J Sign Process Syst (2012) 67:279–290
DOI 10.1007/s11265-010-0556-8

A Flexible PCB Inspection System Based on Statistical Learning

Chia-Te Liao · Wen-Hao Lee · Shang-Hong Lai

Received: 3 March 2010 / Revised: 23 October 2010 / Accepted: 27 October 2010 / Published online: 18 November 2010
© Springer Science+Business Media, LLC 2010

Abstract With the large variations in appearance for different kinds of defects in Printed Circuit Boards (PCBs), conventional rule-based inspection algorithms become insufficient for detecting and classifying defects. In this study, an automated PCB inspection system based on statistical learning strategies is developed. First, the partial Hausdorff distance is used to ascertain the positions of defects. Next, the defect patterns are categorized using the Support Vector Machine (SVM) classifier. Defects without regularities in appearance, which cannot be categorized, are identified through the regional defectiveness by comparing the block-wise probability distributions. Experimental results on a real visual inspection platform show that the proposed system is very effective for inspecting a variety of PCB defects.

Keywords PCB · SVM · Automated visual inspection · Image classification · Defect classification · Image alignment

1 Introduction

With the increasing demand for flawless quality in the PCB assembly process, electronics manufacturers constantly request higher levels of precision in PCB inspection. In general, a good PCB inspection system has to satisfy the requirements of high inspection speed, high detection accuracy, and low false-alarm rate. Commonly used PCB defect detection procedures can be divided into two categories: (a) electrical/contact methods and (b) non-electrical/non-contact methods. The electrical methods can detect some defects, but the inspection capabilities are limited. In addition, electrical methods may cause mechanical damages during the probe contact. For the non-electrical/non-contact PCB inspection methods, numerous techniques have been proposed for *Automated Optical Inspection* (AOI). They can be categorized into three main approaches: reference comparison, non-reference verification, and hybrid approach [1]. The referential approach compares the tested PCB image with a reference PCB image. The main limitations of this approach are the need for very precise alignment and the vulnerability to differences in illumination. The second type of approach (non-reference verification) tests the design rules of the PCB features to determine whether each feature falls within the required range. However, it is not easy for users to design the rules as constraints on image features in the rule-based verification, and it lacks flexibility to distorted features. The hybrid approach concurrently combines the advantages of the aforementioned approaches. As hardware gets cheaper and faster, it is becoming more popular to use hybrid methods that benefit from both approaches.

Numerous PCB inspection algorithms have been proposed and can be found in the literature [2–4]. The earliest developed PCB inspection algorithm is known as template-matching technique as proposed by Akiyarnai et al. [5]. In the following decade, various PCB inspection methods were developed. For example, Tominaga [6] used a dichromatic reflection model to describe object surfaces followed by a hyper-sphere method for defective object identification. Spence [7] converted the currents moving between the devices on PCBs to a magnetic field image for locating the possible faults, and interpreted the magnetic field patterns with artificial neural networks for fault identification. Later, Hodges et al. [8] diagnosed solder

C.-T. Liao · W.-H. Lee · S.-H. Lai (✉)
Department of Computer Science, National Tsing Hua University, Hsinchu 300, Taiwan
e-mail: lai@cs.nthu.edu.tw

pad arrays on PCBs based on template matching, where histogram features were used. In [9], Scaman et al. utilized wide-angled blue lights to differentiate the metal patterns from polyimide background for PCB inspection. Wu et al. [10] proposed the use of an elimination–subtraction procedure to locate and analyze PCB defects by subtracting the template image from inspection image. Loh et al. [11] inspected the surface mount technology (SMT) components on PCBs by using structured-lighting. There, a slant map was extracted as the shape information of the solder joint surface, followed by a rule-based defect classification procedure. In addition, Kim et al. [12] built a PCB inspection system based on a technique that matches blocks between reference and inspected images. Tominaga et al. [13] also proposed a PCB inspection system that first sampled the surface reflectance with a liquid-crystal tunable filter imaging system, and a pixel-wise classification was next performed using *k*-means algorithm. In the system developed by Ibrahim et al. [14], the image difference operation was applied in the HAAR-wavelet domain. An XOR logic operation was next performed to generate the final result for inspection. Garcia et al. [15] explored the multivariate stepwise discriminant methods (MSDM) to select the best features and applied rules for inspection. Acciani et al. [16] exploited neural-networks for diagnosing the solder joint defects on PCBs. Chang et al. [17] adopted the hybrid approach to construct a two-phase PCB inspection system. In the first phase, a set of defect images of several existing basic patterns were stored to form a concept space, where several empirical rules were used for defect classification. In the second phase, a new pattern was evolutionally grabbed by calculating the relative position of several similar cases in the concept space.

