sensors

MDPI

Article

A Hardware–Software Integrated PCB Image Registration Method Based on Local Adaptive KNN and SIFT

Wenjie Su [1], En Fan [1,*], Jilong Wang [2], Siyu Ling [3] and Zhaoxi Fang [1]

[1] Institute of Artificial Intelligence, Shaoxing University, Shaoxing 312000, China; 23136123@usx.edu.cn (W.S.); fangzhaoxi@usx.edu.cn (Z.F.)
[2] Zhejiang Ningwei Machinery Technology Co., Ltd., Shaoxing 311814, China; wjl@zjjilong.com
[3] School of Physics, Shaoxing University, Shaoxing 312000, China; 23517210@usx.edu.cn
* Correspondence: efan@usx.edu.cn; Tel.: +86-139-8955-5027

Abstract

Solder-joint detection and localization on large, complex printed circuit boards (PCBs) remain challenging because PCB images often contain unevenly distributed features, nonuniform illumination, specular reflection, scale variation and geometric distortion. Conventional SIFT-based registration methods usually use fixed matching parameters and therefore cannot adapt well to regions with different component densities. To address this limitation, this study proposes a hardware–software integrated PCB image registration framework that combines a robotic end effector with a locally adaptive K-nearest neighbor (LAKNN) strategy and SIFT descriptors. The hardware platform provides stable image acquisition through a lifting mechanism and ring-light illumination, while the software module adjusts the neighbor-search space according to local feature density and matching confidence. Experimental results show that the proposed LAKNN method achieves 90.8% inlier-match accuracy in the ablation experiment and improves registration robustness under height, region and viewpoint variations. The proposed framework provides a practical basis for automated PCB inspection and robotic soldering alignment.

Keywords: PCB; image registration; SIFT; K-nearest neighbors; adaptive threshold; hardware–software integration

Check for updates

Academic Editors: Hong-Dar Lin, Cheng-Hsiung Hsieh and Hsin-Chieh Wu

Received: 22 May 2026
Revised: 4 July 2026
Accepted: 6 July 2026
Published: 1 August 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

With the rapid development of intelligent manufacturing, the production and inspection of electronic products are becoming increasingly automated. As a core component of electronic products, the printed circuit board (PCB) directly affects product reliability, assembly quality and downstream inspection efficiency [1]. Accurate solder-joint localization and PCB image registration are therefore essential for automated soldering and quality control. However, large PCB images usually contain repeated pads, similar textures, uneven feature distributions and scale or viewpoint changes, which make robust registration difficult [2,3].

In PCB inspection, image registration is commonly used to align a reference image, such as a design drawing or standard template, with a sensed image captured during production [4–6]. Existing approaches include template matching, multi-view stereo methods and laser-scanning-based methods [7]. Multi-template and pyramid-based methods can improve local matching of soldering paths [8], while feature-descriptor methods, such as SIFT and ORB, combined with RANSAC can improve robustness under rotation and scale

changes [9]. Nevertheless, these approaches still suffer from parameter sensitivity and mismatches when a large PCB contains repeated local patterns.

Deep learning methods, including convolutional neural networks (CNNs) [10] and object-detection frameworks such as Faster R-CNN [11], have achieved promising results in general detection and matching tasks. However, they are less suitable for the present industrial scenario for two reasons. First, robust training usually requires a large annotated dataset covering board types, solder-joint appearances, lighting conditions and defect variations. Second, inference on edge devices used in cost-sensitive production environments can impose considerable computational and deployment costs. In contrast, the proposed LAKNN–SIFT method is training-free, interpretable and lightweight, making it more suitable for flexible small-batch and high-mix PCB manufacturing.

Feature-point-based registration methods, especially SIFT and SURF [12,13], have been widely used because they extract local key points and descriptors that are relatively robust to scale and rotation. In a typical SIFT workflow, K-nearest neighbor (KNN) matching is used to find candidate correspondences [14], followed by outlier rejection using geometric constraints such as RANSAC [15]. However, conventional KNN uses a fixed K value or a fixed ratio threshold, which is not optimal for large PCB images. Dense pin regions, sparse substrate regions and repeated solder joints have very different local feature distributions, causing either false matches or missed matches when fixed parameters are used [16–18].

In practical large-PCB alignment, two issues are particularly important. First, the PCB may not be fixed at an identical position or viewing angle, resulting in rotation, perspective and scale variations. Second, solder joints and pads often have repeated textures and uneven spatial density, leading to ambiguous descriptor matches [19–21]. To address these problems, this paper proposes an image registration method that integrates SIFT feature extraction with a locally adaptive KNN matching strategy. The main contribution is to adapt the matching process according to local feature density and descriptor-level matching confidence, thereby improving registration accuracy in feature-rich and feature-sparse PCB regions.

The remainder of this paper is organized as follows. Section 2 describes the hardware experimental system and image-processing workflow. Section 3 presents the traditional baseline methods and the proposed LAKNN–SIFT registration method. Section 4 reports the experimental setup, results and ablation analysis. Section 5 concludes the paper and outlines future work.

2. System Architecture and Main Functions

To enable automated soldering and image-guided alignment of large PCBs, this paper develops a robot-based PCB image registration system, as illustrated in Figure 1. The system first captures an overall PCB reference image and local sensed images of target soldering regions. Feature matching is then used to estimate the homography between the reference and sensed images. Finally, the registration result is transferred to the robotic arm to support precise welding. As shown in Figure 2, the software workflow includes image acquisition and preprocessing, SIFT feature extraction, feature matching, homography estimation with mask generation, result-image generation and visual display. A ray-casting-based accuracy assessment module is also implemented to quantify the correctness of matched points.

2.1. Hardware Design and Implementation

To ensure practical applicability, a dedicated hardware platform was developed for stable image acquisition and robotic operation. The platform integrates a six-axis robotic arm, a customized end effector, a laser displacement sensor, an imaging camera module, a

ring-shaped illumination module and an edge-computing/host-PC processing unit into a unified workflow.

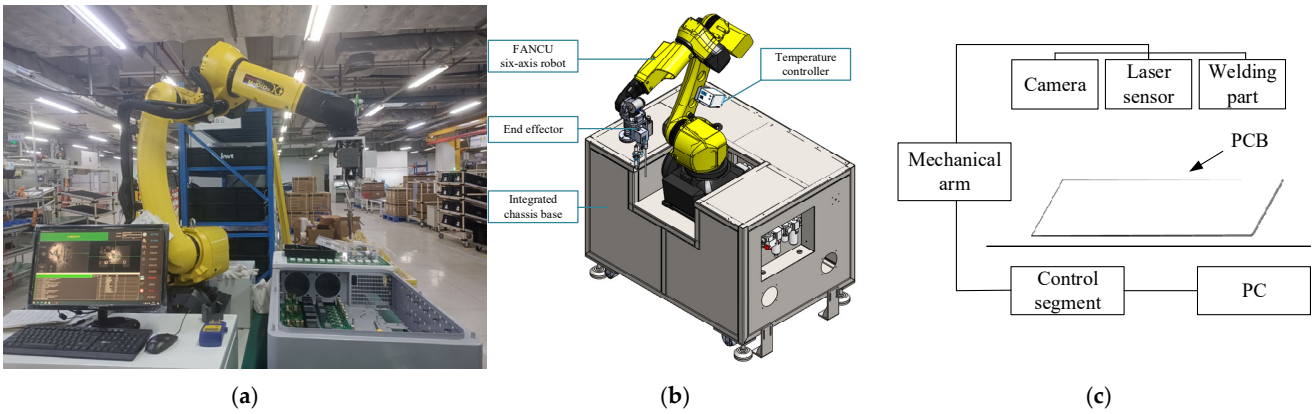

Figure 1. Large-scale PCB robotic welding system; (**a**) physical drawing of the automatic soldering robot; (**b**) schematic diagram of the soldering machine structure; (**c**) soldering system architecture.

