 3545×3520, while larger reference images have resolutions of

10860×8780. The PCB images to be registered are generated based on ODB++ files, which are ASCII-encoded files containing comprehensive information about PCB assembly. Initially, an ODB++ parser was used to interpret the instructions in the ODB++ project file as drawing commands. Subsequently, the PCB images were rendered layer by layer, including the top (or bottom) layer, silk screen layer, solder mask layer, drill hole layer, and other layers, according to the rendering commands. Finally, following the PCB image generation process, the layers were combined and colored, resulting in 456 PCB images to be registered, all with a resolution of 12000×8320. It is worth noting that due to the rich component circuit information in the PCB images, no downsampling was performed to avoid loss of image details.

The registration evaluation metrics in this study include Root Mean Squared Error (RMSE), Structural Similarity Index (SSIM), and Peak Signal-to-Noise Ratio (PSNR). SSIM measures the structural similarity between images, while PSNR reflects image quality.

4.2 Results

First, the performance of the local optimization LoFTR algorithm in PCB image registration is compared with other algorithms, as shown in Table 1 (the best performance is highlighted in bold). Specifically, this comparative experiment strictly controls variables, using the same batch of images after color correction for experimentation. The RMSE metric shows an overall higher bias, attributed to the fact that even with perfect registration, the post-registration PCB images still exhibit some imperfections, making them not entirely identical to the images of the standard board. Therefore, the RMSE metric serves for relative comparison.

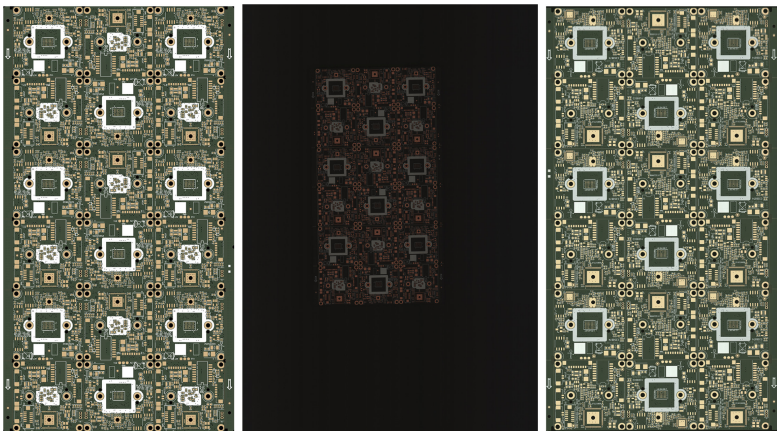
Table 1. PCB Image Registration

Method	SSIM	PSNR	RMSE
SIFT [1]	0.841	20.91	4.86
SURF [7]	0.839	20.88	4.87
SURF w/ Shi-Tomasi [13]	0.844	20.98	4.83
SuperPoint [21]+SuperGlue [22]	0.869	21.56	4.77
LoFTR [28]	0.874	21.48	4.73
Local Optimization LoFTR	**0.891**	**22.86**	**4.56**

As shown in Table 1, when using the local optimization LoFTR feature matching algorithm for image registration, the registered image exhibits lower RMSE, higher SSIM, and higher PSNR. Based on LoFTR or SIFT feature matching algorithms for registration, the RMSE values calculated from the registered images and the reference images are high, 4.73 and 4.86, respectively; while the

SSIM values and PSNR values are low, 0.874, 0.841, and 21.48, 20.91, respectively, which indicates that both LoFTR and SIFT are not accurate enough for PCB image feature matching. In terms of quantitative metrics, the local optimization LoFTR algorithm outperforms traditional feature matching algorithms and LoFTR algorithms in PCB image registration.

Figure 5 illustrates the results of PCB image registration using the proposed feature matching algorithm and preprocessing techniques. High registration accuracy can be seen from the superimposed effect of the reference and registered images (see Fig. 5c). Therefore, both visual and quantitative analyses illustrate the excellent performance of this study on PCB image registration, showcasing its significant practical applications.