Although inspecting bare PCBs is already a successful industrial application, its full potential has not yet been exploited. One reason for the absence of AOI systems with continuous improvements is the lack of flexibility to adapt its algorithms for inspecting new defects. The problem of reconfiguration and reprogramming, combined with the rapid introduction and retirement of electronic products, has deterred equipment manufacturers from investing in the development of AOI systems. Besides, as can be seen, most existing inspection systems are rule-based and designed from human observations. From a statistical point of view, they are not robust enough and unable to generalize well for the unseen defects. Another problem is the requirement to identify both functional and cosmetic defects from their visual appearance [1]. Therefore, for PCB inspection, it is essential to develop methodologies that allow rapid adaptation and generalize well for different variations of defects like human beings do. This research introduces the development of a novel AOI system for PCBs, which has several advantages and contributions. First, instead of using the rules to classify the detected defects heuristically, a learning-based solution is developed through solving an optimization problem where the generalization ability is theoretically maximized. Second, the proposed system is flexible, allowing for potential function extension. By adding defective samples into the database, the inspection system has the capability to classify new defects. It significantly cuts down cost for the eventual upgrading of the inspection system. Third, for cosmetic defects with irregular appearance, such as weave exposure, blowhole, scratches, solder on trace, and problems from solder mask, such as air inclusion, a probabilistic solution is developed to identify the defective regions and measure the defectiveness quantitatively. The key requirements of the reliability and flexibility are therefore achieved.

This paper is organized as follows: In Section 2, some theories related to this research are briefly reviewed. Section 3 describes the components of the proposed inspection system with application to PCBs. Section 4 presents some experimental settings and results. Finally, Section 5 concludes this paper.

2 Related Theories

2.1 Hausdorff Distance

Hausdorff distance [18] is useful in determining the degree of similarity between the edge maps of two templates. It provides a means of computing the distance between two point sets. Let A and B be two compact subsets of points in a metric space M, the definition of Hausdorff distance between A and B is given by

$$d_H(A,B) := \max\{h(A,B), h(B,A)\}, \tag{1}$$

$$h(A,B) := \max_{a\in A} \min_{b\in B} d(a,b) \tag{2}$$

where $d(a,b)$ denotes the distance between two points a and b in M, usually the Euclidean distance. In practice, to match a template image and a target image, it is necessary to first apply an edge detector to derive the corresponding point sets of the images. In Section 3.1, Hausdorff distance computation is exploited as the technique to align an inspection image I with a reference image R by finding the minimal distance value with respect to different geometric transformations.

2.2 Support Vector Machine

Support Vector Machine (SVM) [19] is a learning system for efficiently training linear learning machines from

optimization theory, while the formulation leads to better generalization with the maximum margin criterion [20]. Let $S = \{\mathbf{x}_l, y_l\}_{l=1}^m$ be a set of training examples, where $\mathbf{x}_l \in \mathbb{R}^n$ is the lth example associated with the class label $y_l \in \{1, -1\}$. The aim is to define a separating hyperplane for which the margin between the positive and negative examples is maximized. This amounts to finding the normal $\mathbf{w}$ and the bias b so the distance between the closest points to the hyperplane, i.e. $1/\|\mathbf{w}\|$, is maximized:

$$\begin{aligned}&\min_{\mathbf{w},\xi} \tfrac{1}{2}\|\mathbf{w}\|^2 + C\textstyle\sum_{l=1}^m \xi_l \\ &s.t.\, y_l(\mathbf{w}^T\mathbf{x}_l + b) \geq 1 - \xi_l \, . \\ &\xi_l \geq 0,\ l = 1, 2, ..., m\end{aligned} \tag{3}$$