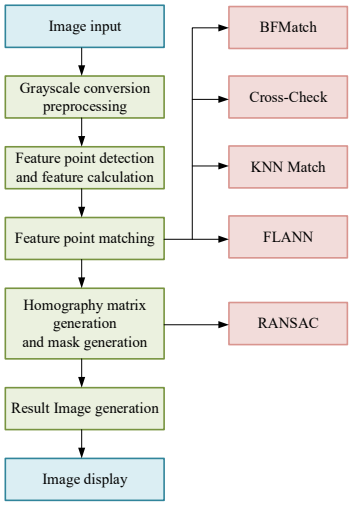

Figure 2. Functional architecture of the system.

The six-axis robotic arm serves as the core actuator. The customized end effector integrates the laser displacement sensor, imaging camera module and soldering unit (Figure 3), enabling height measurement, image acquisition and welding within the same robotic framework. A ring-shaped blue LED array is mounted around the camera lens to provide directional and uniform illumination. This configuration reduces shadows on uneven PCB surfaces, improves contrast on green PCB substrates and suppresses specular reflection, thereby facilitating more stable SIFT feature extraction.

In addition, a lifting mechanism was incorporated into the double-speed conveyor line to temporarily decouple the fixture board from conveyor vibration, thereby providing a stable platform for visual positioning and alignment. Preliminary image preprocessing is performed on the edge-computing unit, while the full LAKNN–SIFT registration and experimental evaluation are executed on the host PC when higher computational resources are required.

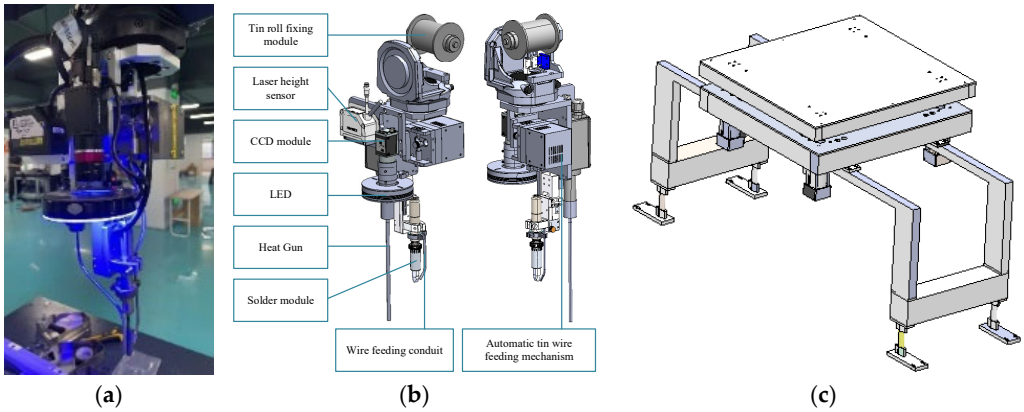

Figure 3. Large-scale PCB robotic welding system: (**a**) physical drawing of the end effector; (**b**) schematic diagram of the end effector structure; (**c**) schematic diagram of the lifting mechanism.

The image acquisition conditions were specified as follows. A Sony FCB-EW9500H color camera (Sony Corporation, Tokyo, Japan) block equipped with a 1/1.8-type STARVIS CMOS sensor was used for image acquisition. To avoid variations caused by automatic optical adjustment, the zoom, focus, exposure and illumination settings were fixed during each group of experiments. The image size used for registration was 2448 × 2048 pixels after cropping, and the exposure time was fixed at 8 ms. Illumination was provided by a ring-shaped blue LED array mounted around the camera lens. The LED array had a peak wavelength of 465 nm, and the illuminance at the PCB surface was approximately 12,000 lux. The effective spatial resolution was estimated according to the calibrated field coverage at each acquisition height. The estimated resolutions corresponding to acquisition heights of 1, 2, 3, 5, 8, 11 and 14 cm were 0.0062, 0.0124, 0.0186, 0.0310, 0.0496, 0.0682 and 0.0868 mm/pixel, respectively.

Before the registration experiments, a preliminary imaging-quality study was performed to validate the acquisition setup. Candidate lighting and acquisition-height settings were compared using the number of detected SIFT key points, the variance of the Laplacian for image sharpness, and the saturated/specular pixel ratio. Under the selected setup, the SIFT detector produced an average of 1847 ± 126 key points per image on the current test image set. The variance of the Laplacian reached 312.6 ± 28.4, indicating sufficient image sharpness, and the saturated/specular pixel ratio was 1.8% ± 0.4%, where saturated pixels were defined as pixels with gray values greater than or equal to 250. These results support the selected acquisition height and lighting configuration.

2.2. Image Acquisition and Processing

In the proposed registration workflow, the reference image is either a design drawing or a standard PCB template, whereas the sensed image is captured during production. Because the present study focuses on structural features rather than color information, the images are converted to grayscale before SIFT feature extraction. This conversion reduces color-channel interference and highlights edges, corners and local intensity changes that are important for image alignment.

(i) Image acquisition: The reference and sensed images are loaded using the image-processing library in BGR format and then converted to grayscale. In the revised experiment description, the camera model, sensor, lighting configuration, acquisition height, image size and spatial resolution are explicitly reported to ensure reproducibility.

(ii) Grayscale preprocessing: Grayscale images contain luminance information only, which reduces computational complexity and makes feature extraction less sensitive to

color variations caused by illumination. This step improves the stability of subsequent SIFT descriptor extraction and matching.

2.3. Feature-Point Detection and Descriptor Extraction

2.3.1. Construction of Scale Space

The SIFT algorithm constructs a scale space by applying Gaussian blurring at multiple levels to generate an image pyramid [22]. Each layer of the pyramid corresponds to a different degree of Gaussian smoothing applied to the original image. The specific scale space is constructed as follows:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{1}$$

Here, $L(x, y, \sigma)$ is the scale-space representation of the image. $G(x, y, \sigma)$ is a Gaussian kernel function. $I(x, y)$ is the original image, $*$ denotes the convolution operation, and σ is the standard deviation of the Gaussian blur that controls the degree of blurring.

2.3.2. Key Point Detection

In the constructed scale space, potential key points are identified by searching for extreme points (i.e., local maxima or minima within the scale-space neighborhood) [23]. Each key point corresponds to a local extremum in scale space, typically associated with regions of high contrast or significant intensity variations. Specifically, each point is compared to its 8 neighbors on the same scale, as well as the 9 corresponding pixels in the scales above and below (3×3 neighborhoods in adjacent scales). If the point is a maximum or minimum among all 26 neighbors, it is considered a potential key point. If this point is a local maximum or minimum across all these neighborhoods, it is identified as a potential key point.

2.3.3. Precise Location of Key Points

For each detected extremum, the SIFT algorithm refines its location by fitting a 3D quadratic function to interpolate the position and scale, achieving sub-pixel and sub-scale accuracy [24]. This process serves to suppress potential false positives and ensure precise localization of the true key point within the image.

2.3.4. Direction Assignment

To achieve rotation invariance, each feature point is assigned a dominant orientation based on the gradient directions in its local neighborhood. Specifically, for each feature point, the histogram of the gradient direction is computed in its neighborhood, and the largest peak in the histogram is selected as the main direction of the point.