(a) Reference image (b) Image to be registered (c) Overlay of registered and reference images

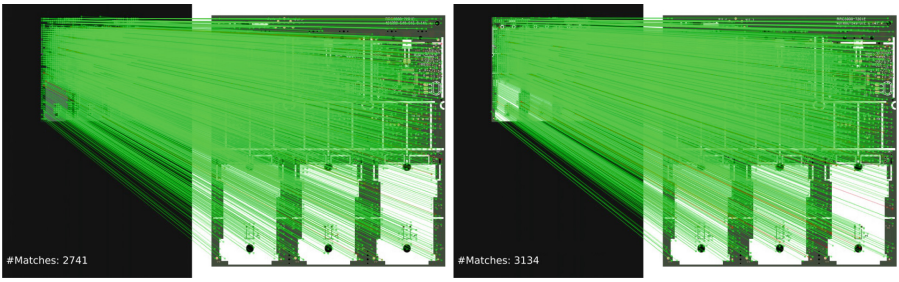
Fig. 5. PCB Image Registration

As shown in Table 2 (the optimal effect is set to bold font, w/o on behalf of the removal of the module, w/ on behalf of having the module), the results of the ablation study verifies the validity of our PCB registration method for each module.

As shown in Fig. 6, after color correcting the PCB to be aligned image, the feature matching is able to get more matching pairs. As shown in Fig. 6a, Matching the uncolor-corrected image with the template image results in 2,741 matching pairs, while matching the color-corrected image with the template image results in 3,134 matching pairs, which is a significant increase (see Fig. 6b). It can also be seen from the figure that the color of the color-corrected image is more similar to the template image, which reduces the interference of chromatic aberration.

Table 2. Ablation Study

Method	SSIM	PSNR	RMSE
w/o MAGSAC++ (w/ RANSAC)	0.883	22.56	4.59
w/o Color Correction	0.879	21.52	4.79
w/o Local Optimization (w/ LoFTR)	0.874	21.48	4.73
Local Optimization LoFTR	**0.891**	**22.86**	**4.56**



(a) without Color Correction (b) with Color Correction

Fig. 6. The Impact of Color Correction on PCB Image Registration

5 Conclusion

This paper proposes an image registration method for large color difference and large size PCBs. Firstly, the color difference problem of printed circuit boards is corrected by the region-based color correction algorithm; for the problem that the LoFTR algorithm loses the information of the first window vectors, a local optimization LoFTR feature matching algorithm for PCB images is proposed. Finally, the MAGSAC++ algorithm is used to match PCB images. Experiments show that the PCB registration method proposed in this paper performs well on PCB images with large color difference and large size.

The feature matching algorithm can be further investigated in the future. In addition, although the current algorithmic process has been relatively efficient, algorithms such as color correction have limitations in terms of time complexity that can be further investigated.

References

1. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. (IJCV) **60**, 91–110 (2004)
2. Mashohor, S., Evans, J.R., Arslan, T.: Image registration of printed circuit boards using hybrid genetic algorithm. In: IEEE International Conference on Evolutionary Computation (CEC), pp. 2685–2690 (2006)