Here, ξ_l is the positive slack variable introduced to handle the non-separable cases [19], and C is chosen to control the tradeoff between errors and functional complexity. Since $\|\mathbf{w}\|^2$ is convex, minimization can be achieved by differentiating the Lagrangian of (3), and it leads to an equivalent quadratic programming (QP) problem:

$$\begin{aligned}&\min_{\alpha} \alpha^T Q\alpha - \mathbf{e}^T\alpha \\ &s.t.\, 0 \leq \alpha_l \leq C,\ l = 1, 2, ..., m\end{aligned} \tag{4}$$

where $Q_{i,j} = y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$ is an m x m matrix, and K is a symmetric kernel function [19, 21]. Some frequently used kernels include polynomial kernel $K(\mathbf{x}_i,\mathbf{x}_j) := (\mathbf{x}_i^T\mathbf{x}_j+1)^d$ and RBF kernel $K(\mathbf{x}_i,\mathbf{x}_j) := \exp(-\|\mathbf{x}_i,\mathbf{x}_j\|^2/2\sigma^2)$. Assume the optimal solution $\boldsymbol{\alpha}^* = (\alpha_1, \alpha_2, \ldots, \alpha_m)$ is found by solving Eq. 4 with a standard QP solver [22]. For a new test sample $\mathbf{x}_{test}$, it is classified by $f(\mathbf{x}_{test}; \alpha^*, b^*) = \text{sgn}\left(\sum_{l=1}^m \alpha_l y_l K(\mathbf{x}_l, \mathbf{x}_{test}) + b^*\right)$. In Section 3.2, the located PCB defects are expressed by the extracted features and fed into the SVM learning system for training and testing procedures.

2.3 Multi-Class SVM Classification and Probability Estimates

Since SVM is designed as a binary classifier, different approaches have been proposed to use the two-class decisions for multi-class classification [23, 24]. In many situations, however, probability estimates are desired to quantify the results rather than only predicting the class labels. The scenario of SVM multi-class probability estimation can be summarized as follows: Let $S = \{\mathbf{x}_l, y_l\}_{l=1}^m$ be the set of training examples and $y_l \in \{1, 2, ..., k\}$ be the class label associated to $\mathbf{x}_l$. One first needs to construct $k(k$-1)/2 pair-wised SVM classifiers where each one is trained on data from two classes, say, the ith and j^{th} classes. For any new test sample $\mathbf{x}_{test}$, the pair-wise probabilistic outputs are computed from the SVM for the ith and j^{th} classes, called the pair-wise class probabilities $r_{ij} = P(y_{test} = i|\mathbf{x}_{test}, y_{test} \in \{i, j\})$, and r_{ji}=1-r_{ij}. Next, the multi-class probability distribution $\mathbf{p}=(p_1,\ldots,p_k)^T$ is computed from all pair-wise probabilities r_{ij}'s. In this research, Platt's method [23] is adopted to calculate r_{ij} between all class-pairs for $\mathbf{x}_{test}$. As a result, $\mathbf{x}_{test}$'s multi-class probability distribution $\mathbf{p}$ is estimated by solving the following optimization problem [25]:

$$\begin{aligned}&\min_{\mathbf{p}} \textstyle\sum_{i=1}^k \sum_{j:j\neq i} \left(r_{ji}p_i - r_{ij}p_j\right)^2 \\ &s.t.\, \textstyle\sum_{i=1}^k p_i = 1, p_i \geq 0, \forall i\end{aligned} \, . \tag{5}$$

where $p_i := P(y_{test}=i|\mathbf{x}_{test})$, i=1,...,k is the posterior class probability. Equation 5 can be rewritten as:

$$\begin{aligned}&\min_{\mathbf{p}} 2\mathbf{p}^T Q'\mathbf{p} \\ &s.t.\, p_1 + p_2 + \ldots p_k = 1\, ,\end{aligned} \tag{6}$$

where $Q'_{ij} = \sum_{s:s\neq i} r_{si}^2$ if i=j, and $Q'_{ij} = -r_{ji}r_{ij}$ otherwise. Since Q' is a positive semi-definite matrix, $\mathbf{p}$ is the global optimum if it satisfies the first order necessary optimality condition:

$$\begin{bmatrix} Q' & \mathbf{e} \\ \mathbf{e}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}, \tag{7}$$

where λ is the Lagrangian multiplier, $\mathbf{e}$ and $\mathbf{0}$ are k by 1 vectors for all ones and all zeros respectively. In Section 3.3, the cosmetic defectiveness is evaluated through block-wise comparing the multi-class probability distributions of image blocks in the inspection-reference image pairs.