The formula for calculating the gradient is as follows:

$$Gradient(x, y) = \left(\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right) \tag{2}$$

Here, $\frac{\partial I}{\partial x}$ and $\frac{\partial I}{\partial y}$ are the gradients of the image in the x and y directions, respectively. The gradient direction is:

$$\theta = atan2\left(\frac{\partial I}{\partial y}, \frac{\partial I}{\partial x} \right) \tag{3}$$

This step ensures that the feature descriptors are invariant to image rotation.

2.3.5. Feature Descriptor Computation

To facilitate image matching, SIFT generates a descriptor for each key point that captures the gradient distribution in its local neighborhood. Typically, the SIFT descriptor

is extracted from a fixed 16 × 16 pixel region centered on the key point. This region is subdivided into 16 smaller blocks, and within each block, an 8-bin histogram of gradient directions is computed. Concatenating these histograms produces a 128-dimensional feature vector that robustly represents the local image structure.

The computation process for each descriptor can be represented as follows:

$$D = [\mathrm{Hist}_1, \mathrm{Hist}_2, \ldots, \mathrm{Hist}_{16}] \tag{4}$$

Here, Hist_i is the gradient orientation histogram of the ith sub-region, containing 8 bins representing accumulated gradient magnitudes in each direction. Eventually, the gradient information of the 16 sub-regions is stitched into a 128-dimensional vector.

2.4. Feature-Point Matching

Feature-point matching is a crucial step in image registration. In this module, several common matching methods are implemented, including BFMatcher [25], FLANN [26] and KNN [27] (see Section 3). Their effectiveness is evaluated experimentally using the same PCB image pairs and the same inlier-match-ratio metric.

2.5. Homography Matrix Generation and Mask Generation

Based on the matched points, the homography matrix—i.e., the projective transformation from the reference image to the sensed image—is estimated using the RANSAC algorithm (see Section 3). RANSAC iteratively samples subsets of the matched point pairs, estimates candidate homography matrices and identifies inliers that are consistent with the projective model [28].

Each inlier correspondence imposes the following homogeneous constraint on the homography matrix:

$$\mathbf{x}_{\mathrm{obj}} \times (\mathbf{H}\mathbf{x}_{\mathrm{ref}}) = 0 \tag{5}$$

Here, $\mathbf{x}_{ref} = [x_{ref}, y_{ref}, 1]^{\mathrm{T}}$ and $\mathbf{x}_{obj} = [x_{obj}, y_{obj}, 1]^{\mathrm{T}}$ denote the homogeneous coordinate vectors of the corresponding points in the reference and sensed images, respectively. $\mathbf{H}$ is the 3×3 homography matrix, and $\times$ denotes the vector cross product. Thus, Equation (5) expresses the collinearity constraint used to estimate the homography matrix from the inlier correspondences.

Mask generation is employed to spatially mark inliers (valid matching points) identified by the RANSAC algorithm. Specifically, a binary mask image is constructed, with pixels corresponding to inliers set to 1 (foreground) and the others to 0 (background). This mask serves two key roles: (i) it filters out outliers during the homography matrix application, ensuring only reliable feature points contribute to the image transformation; (ii) it facilitates visualization of matching quality by highlighting valid correspondences, aiding in qualitative evaluation of registration results. The mask is generated synchronously with homography matrix estimation using the inlier indices from RANSAC to map valid points to the image coordinate system.

2.6. Resulting Image Generation

The estimated homography matrix defines the forward mapping from the reference-image coordinate system to the sensed-image coordinate system. For a point in the reference image, the corresponding point in the sensed image is obtained by applying the homography matrix and normalizing the resulting homogeneous coordinates:

$$\mathbf{x}_{\mathrm{obj}} = \frac{\mathbf{H}\mathbf{x}_{\mathrm{ref}}}{(\mathbf{H}\mathbf{x}_{\mathrm{ref}})_3} \tag{6}$$

Equation (6) obtains the Cartesian image coordinates by normalizing the transformed homogeneous vector with its third component. To generate the final registered image, the sensed image is warped into the reference-image coordinate system using the inverse homography matrix. The transformed image can then be superimposed on the reference image using an image-fusion method, such as weighted averaging. The inverse mapping is expressed as follows:

$$\mathbf{x}_{ref} = \mathbf{H}^{-1} \cdot \mathbf{x}_{obj} \tag{7}$$

2.7. Accuracy Assessment Based on the Ray Method

To evaluate alignment accuracy, the ray-casting method [29] is employed to determine whether matched points fall within a specified region of the sensed image. The ray-casting method determines whether a point falls within a region by shooting a ray from the point to be tested and checking whether it intersects with the boundary of the target region. The distance from the point to the boundary of the target region is calculated and compared with a set threshold. If the distance from the point to the region boundary is below the threshold, the point is considered a correct match. The matching accuracy is then computed as the ratio of correct matches to total matches:

$$\text{Accuracy} = \frac{\text{correct matches}}{\text{total matches}} \times 100\% \tag{8}$$

where "correct matches" refers to the number of point pairs that meet the distance threshold and "total matches" is the total number of matched point pairs.

3. Proposed Image Registration Method

3.1. Introduction to Traditional Image Registration Methods

3.1.1. Brute-Force Matching (BF Match)

The brute-force matching method is a fundamental method for matching 2D feature points. For each descriptor R_i in the first image, brute-force matching computes its distance to all descriptors $\{S_1, S_2, \ldots, S_n,\}$ in the second image and then selects the most similar descriptor S_j as the candidate match. The Euclidean distance is used to measure similarity between descriptors:

$$d_{\text{Euclidean}}(R_i, S_j) = \sqrt{\sum_{k=1}^{d} \left(R_i(k) - S_j(k)\right)^2} \tag{9}$$

Here, d is the dimension of the descriptor, and $R_i(k)$ and $S_j(k)$ denote the value of the kth element in the descriptors R_i and S_j, respectively. The accuracy of matching can be improved to some extent by methods such as cross-validation or setting distance thresholds.

3.1.2. Cross-Matching (Cross-Check)

Cross-check is a strategy used to improve feature-matching accuracy, aiming to filter out incorrect pairs of matches, especially reducing the number of "one-to-many" or "many-to-one" wrong matches. The basic idea is to perform reverse verification after brute-force matching is completed. Specifically, if the descriptor R_i in the reference image and the descriptor $S_j$in the sensed image are the best match for each other (i.e., R_i is the best match for S_j, and vice versa), they are considered as correct matches. Through reverse verification, cross-validation can effectively exclude matches that occur only in one-way matching, thereby reducing false matches, especially when duplicate features or background interference are present in the image.

3.1.3. Fast Library for Approximate Nearest Neighbors (FLANN)

FLANN is a library designed for fast approximate nearest neighbor searches in large-scale datasets. By constructing efficient data structures, FLANN can greatly accelerate the matching process [30]. Compared to brute-force matching, FLANN demonstrates superior performance on large datasets and is especially well-suited for real-time applications. It commonly employs k-d tree indexing to enable efficient nearest neighbor searches, where the data space is recursively partitioned along different dimensions to speed up the search process [31]. Given a dataset $D = \{x_1, x_2, \ldots, x_n\}$ and a query point $q \in R^k$, the k-d tree is constructed as follows:

(i) Selection of dividing dimensions: one dimension $d \in \{1, 2, \ldots, k\}$ is selected for each layer, step by step, in a cyclic manner;

(ii) Node construction: sort the data points along the selected dimension, and choose the median x_m as the node, which partitions the space into left and right subtrees;

(iii) Recursive splitting: repeat the above for the left and right subtrees until the stopping condition is satisfied. The query process aims to find the nearest neighbor x_{NN} of the query point q, minimizing the distance function:

$$d(q, x_{NN}) = \min_{x_i \in D} \parallel q - x_i \parallel) \tag{10}$$

Here, $\parallel q - x_i \parallel$ is the Euclidean distance as follows:

$$\parallel q - x_i \parallel = \sqrt{\sum_{d=1}^{k} (q_d - x_{id})^2} \tag{11}$$

By traversing the k-d tree, FLANN can achieve a fast approximate nearest neighbor search.