3. Viola, P., Wells, W.M.: Alignment by maximization of mutual information. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 16–23. IEEE (1995)
4. Gruen, A.: Adaptive least squares correlation: a powerful image matching technique. South Afr. J. Photogramm. Remote Sens. Cartography **14**(3), 175–187 (1985)
5. Foroosh, H., Zerubia, J.B., Berthod, M.: Extension of phase correlation to subpixel registration. IEEE Trans. Image Process. (TIP) **11**(3), 188–200 (2002)
6. Ordonez, A., Argüello, F., Heras, D.B.: GPU accelerated FFT-based registration of hyperspectral scenes. IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens. **10**(11), 4869–4878 (2017)
7. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: speeded up robust features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006). https://doi.org/10.1007/11744023_32
8. Rosten, E., Drummond, T.: Machine learning for high-speed corner detection. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 430–443. Springer, Heidelberg (2006). https://doi.org/10.1007/11744023_34
9. Calonder, M., Lepetit, V., Strecha, C., Fua, P.: BRIEF: binary robust independent elementary features. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6314, pp. 778–792. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15561-1_56
10. Dai, L., Guan, Q., Liu, H.: Robust image registration of printed circuit boards using improved SIFT-PSO algorithm. J. Eng. **2018**(16), 1793–1797 (2018)
11. Hua, G., Huang, W., Liu, H.: Accurate image registration method for PCB defects detection. J. Eng. **2018**(16), 1662–1667 (2018)
12. Huang, J., Li, J., Liu, L., Luo, K., Chen, X., Liang, F.: PCB image registration based on a priori threshold surf algorithm. In: Ni, S., Wu, T.Y., Chang, T.H., Pan, J.S., Jain, L. (eds.) Advances in Smart Vehicular Technology, Transportation, Communication and Applications: Proceeding of the Second International Conference on Smart Vehicular Technology, Transportation, Communication and Applications, 25–28 October 2018, Mount Emei, China, Part 1 2, pp. 440–447. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-04582-1_51
13. Li, X., He, H., Huang, C., Shi, Y.: PCB image registration based on improved surf algorithm. In: 2022 International Conference on Image Processing, Computer Vision and Machine Learning (ICICML), pp. 76–79. IEEE (2022)
14. Zhang, Y., Yan, Z., Sun, X., Diao, W., Kun, F., Wang, L.: Learning efficient and accurate detectors with dynamic knowledge distillation in remote sensing imagery. IEEE Trans. Geosci. Remote Sens. **60**, 1–19 (2021)
15. Wang, Y., et al.: A neural corpus indexer for document retrieval. Adv. Neural. Inf. Process. Syst. **35**, 25600–25614 (2022)
16. Fischer, P., Dosovitskiy, A., Brox, T.: Descriptor matching with convolutional neural networks: a comparison to SIFT (2015)
17. Han, X., Leung, T., Jia, Y., Sukthankar, R., Berg, A.C.: MatchNet: unifying feature and metric learning for patch-based matching. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3279–3286. IEEE (2015)
18. Simo-Serra, E., Trulls, E., Ferraz, L., Kokkinos, I., Fua, P., Moreno-Noguer, F.: Discriminative learning of deep convolutional feature point descriptors. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 118–126. IEEE (2015)

19. Choy, C.B., Gwak, J., Savarese, S., Chandraker, M.: Universal correspondence network. In: Advances in Neural Information Processing Systems (NeurIPS), vol. 29 (2016)
20. Yi, K.M., Trulls, E., Lepetit, V., Fua, P.: LIFT: learned invariant feature transform. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 467–483. Springer, Cham (2016)
21. DeTone, D., Malisiewicz, T., Rabinovich, A.: SuperPoint: self-supervised interest point detection and description. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, pp. 224–236. IEEE (2018)
22. Sarlin, P.-E., DeTone, D., Malisiewicz, T., Rabinovich, A.: SuperGlue: learning feature matching with graph neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4938–4947. IEEE (2020)
23. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems (NeurIPS), vol. 30 (2017)
24. Rocco, I., Cimpoi, M., Arandjelović, R., Torii, A., Pajdla, T., Sivic, J.: Neighbourhood consensus networks. In: Advances in Neural Information Processing Systems, vol. 31 (2018)
25. Rocco, I., Arandjelović, R., Sivic, J.: Efficient neighbourhood consensus networks via submanifold sparse convolutions. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12354, pp. 605–621. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58545-7_35
26. Li, X., Han, K., Li, S., Prisacariu, V.: Dual-resolution correspondence networks. In: Advances in Neural Information Processing Systems (NeurIPS), vol. 33, pp. 17346–17357 (2020)
27. Jiang, W., Trulls, E., Hosang, J., Tagliasacchi, A., Yi, K.M.: COTR: correspondence transformer for matching across images. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 6207–6217. IEEE (2021)
28. Sun, J., Shen, Z., Wang, Y., Bao, H., Zhou, X.: LoFTR: detector-free local feature matching with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8922–8931. IEEE (2021)
29. Reinhard, E., Adhikhmin, M., Gooch, B., Shirley, P.: Color transfer between images. IEEE Comput. Graphics Appl. **21**(5), 34–41 (2001)
30. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S.: SLIC superpixels compared to state-of-the-art superpixel methods. IEEE Trans. Pattern Anal. Mach. Intell. **34**(11), 2274–2282 (2012)
31. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**(6), 381–395 (1981)
32. Barath, D., Matas, J., Noskova, J.: MAGSAC: marginalizing sample consensus. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 10197–10205. IEEE (2019)
33. Barath, D., Noskova, J., Ivashechkin, M., Matas, J.: MAGSAC++, a fast, reliable and accurate robust estimator. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1304–1312. IEEE (2020)