3 The Learning-Based Inspection System

As the flow diagram in Fig. 1 shows, the PCB inspection system consists of an image acquisition system, and a processing system to perform the inspection on the acquired image. The image acquisition system has a mechanical positioning tool to scan and acquire the PCB image. The inspection process itself has two steps: image alignment and image diagnosis. First, the image acquired by the camera is aligned to the stored reference image as described in Section 3.1. Next, the PCB diagnosis is performed by detecting two kinds of defects in two different ways, namely: 1) the defects that appear as "patterns," and 2) the cosmetic defects without regular appearance, which are usually caused by a failed electroplating process or a failed rising procedure between fabrication stages. The diagnosis processes of feature extraction and defect identification are respectively described in Section 3.2 and Section 3.3 for two kinds of defects. The diagnosis results of these two sequences provide the defect inspection of the PCBs. Each component of these steps is detailed in the following subsections.

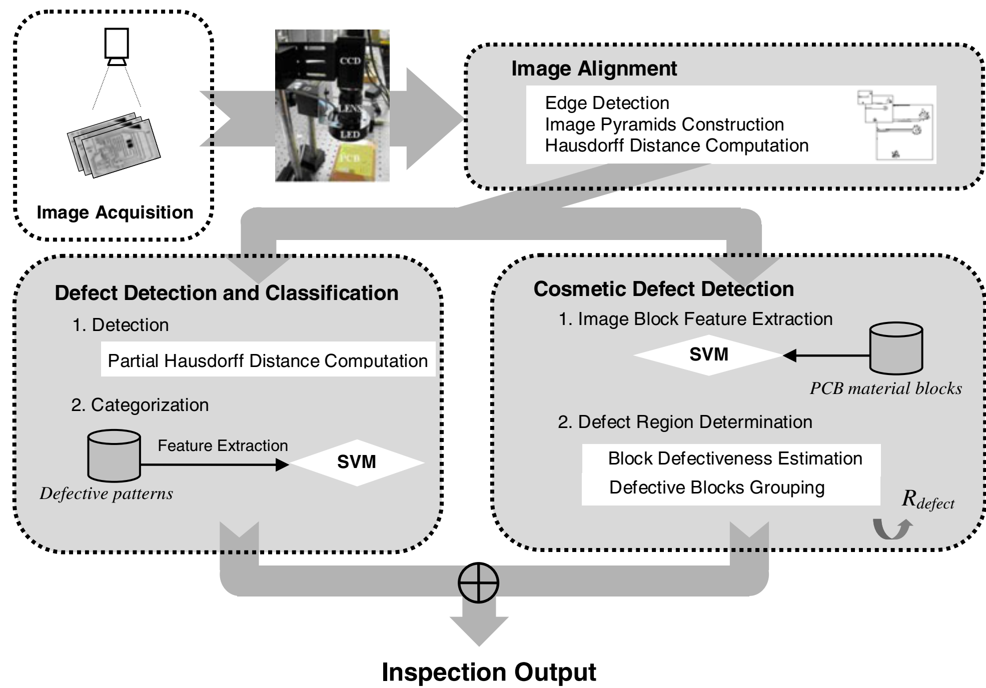

Figure 1 Flowchart of the proposed PCB inspection system.

3.1 Image Alignment

The inspection process starts with the Hausdorff-distance based image alignment procedure that aligns the inspection image I to the reference image R. The notation R stands for an idealized image on which the alignment and the subsequent inspection procedures rely, and it can be obtained by a compilation or a rolling average of several defect-free boards in practice. In the experiments, R is constructed from the compilation of several good PCBs. The image alignment procedure is as follows: First, a *Sobel* edge detector is applied to I and R to extract the corresponding point sets I_{edge} and R_{edge} respectively. Next, by finding the minimal Hausdorff-distance value between I_{edge} and R_{edge} with respect to different geometric transformations, i.e. $\arg\min_{\theta} d_H(I_{edge}, a_{\theta}(R_{edge}))$, where $a_{\theta}(\bullet)$ is the rigid transformation with parameter θ, the image alignment process is accomplished.