3.1.4. Random Sample Consensus (RANSAC)

RANSAC is a classical parameter estimation method for robust model fitting in the presence of outliers. The basic idea is to iteratively sample random subsets of the data to estimate a model and then identify inliers—points consistent with the model—by evaluating the model against the entire dataset.

In feature matching, RANSAC can effectively remove the wrong matching points. The process includes calculating the single response matrix between two images based on the matching points and further decomposing the rotation matrix R and translation vector t of the camera [32]. The 3D coordinates of the matching points are estimated via triangulation. These 3D points are then projected back onto the image plane, and the reprojection error is computed. If the error between the projected coordinates and the original observation is below a predefined threshold, the matched point pair is classified as an inlier.

3.1.5. K-Nearest Neighbors Matching (KNN Match)

The K-nearest neighbor matching algorithm is used to improve the matching accuracy by selecting the first k-nearest neighbors of the feature points and performing distance-ratio filtering. For each reference feature point R_i, the algorithm first identifies the two nearest neighbors in the sensed image: S_{j1} (the closest match) and S_{j2} (the second closest match). The pair (R_i, S_{j1}) is accepted as a valid match only if the distance ratio between the optimal and suboptimal matches is below a predefined threshold $t \in [0.5, 0.9]$. This ratio test helps ensure matching distinctiveness.

The distance ratio is computed for each reference feature point R_i and its two nearest neighbors, S_{j1} (the nearest neighbor) and S_{j2} (the second-nearest neighbor), in the target image:

$$r_i = \frac{d(R_i, S_{j1})}{d(R_i, S_{j2})} \tag{12}$$

where $d(R_i, S_{j1})$ and $d(R_i, S_{j2})$ denote the Euclidean distances from R_i to its nearest and second-nearest neighbors, respectively. The binary matching-decision indicator A_i is defined as

$$A_i = \begin{cases} 1, & \text{if } r_i < t \\ 0, & \text{if} r_i \geq t \end{cases} \tag{13}$$

where t is the predefined distance-ratio threshold. A_i = 1 indicates that the candidate match (R_i, S_{j1}) is accepted, whereas A_i = 0 indicates that it is rejected. A smaller distance ratio means that the nearest neighbor is substantially more similar to R_i than the second-nearest neighbor, thereby reducing ambiguous matches and improving matching reliability.

3.2. PCB Image Registration Method Based on LAKNN and SIFT

In traditional KNN-based image registration methods, a fixed K value is commonly used. However, when applied to complex PCB images, this fixed setting may reduce matching accuracy. To address this issue, this paper presents an optimized KNN registration algorithm that adaptively adjusts the K value based on local density estimation and matching confidence. The proposed method proceeds through four main steps: (i) SIFT feature extraction, where feature points and their descriptors are extracted from the images; (ii) LAKNN matching, in which the K value is dynamically adjusted according to local feature density and matching confidence; (iii) mismatch elimination using the RANSAC algorithm; and (iv) affine transformation and final registration output.

3.2.1. Adaptive K Value Collaborative Adjustment Strategy

Specifically, the kernel density estimation method is adopted to estimate the local density of a certain feature point x, and the K value is dynamically adjusted in combination with its matching confidence [33]. The local density estimation is calculated using the Gaussian kernel function as follows:

$$\hat{\rho}(x) = \frac{1}{n}\sum_{i=1}^{n} \exp\left(-\frac{\| x - x_i \|^2}{2h^2}\right) \tag{14}$$

Here, x_i is the neighbor of the feature point x, and h is the bandwidth parameter of the kernel function. This density reflects the local density of the current feature points in space.

Matching confidence is used to measure the degree of credibility between a certain feature point and its best matching point and is defined as follows:

$$C_i = 1 - \frac{d_1}{d_2} \tag{15}$$

Here, d_1 and d_2 represent the Euclidean distances between the current point and its best match and suboptimal match. When C_i is larger, it indicates a more reliable match.

The joint adjustment strategy for the K value is as follows:

$$k_{local}(x) = round(k_{base}(1 + \alpha(1 - \hat{\rho}(x)))(1 - \beta C_i)) \tag{16}$$

Here, k_{base} is the base K value, and α and β are hyperparameters used to balance the influence of local density and matching confidence.

This strategy achieves a decrease in K when the feature-point density is high and an increase in K when the matching confidence is low, thereby enhancing the adaptability of KNN matching to different image regions. The advantage of the proposed strategy over standard nearest neighbor (NN) matching or fixed ratio tests is particularly evident in complex PCB images. The feature density varies drastically between integrated circuit (IC) pins (high density) and empty substrates (low density). A fixed k value or a static ratio threshold inevitably leads to either missed matches in dense areas or false positives in sparse areas. The proposed LAKNN strategy effectively relaxes constraints in sparse regions and tightens in dense regions, mimicking human visual confirmation mechanisms.

Regarding the hyperparameters in Equation (16), α and β determine the sensitivity of the adaptive k value to the local density (ρ_x) and matching confidence (C_i), respectively. These parameters were determined empirically through a grid search experiment on a validation set consisting of 50 representative PCB images. During the optimization process, we varied $\alpha \in [0.5,\ 2.0]$ and $\beta \in [0.1,\ 0.9]$. The optimal combination ($\alpha = 1.0,\ \beta = 0.5$) was selected as it maximized the inlier-match ratio while maintaining a recall rate above 85%.

3.2.2. Characteristic Distance Calculation and Matching Discrimination

To evaluate the similarity between SIFT feature points, this paper adopts Euclidean distance as the measurement method of feature descriptors. For two N-dimensional SIFT feature descriptors f_1 and f_2 ($N = 128$), the distance is defined as:

$$d(f_1, f_2) = \sqrt{\sum_{j=1}^{N} (f_{1j} - f_{2j})^2} \tag{17}$$

In KNN matching, we find the top K neighbors with the smallest distances for each feature point and then use the distance-ratio criterion for validity screening. Only when the following conditions are satisfied will the matched pair be retained:

$$\frac{d_1}{d_2} < \tau \tag{18}$$

Here, the distance-ratio threshold τ was set to 0.75. This setting follows Lowe's classical SIFT ratio-test principle [12], in which a candidate match is accepted only if the nearest neighbor descriptor distance is sufficiently smaller than the second-nearest neighbor distance. To verify its suitability for PCB images, a threshold-sensitivity experiment was conducted at τ = 0.65, 0.70, 0.75, 0.80 and 0.85. The results showed that τ = 0.75 provided the best overall trade-off between false matches and missed matches under the present PCB imaging conditions, achieving an inlier ratio of 81.3%, a false-match rate of 6.2% and a recall of 89.7%. Although larger thresholds retained more candidate matches, they also introduced more ambiguous correspondences, while smaller thresholds removed many valid matches and reduced recall. Therefore, τ = 0.75 was used in all subsequent experiments.

In summary, the proposed LAKNN–SIFT method not only maintains high precision but also significantly enhances robustness against complex PCB images with rotations, scaling variations, and multi-view scenarios. Its core innovation lies in the LAKNN strategy, which jointly leverages kernel density estimation of feature distributions and confidence-guided thresholding to dynamically adjust the matching process. Unlike conventional fixed-parameter KNN approaches, this adaptive mechanism fully accounts for local differences in feature-point distributions and matching confidence, ensuring stable performance across diverse and challenging PCB registration conditions.

From an industrial perspective, the proposed method is embedded into a hardware–software co-designed platform that includes a robotic arm, adaptive illumination, and edge computing units. This integration enables closed-loop PCB inspection and auto-

mated soldering operations, demonstrating the practical feasibility of the approach beyond algorithmic enhancement.

4. Experimental Verification

4.1. Dataset, Target PCB and Evaluation Setup

The target specimen was a four-layer FR4 industrial control motherboard with physical dimensions of 380 mm × 280 mm × 1.6 mm. It contained QFP packages, BGA components, 0402 passive elements, through-hole connectors, repeated solder pads, connector-pin arrays, and regions with both dense and sparse feature distributions. This PCB was selected for its large size and highly heterogeneous component density. It includes dense IC pad clusters alongside sparse connector regions. These characteristics provide a representative and rigorous test case for evaluating adaptive feature-matching performance under realistic industrial conditions. The reference image was obtained from a whole-board scan, whereas the sensed images were captured using the camera system under controlled acquisition-height and viewpoint conditions. The four local regions in Figure 4 were selected to cover feature-rich, feature-sparse and nonuniform feature-distribution cases.

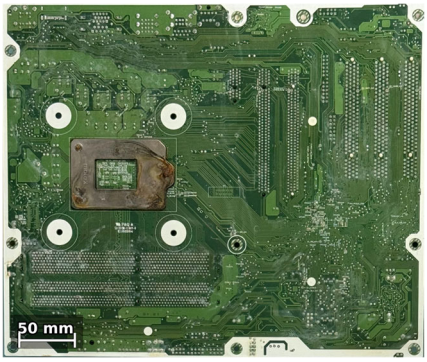

Figure 4. Reference image.

In total, 20 unique image pairs were used in the main comparative experiments, comprising 7 pairs for acquisition-height evaluation, 4 pairs for regional robustness evaluation, and 9 viewpoint pairs. The 9 viewpoint pairs were evaluated in both forward and reverse matching directions at angles of 0°, 2°, 3°, 5°, 10°, 15°, 20°, 25° and 30°, resulting in 18 registration evaluations. Thus, the complete experimental suite includes 29 registration evaluations in total. For the representative PCB images, scale bars were added according to the image type and the calibrated spatial resolution. Board-level scale bars were used for whole-board, registration-effect and region-selection images, while local-scale bars were used for local sensed images. The spatial resolution was reported according to the corresponding acquisition height. This experimental design enabled a systematic evaluation of the proposed LAKNN method under changes in image scale, local feature density and viewpoint deflection.

All computations were performed on an Intel Core i7-12700H CPU @ 2.30 GHz with 32 GB DDR5 memory. No dedicated GPU acceleration was used for the registration pipeline. The experiments were conducted under Ubuntu 20.04 LTS using Python 3.9 and OpenCV 4.8.0. Runtime was measured as the average over five independent runs for BF, BF_CrossCheck, FLANN, KNN_Low, KNN_High and LAKNN under identical hardware and software conditions.

4.2. Experiment 1: PCB Images for Different Heights

In this experiment, the reference target described in Section 4.1 was used, as shown in Figure 4. The reference image was obtained by scanning the whole PCB at 600 dpi, and the image size used for registration was 2448 × 2048 pixels after cropping. The sensed images were captured using a Sony FCB-EW9500H color camera block (Sony Corporation, Tokyo, Japan) equipped with a 1/1.8-type STARVIS CMOS sensor. To ensure consistent acquisition conditions, the zoom, focus, exposure and illumination settings were fixed during each group of experiments. The spatial resolutions corresponding to the tested acquisition heights are reported in Section 2.1.

Figure 5 shows a representative sensed image captured at an acquisition height of 5 cm. Figure 6 illustrates the corresponding registration result, in which the sensed image is transformed into the reference-image coordinate system. The close overlap of the board contours and component structures provides a qualitative indication that the estimated homography yields effective alignment under this representative acquisition condition.

Figure 5. Sensed image with a height of 5 cm.

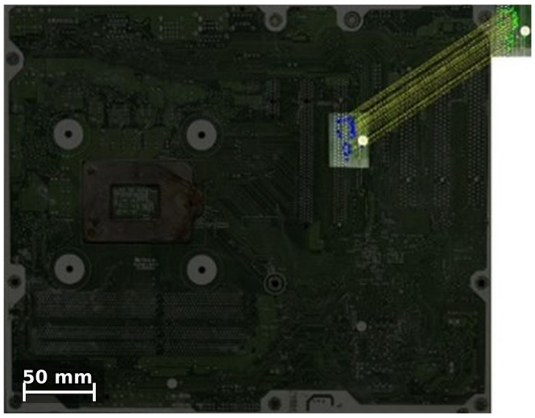

Figure 6. PCB registration effect diagram (with a height of 5 cm as an example).

In practical applications, the camera installation height affects image scale, feature size and registration accuracy [34]. Therefore, sensed images were captured at *Z*-axis acquisition heights of 1 cm, 2 cm, 3 cm, 5 cm, 8 cm, 11 cm and 14 cm. Scale bars were added to the representative PCB images according to the spatial scale of each figure, and the inlier-match ratio was used as the primary evaluation metric.

From Figure 7 and Table 1, the proposed LAKNN method achieves the highest inlier ratio across all tested heights with a clear peak around 5–8 cm (81.8% and 70.0%). Performance degrades at extreme close-range and long-range acquisitions—expected due to scale variation and loss of detail—but LAKNN still maintains better robustness than all baseline methods across the tested range. The arithmetic mean across the seven discrete heights is 49.3%.

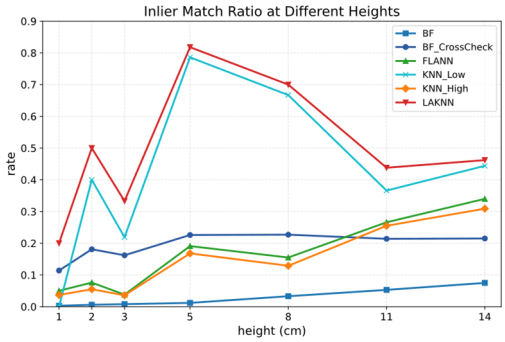

Figure 7. Inlier-match accuracy point-line diagram at different heights.

Table 1. Inlier-match accuracy record at different heights.