To enhance the robustness of alignment, the one-way Hausdorff distance from R_{edge} to I_{edge} was actually used for reducing the influence of outlier points in I_{edge}. In addition, the original max–min function in Eq. 2 was modified from calculating the maximum of minima to the mean of minima in the implementation because the result of applying the max function is sensitive to a few points located in the previously identified problem areas. The modified h function is defined as:

$$h_{mean}(A,B) := \frac{\sum_{a\in A} \min_{b\in B} ||a-b||}{|A|}, \tag{8}$$

where $|A|$ denotes the cardinality of point set A. Furthermore, the techniques of image pyramid and distance map images were used to speedup the alignment process. On the former technique, by constructing a layered image pyramid,

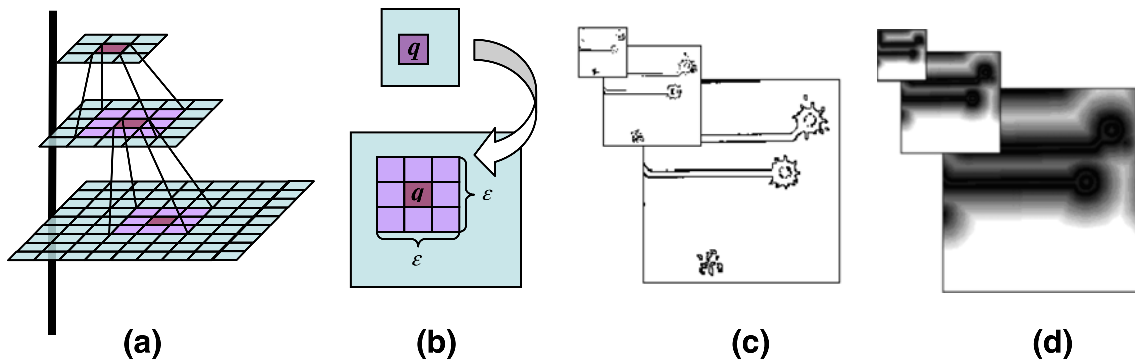

Figure 2 Speeding up of alignment process using an image pyramid. **a** An example of three-layer pyramid for image alignment, **b** the local search area for refinement in the next layer, **c** an example of three-layer edge map pyramid computed from an inspection image I, and **d** the three-layer pyramid of distance map R_{dist} constructed from the corresponding reference image R.

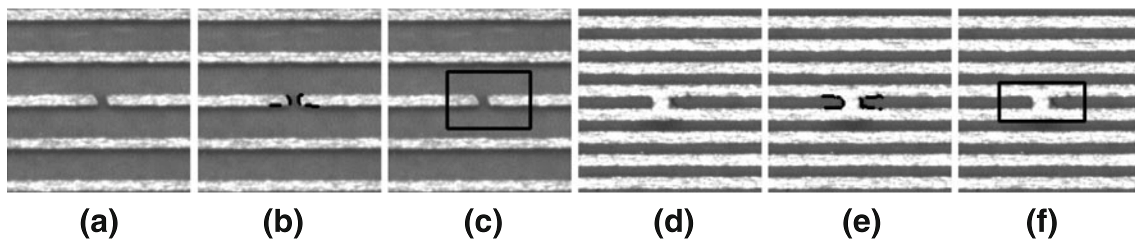

Figure 3 Two examples of defect detection: **a** an example of open defect, **b** edge points are marked as defective, **c** the image patch of defect is cropped for the use of subsequent classification, **d**–**f** another example of short defect.

full-search is applied only at the coarsest layer, and image alignment is successively refined through local searches on the candidates from the coarser layer. The searching procedure is stopped at θ^* of the lowest value in the finest layer. And on the latter technique, a distance map R_{dist} is used. The numerator of Eq. 8 can be rapidly retrieved in the execution stage by accessing R_{dist} from storage. Figure 2 illustrates the use of the coarse-to-fine approach and the distance map for speeding up the image alignment process.