Rate	BF	BF_CrossCheck	FLANN	KNN_Low	KNN_High	LAKNN
1 cm	0.3%	11.4%	5.0%	0.0%	3.7%	**20.0%**
2 cm	0.6%	18.1%	7.6%	40.0%	5.5%	**50.0%**
3 cm	0.8%	16.2%	3.8%	21.9%	3.6%	**33.3%**
5 cm	1.2%	22.6%	19.1%	78.6%	16.8%	**81.8%**
8 cm	3.3%	22.7%	15.5%	66.7%	12.9%	**70.0%**
11 cm	5.3%	21.4%	26.6%	36.6%	25.5%	**43.8%**
14 cm	7.5%	21.5%	34.0%	44.4%	30.9%	**46.2%**
Average	2.7%	19.1%	15.9%	41.2%	14.1%	**49.3%**

4.3. Experiment 2: PCB Images for Different Regions

To verify the robustness of the proposed LAKNN method, four different regions of the reference image were also selected for this study, as shown in Figure 8. Six image registration methods were used for comparison in the experiment. The results are shown in Figure 9 and Table 2. It is noteworthy that KNN_Low achieves competitive results in Areas 3 and 4 (85.0% and 85.7%). This is likely because these regions contain highly distinctive, locally clustered features that align well with a small, fixed neighborhood size. However, its performance drops drastically in Area 2 (52.0%) and under viewpoint changes (Table 3), confirming that a fixed K value lacks the robustness required for heterogeneous PCBs. This observation further validates the necessity of the proposed adaptive strategy.

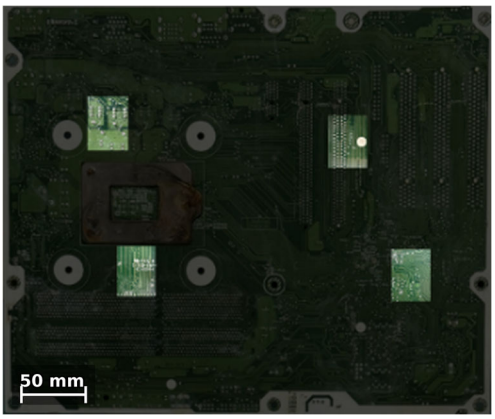

Figure 8. PCB different area selection diagram.

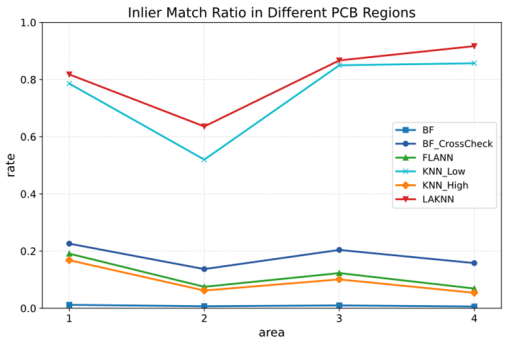

Figure 9. Inlier-match accuracy point-line graph in different areas.

Table 2. Inlier-match accuracy record in different areas.

Rate	BF	BF_CrossCheck	FLANN	KNN_Low	KNN_High	LAKNN
Area1	1.2%	22.6%	19.1%	78.6%	16.8%	**81.8%**
Area2	0.7%	13.7%	7.5%	52.0%	6.2%	**63.6%**
Area3	1.0%	20.4%	12.3%	85.0%	10.1%	**86.7%**
Area4	0.6%	15.8%	6.9%	85.7%	5.4%	**91.7%**
Average	0.9%	18.1%	11.5%	75.3%	9.6%	**81.0%**

Table 3. Inlier-match accuracy record from different perspectives (reverse).

Rate	BF	BF_CrossCheck	FLANN	KNN_Low	KNN_High	LAKNN
0°	21.0%	22.6%	75.0%	N/A	70.8%	**90.0%**
2°	21.6%	22.1%	68.4%	N/A	71.8%	**88.2%**
3°	23.1%	22.1%	68.4%	N/A	69.2%	**77.8%**
5°	11.2%	13.9%	63.9%	N/A	59.5%	**66.7%**
10°	9.7%	10.9%	42.9%	N/A	44.4%	**66.7%**
15°	12.1%	13.6%	57.7%	N/A	56.5%	**60.0%**
20°	13.5%	15.3%	62.9%	N/A	68.3%	**92.3%**
25°	10.8%	13.1%	57.1%	N/A	61.2%	**87.3%**
30°	6.9%	9.4%	34.7%	N/A	38.9%	**73.8%**
Average	14.4%	15.9%	59.0%	N/A	60.1%	**78.1%**

4.4. Experiment 3: PCB Images from Different Perspectives

To address potential camera viewing-angle deflection in practical applications, this study also investigates the accuracy of various methods across different viewing angles [35]. In practical application, when aligning the components or the details of the solder joints, the *Z*-axis acquisition height is usually about 5–8 cm, so 5 cm was selected as the standard height for testing viewing-angle variations, which were also evaluated across six experimental groups.

In image registration, perspective deflection frequently hinders feature-point matching, especially under large-angle deflection, where the mismatch rate increases. To systematically evaluate robustness against viewpoint changes, both forward and reverse matching were performed. In reverse matching, the reference image and sensed image are exchanged, which provides a control experiment for evaluating descriptor consistency and registration stability under viewpoint deflection. Additional measurements at 25° and 30° were included to verify whether the accuracy trend continues beyond 20°.

The results in Figure 10 and Table 3 show that the accuracy of traditional matching methods, such as BF and KNN_High, gradually decreases as the viewing angle increases from 0° to 30°. At a 20° reverse-viewing-angle deflection, the BF and KNN_High accuracies are 13.5% and 68.3%, respectively, while the proposed LAKNN method reaches 92.3%.

The additional measurements at 25° and 30° further show that LAKNN maintains higher accuracy than the baseline methods, with inlier-match ratios of 87.3% and 73.8%, respectively. The KNN_Low method failed to produce stable matches in the reverse matching experiments, as indicated by N/A in Table 3. These results demonstrate that LAKNN provides more reliable registration under large viewpoint deflections.

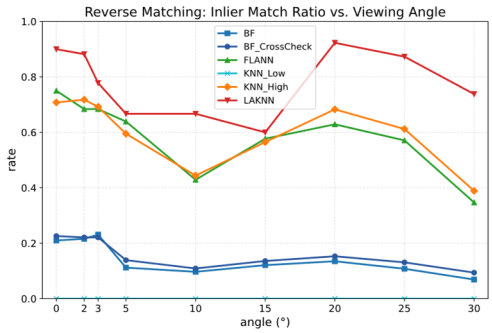

Figure 10. Inlier-match accuracy point-line graph from different perspectives (reverse).

In the forward matching experiments, Figure 11 and Table 4 show that LAKNN performs well under small viewing-angle deflections from 0° to 5°, with an inlier-match accuracy of 81.8% at 0° and 83.3% at 5°. As the viewing angle increases, the forward matching accuracy decreases; at 20°, it remains 72.7%, and at 25° and 30° conditions, it reaches 68.9% and 61.7%, respectively. Although the forward accuracy is lower than the reverse-matching result at large angles, it remains higher than or comparable to the other traditional methods.

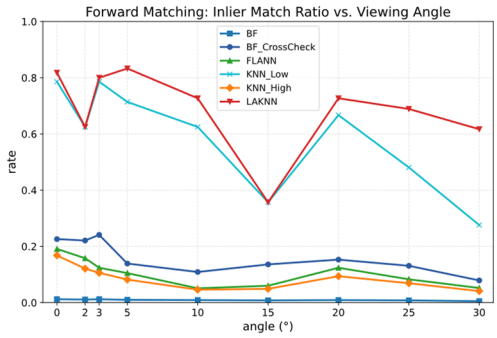

Figure 11. Inlier-match accuracy point-line graph from different perspectives (positive).

Table 4. Inlier-match accuracy record from different perspectives (positive).