3.2 Defect Detection and Classification

Given a well-aligned image pair of inspection and reference images, the following step is to handle defective patterns with regular appearance. The defects are first located using the partial Hausdorff distance based strategy, and subsequently categorized by SVM learning approach.

3.2.1 Defect Detection via Partial Hausdorff Distance

Suppose $e_p \in I_{edge}$ is an edge point at position p on the inspection image I. It is easy to see, when referring e_p to R_{dist} at position p, a large value will be derived if the reference image does not contain such an edge point at p. This observation points at the possibility of locating defects by accessing the stored Hausdorff-distance map R_{dist}. However, if outliers exist in the compared objects, the mismatch will become too large. For this reason, use of the partial Hausdorff distance is more reasonable than the original Hausdorff distance computation. The partial Hausdorff distance takes the Kth ranked highest value instead of the overall maximum:

$$d_H^p(A,B) = \min\left(h_p(A,B), h_p(B,A)\right),$$

where $h_p(A,B) := K^{th}_{a\in A}\min_{b\in B}\|a-b\| = K^{th}_{a\in A} d_B(a)$.

Here $K^{th}_{a\in A}$ denotes the Kth ranked value in the set of $d_B(a)$ distances, which are the minimum distance values from each location a to the point set B. It is empirically set as the third quartile in the experiments.

In summary, given an inspected image I, the defect location procedure is as follows: Initially, by referring every edge point e_p to R_{dist} at position p, the distance map I_{dist} is generated. Next, the edge pixels are marked as defective if the associated partial Hausdorff distance values on I_{dist} are larger than a threshold. Finally, any defective pixels within a certain distance are labeled as one defect. Figure 3 demonstrates some example images for illustration. In the next section, the image patches of defects are fed into SVM for further evaluation.

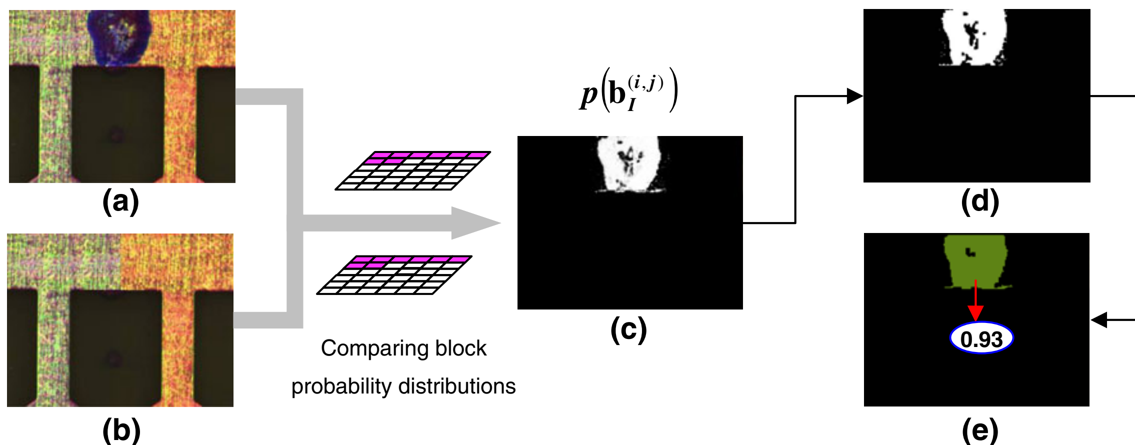

Figure 4 The process of defect detection and defectiveness computation. **a** The inspected image. **b** The normal reference image. **c** The block defect probability displayed by intensity. **d** The defect region after applying closing operator. **e** The labeled defect region with defectiveness R_{defect}.