Rate	BF	BF_CrossCheck	FLANN	KNN_Low	KNN_High	LAKNN
0°	1.2%	22.6%	19.1%	78.6%	16.8%	**81.8**%
2°	1.1%	22.1%	15.8%	62.5%	12.1%	**62.5**%
3°	1.2%	24.1%	12.4%	78.6%	10.6%	**80.0**%
5°	1.0%	13.9%	10.5%	71.4%	8.2%	**83.3**%
10°	0.9%	10.9%	5.1%	62.5%	4.6%	**72.7**%
15°	0.8%	13.6%	6.0%	35.7%	4.9%	**35.7**%
20°	0.9%	15.3%	12.4%	66.7%	9.4%	**72.7**%
25°	0.8%	13.1%	8.3%	48.1%	6.9%	**68.9**%
30°	0.5%	7.9%	5.2%	27.6%	4.1%	**61.7**%
Average	0.9%	15.9%	10.5%	59.1%	8.6%	**68.8**%

Overall, forward matching with LAKNN performs well under small-angle deflections (0°–5°), while reverse matching shows stronger robustness under larger angles (≥15°).

This suggests that two-way verification could further improve registration robustness in practice.

4.5. Ablation Experiment

To validate the effectiveness of the adaptive K-value strategy and matching confidence adjustment mechanism in the proposed LAKNN algorithm, this section conducts an ablation study. The LAKNN-full method is compared with its simplified variants to evaluate the individual contributions of key modules to the overall performance.

Four methods' variants were designed for comparison: (1) LAKNN-Full. The LAKNN-full method incorporates both the adaptive K-value adjustment and matching confidence weighting strategies. (2) LAKNN-FixedK. A variant with the adaptive K-value strategy removed (K is fixed at 2), but the matching confidence mechanism is retained. (3) LAKNN-NoConf. Variant with the matching confidence strategy removed while retaining the adaptive K-value adjustment. (4) KNN-Basic. The conventional KNN-based matching method has a fixed K-value and no matching confidence mechanism.

All variants were evaluated on 20 registration trials captured at a height of 5 cm under standard viewing conditions. The inlier-match ratio was used as the primary accuracy metric, and average matching time was measured over five independent trials to reflect computational efficiency. The same host-PC configuration described in Section 4.1 was used for all variants.

The ablation results in Table 5 and Figure 12 show that both the adaptive K-value strategy and the matching-confidence mechanism improve registration performance. Removing either module leads to a decline in accuracy, confirming their individual contributions. Compared with KNN-Basic, LAKNN-NoConf improves accuracy by 7.9 percentage points, LAKNN-FixedK improves accuracy by 5.9 percentage points, and the full LAKNN method improves accuracy by 12.2 percentage points, reaching 90.8%. Although the full model requires slightly more matching time, the accuracy gain is important for PCB inspection tasks where incorrect localization may lead to welding errors.

Table 5. Ablation results and approximate matching times of LAKNN variants.

Method	Adaptive K	Confidence Weighting	Inlier Accuracy	Avg. Matching Time	Improvement
KNN-Basic	No	No	78.6%	0.095 s	Baseline
LAKNN-FixedK	No	Yes	84.5%	0.102 s	+5.9 p.p.
LAKNN-NoConf	Yes	No	86.5%	0.108 s	+7.9 p.p.
LAKNN-Full	Yes	Yes	90.8%	0.112 s	+12.2 p.p.

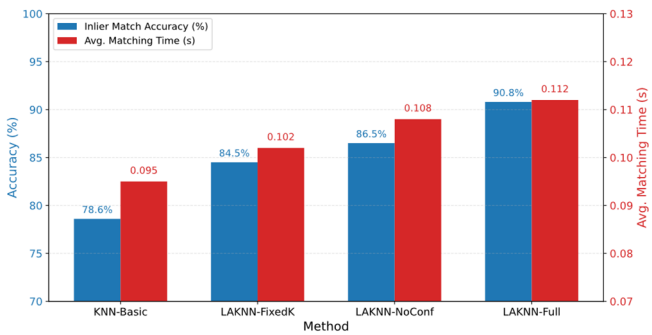

Figure 12. Performance comparison of LAKNN variants in the ablation study. The average matching time includes natural fluctuations caused by variations in local feature distributions and hardware processing load. Each result is averaged over five independent trials.

To assess the computational overhead of the proposed method against the six matching baselines, Table 6 compares their mean matching times under identical hardware conditions. LAKNN requires 0.112 s per match, which is 39.8% faster than BF and only 0.016 s slower than KNN_Low. Although FLANN is the fastest method (0.082 s), the accuracy results in Sections 4.2–4.4 show that its lower runtime is accompanied by lower inlier-match ratios. Thus, LAKNN provides a practical balance between registration accuracy and computational cost.

Table 6. Computational cost comparison under identical hardware conditions.

Method	Mean Matching Time	Relative Change vs. BF
BF	0.186 s	Baseline
BF_CrossCheck	0.228 s	+22.6%
FLANN	0.082 s	−55.9%
KNN_Low	0.096 s	−48.4%
KNN_High	0.101 s	−45.7%
LAKNN	0.112 s	−39.8%

4.6. Discussion of Hardware–Software Synergy

The superior registration performance reported in this study is attributed to the synergy between stable image acquisition and algorithmic adaptation. The lifting mechanism reduces platform vibration and motion blur, while the ring-shaped blue LED illumination reduces shadows and specular reflection. These hardware improvements provide higher-quality input images for SIFT feature extraction. However, hardware improvements alone do not resolve geometric distortion and repeated local patterns. As shown in Table 5, KNN-Basic reaches 78.6% accuracy under the same acquisition conditions, whereas the full LAKNN method reaches 90.8%. Therefore, the hardware ensures data stability, and the LAKNN algorithm improves final registration precision. To quantify the hardware contribution, blur and reflection were measured using the variance of the Laplacian and the saturated-pixel ratio, respectively. Compared with a single fixed white-light source without the ring-shaped blue LED array, the proposed setup changed these metrics from 184.3 ± 21.7 and 7.6% ± 1.1% to 312.6 ± 28.4 and 1.8% ± 0.4%, respectively. This indicates improved sharpness and a lower proportion of saturated/specular pixels under the proposed acquisition setup.

5. Conclusions and Future Work

This paper proposes a hardware–software integrated PCB image registration method based on LAKNN and SIFT. By combining stable robotic image acquisition with locally adaptive feature matching, the method improves registration robustness under height variation, regional feature-density variation and viewing-angle deflection. The experimental results show that LAKNN achieves higher inlier-match ratios than BF, FLANN and fixed-parameter KNN baselines under the tested conditions.

Future work will focus on three directions. First, the hardware platform will be deployed on real PCB assembly lines for long-term validation under production conditions. Second, a larger and more diverse PCB dataset will be constructed, covering different board types, component layouts, lighting conditions and manufacturers. Third, controlled experiments under different blur, reflection and production-vibration levels will be further expanded to evaluate robustness and computational cost in real industrial environments.

Author Contributions: Conceptualization, W.S. and E.F.; methodology, W.S.; software, W.S. and Z.F.; validation, W.S., S.L. and J.W.; formal analysis, W.S. and S.L.; investigation, W.S.; resources, E.F.; data curation, Z.F. and J.W.; writing—original draft preparation, W.S.; writing—review and editing, E.F.

and Z.F.; visualization, W.S.; supervision, E.F.; project administration, E.F.; funding acquisition, E.F. and J.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was financially supported by the Natural Science Foundation of Zhejiang Province (No. LGG22F010004), and National Innovation and Entrepreneurship Training Program for College Students (No. 202510349035).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: Author Jilong Wang was employed by the company Zhejiang Ningwei Machinery Technology Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Ulger, F.; Yuksel, S.E.; Yilmaz, A.; Gokcen, D. Solder joint inspection on printed circuit boards: A survey and a dataset. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 2515121. [CrossRef]
2. Li, C.Y.; Zhang, P.; Wang, S.M.; Liu, L.; Shi, M.Q. Industrial fusion cascade detection of solder joint. *Comput. Mater. Contin.* **2024**, *81*, 1197–1214. [CrossRef]
3. Si, S.; Li, Z.; Lin, Z.; Xu, X.; Zhang, Y.; Xie, S. A robust 2D/3D medical image registration based on feature-point matching. *IEEE Trans. Instrum. Meas.* **2024**, *73*, 5037609. [CrossRef]
4. Zheng, J.; Peng, W.; Wang, Y.; Zhai, B. Accelerated RANSAC for accurate image registration in aerial video surveillance. *IEEE Access* **2021**, *9*, 36775–36790. [CrossRef]
5. Hu, B.; Wang, J. Detection of PCB surface defects with improved faster-RCNN and feature pyramid network. *IEEE Access* **2020**, *8*, 108335–108345. [CrossRef]
6. Dai, L.; Guan, Q.; Liu, H. Robust image registration of printed circuit boards using improved SIFT-PSO algorithm. *J. Eng.* **2018**, *2018*, 1793–1797. [CrossRef]
7. Ling, Q.; Isa, N.A.M. Printed circuit board defect detection methods based on image processing, machine learning and deep learning: A survey. *IEEE Access* **2023**, *11*, 15921–15944. [CrossRef]
8. Dai, W.; Mujeeb, A.; Sourin, A. Soldering defect detection in automatic optical inspection. *Adv. Eng. Inform.* **2020**, *43*, 100975. [CrossRef]
9. Hossein-Nejad, Z.; Nasri, M. An adaptive image registration method based on SIFT features and RANSAC transform. *Comput. Electr. Eng.* **2017**, *62*, 524–537. [CrossRef]
10. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef] [PubMed]
11. Sushma Sri, M.; Rajendra Naik, B.; Jaya Sankar, K. Object detection based on Faster R-CNN. *Int. J. Eng. Adv. Technol.* **2021**, *10*, 72–76. [CrossRef]
12. Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
13. Bay, H.; Ess, A.; Tuytelaars, T.; Van Gool, L. Speeded-up robust features (SURF). *Comput. Vis. Image Underst.* **2008**, *110*, 346–359. [CrossRef]
14. Yong, A.; Hong, Z. SIFT matching method based on K nearest neighbor support feature points. In Proceedings of the 2016 IEEE International Conference on Signal and Image Processing (ICSIP), Beijing, China, 13–15 August 2016; pp. 64–68. [CrossRef]
15. Martínez-Otzeta, J.M.; Rodríguez-Moreno, I.; Mendialdua, I.; Sierra, B. RANSAC for robotic applications: A survey. *Sensors* **2023**, *23*, 327. [CrossRef] [PubMed]
16. Muja, M.; Lowe, D.G. Scalable nearest neighbor algorithms for high dimensional data. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 2227–2240. [CrossRef] [PubMed]
17. Tsourounis, D.; Kastaniotis, D.; Theoharatos, C.; Kazantzidis, A.; Economou, G. SIFT-CNN: When convolutional neural networks meet dense SIFT descriptors for image and sequence classification. *J. Imaging* **2022**, *8*, 256. [CrossRef] [PubMed]
18. Zhao, W.; Tabak, E.G. Adaptive kernel conditional density estimation. *Inf. Inference J. IMA* **2025**, *14*, iaae037. [CrossRef]
19. Bhattacharya, A.; Cloutier, S.G. End-to-end deep learning framework for printed circuit board manufacturing defect classification. *Sci. Rep.* **2022**, *12*, 12559. [CrossRef] [PubMed]
20. Zhang, Z.; Zhang, W.; Zhu, D.; Xu, Y.; Zhou, C. Printed circuit board solder joint quality inspection based on lightweight classification network. *IET Cyber-Syst. Robot.* **2023**, *5*, e12101. [CrossRef]

21. Deng, S.; Deng, L.; Meng, X.; Sun, T.; Chen, B.; Chen, Z.; Hu, H.; Xie, Y.; Yin, H.; Yu, S. EHIR: Energy-based hierarchical iterative image registration for accurate PCB defect detection. *Pattern Recognit. Lett.* **2024**, *185*, 38–44. [CrossRef]
22. Misra, I.; Rohil, M.K.; Subbiah, M.M.; Dhar, D. Feature based remote sensing image registration techniques: A comprehensive and comparative review. *Int. J. Remote Sens.* **2022**, *43*, 4477–4516. [CrossRef]
23. Liu, C.; Xu, J.; Wang, F. A review of keypoints' detection and feature description in image registration. *Sci. Program.* **2021**, *2021*, 8509164. [CrossRef]
24. Joglekar, J.; Gedam, S.S.; Mohan, B.K. Image matching using SIFT features and relaxation labeling technique—A constraint initializing method for dense stereo matching. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 5643–5652. [CrossRef]
25. Noble, F.K. Comparison of OpenCV's feature detectors and feature matchers. In Proceedings of the 2016 23rd International Conference on Mechatronics and Machine Vision in Practice (M2VIP), Nanjing, China, 28–30 November 2016; pp. 1–6. [CrossRef]
26. Wang, S.; Guo, Z.; Liu, Y. An image matching method based on sift feature extraction and FLANN search algorithm improvement. *J. Phys. Conf. Ser.* **2021**, *2037*, 012037. [CrossRef]
27. Bo, D.; Liang, Z.G.; Long, C.X. An algorithm of image matching based on mahalanobis distance and weighted KNN graph. In Proceedings of the 2015 2nd International Conference on Information Science and Control Engineering, Shanghai, China, 24–26 April 2015; pp. 116–121. [CrossRef]
28. Chum, O.; Matas, J. Optimal randomized RANSAC. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *30*, 1472–1482. [CrossRef] [PubMed]
29. Ye, Y.; Guangrui, F.; Shiqi, O. An algorithm for judging points inside or outside a polygon. In Proceedings of the 2013 Seventh International Conference on Image and Graphics, Qingdao, China, 26–28 July 2013; pp. 690–693. [CrossRef]
30. Li, W.; Zhang, Y.; Sun, Y.; Wang, W.; Li, M.; Zhang, W.; Lin, X. Approximate nearest neighbor search on high dimensional data—Experiments, analyses, and improvement. *IEEE Trans. Knowl. Data Eng.* **2019**, *32*, 1475–1488. [CrossRef]
31. Chatterjee, B.; Walulya, I.; Tsigas, P. Concurrent linearizable nearest neighbour search in LockFree-KD-tree. *Theor. Comput. Sci.* **2021**, *886*, 27–48. [CrossRef]
32. Sangappa, H.K.; Ramakrishnan, K.R. A probabilistic analysis of a common RANSAC heuristic. *Mach. Vis. Appl.* **2019**, *30*, 71–89. [CrossRef]
33. Cao, M.; Jia, W.; Lv, Z.; Li, Y.; Xie, W.; Zheng, L.; Liu, X. Fast and robust feature tracking for 3D reconstruction. *Opt. Laser Technol.* **2019**, *110*, 120–128. [CrossRef]
34. Dufournaud, Y.; Schmid, C.; Horaud, R. Image matching with scale adjustment. *Comput. Vis. Image Underst.* **2004**, *93*, 175–194. [CrossRef]
35. Hua, G.; Huang, W.; Liu, H. Accurate image registration method for PCB defects detection. *J. Eng.* **2018**, *2018*, 1662–1667. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.