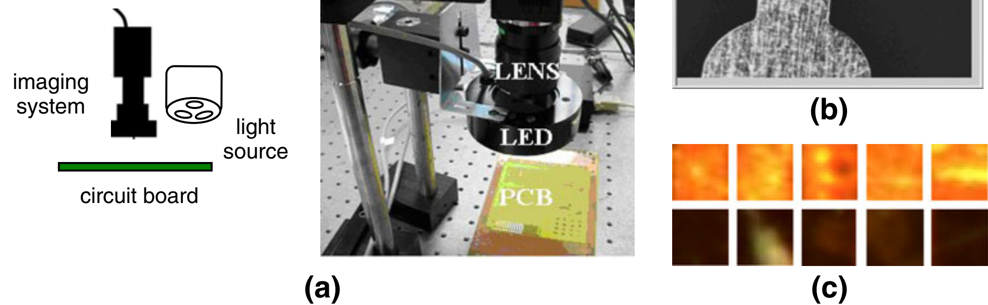

Figure 5 Image acquisition components and the examples for SVM training and testing. **a** Image acquisition set-up for PCB defect inspection, **b** a snapshot of image acquisition system output, and **c** examples of the observed metal and substrate images.

3.2.2 Defect Classification via Support Vector Machine

For an inspection system, the ability to analyze the statistics of defects in order to reduce the defective rate during manufacturing is extremely important. However, one of the most crucial factors for the performance of an inspection system is the quality of the inspection features. Here, two kinds of descriptors are designed to extract the discriminant features to better characterize the defective patterns for classification. They are:

a. Horizontal / Vertical cross-count: The cross-counts with 16 equally-spaced scan-lines in horizontal and vertical directions are measured.
b. Projection variations: The variation in projection with respect to 16 equally-spaced scan-lines is measured for both horizontal and vertical directions.

Having a set of collected samples of defective image patches, the descriptors were applied to convert image patches from pixels into the feature vectors for inspection. Specifically, let $S = \{\mathbf{x}_l, y_l\}_{l=1}^{m}$ be the set of collected defective patterns, where $\mathbf{x}_l$ is the feature vector converted from the lth defective pattern, and $y_l \in \{1, 2, ..., k\}$ is the defect type associated with $\mathbf{x}_l$. An SVM optimization problem is then formulated as introduced in Section 2.2 in order to train the classifier for the types of defective regions. In the testing phase, the types of defective regions are recognized using the trained SVM classifier for categorization.

3.3 Cosmetic Defect Detection

On the one hand, it is viable to identify some defects because of the regular patterns in defect appearance. On the other hand, because of the diverse appearance and the difficulty to characterize them, identifying cosmetic defects, which are caused by failures in the electroplating or rising processes between fabrication stages, is not easy. In the following section, through block-wise comparison of the multi-class probability distributions of image blocks, such non-regular defects are thus identified. Furthermore, a quantitative measure for the regional defectiveness is provided for the defective regions. The procedure for detecting the cosmetic defects consists of: 1) block feature extraction, and 2) block-wise and regional defect probability computation; which are elaborated below.

3.3.1 Block Feature Extraction

Identification of cosmetic defects is performed by block-wise comparing the SVM multi-class probability of PCB material blocks between the inspection and reference images. Suppose a set of image blocks of various PCB materials is collected in advance. To train the SVM model for estimating the multi-class probability for every image block, an effective feature extraction procedure is required. To better discriminate color textures, for each image block the feature is extracted by: First, the mean values with respect to

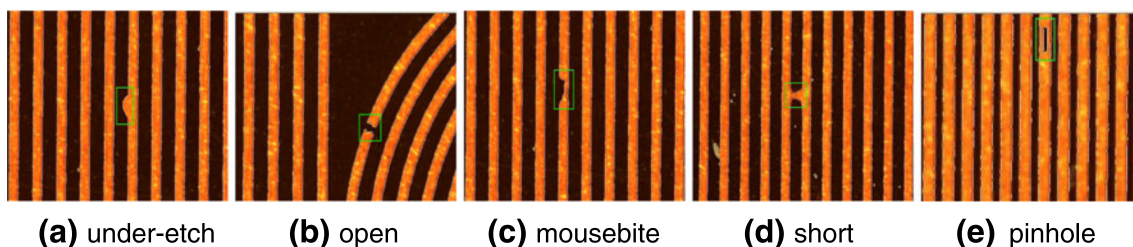

Figure 6 Inspection results with different defects marked on PCBs.

Table 1 Comparison of SLP and SVMs with different kernels by 5-fold cross validation.

	linear	poly., d=2	poly., d=3	poly., d=4	poly., d=5	RBF, σ^2=500	SLP
ave. accuracy	91.5%	87%	89%	80%	86.3%	93.5%	85%
ave. # of SVs	100.6	159.2	100.6	149	94	234.2	–

each color channel (μ^R, μ^G, μ^B) and the color of the central pixel (c^R, c^G, c^B) are calculated and stored. Second, the variance to each color channel (i.e. v^R, v^G, v^B), is considered as well, to keep the block's color variation information:

$$v^{color} = \sum_{i=1}^{h} \sum_{j=1}^{w} \left| P_{ij}^{color} - \mu^{color} \right|, \tag{9}$$

where P_{ij}^{color} denotes the intensity with respect to each *color* channel of $\{R,G,B\}$, and (i, j) stands for the pixel position in the h x w block. Third, three rotationally-invariant features (l^R, l^G, l^B) are computed by applying the *LoG* (Laplacian of Gaussian) filter to the image block. In summary, for each image block of PCB material, a 12-dimentional feature vector $(\mu^R, \mu^G, \mu^B, c^R, c^G, c^B, v^R, v^G, v^B, l^R, l^G, l^B)$ is extracted. Let $S = \{\mathbf{x}_l, y_l\}_{l=1}^m$ be the set of examples, where $\mathbf{x}_l \in \mathbb{R}^{12}$ is the corresponding feature vector associated with $y_l \in \{1, 2, ..., k\}$, the type of PCB material. As introduced in Section 2.3, the SVM model θ_{SVM} for multi-class probability estimation is trained with S, and it will be used to assign the probability of the material type to image blocks in the next section.

3.3.2 Defect Region Determination

Let $\mathbf{b}_I^{(i,j)}$ be the $(i,j)^{\text{th}}$ image block of the inspection image I. After learning from S, the SVM model θ_{SVM} is able to associate $\mathbf{b}_I^{(i,j)}$ with a probability $P(\mathbf{b}_I^{(i,j)}|\theta_{SVM}) = (p_1^{(i,j)}, p_2^{(i,j)}, ..., p_k^{(i,j)})^T$, which represents the probability of the material type for $\mathbf{b}_I^{(i,j)}$. The defectiveness of $\mathbf{b}_I^{(i,j)}$ is next measured by computing the distance between $P(\mathbf{b}_I^{(i,j)})$ and $P(\mathbf{b}_R^{(i,j)})$, where $P(\mathbf{b}_R^{(i,j)})$ denotes the probability distribution computed from the corresponding block $\mathbf{b}_R^{(i,j)}$ in the reference image. Clearly, the farther the distributions $P(\mathbf{b}_I^{(i,j)})$ and $P(\mathbf{b}_R^{(i,j)})$ are, the more likely that block $\mathbf{b}_I^{(i,j)}$ is defective. The *relative entropy* [26], also known as *Kullback-Leibler* (KL) *distance*, is used here to measure the distance between the two probability distributions $P(\mathbf{b}_I^{(i,j)})$ and $P(\mathbf{b}_R^{(i,j)})$:

$$KL\left(P\left(\mathbf{b}_I^{(i,j)}\right), P\left(\mathbf{b}_R^{(i,j)}\right)\right) = \sum_{l=1}^{k} P\left(\mathbf{b}_I^{(i,j)}\right)_l \log\left(\frac{P\left(\mathbf{b}_I^{(i,j)}\right)_l}{P\left(\mathbf{b}_R^{(i,j)}\right)_l}\right), \tag{10}$$

where $P\left(\mathbf{b}_I^{(i,j)}\right)_l$ and $P\left(\mathbf{b}_R^{(i,j)}\right)_l$ denote the lth entry of the corresponding probability distributions. Finally, a logistic sigmoid function is fit to transform $KL(P(\mathbf{b}_I^{(i,j)}), P(\mathbf{b}_R^{(i,j)}))$ to the probability of the block defectiveness:

$$p(\mathbf{b}_I^{(i,j)}) = 1 - \exp(-\beta \times KL(P(\mathbf{b}_I^{(i,j)}), P(\mathbf{b}_R^{(i,j)}))), \tag{11}$$

where β is an empirically-chosen parameter. For better representation, the neighbouring defective blocks are grouped as a single defective region using the standard morphological operators and the connected component approach [27]. The regional defectiveness of this cosmetic defect is defined as:

$$R_{defect}(D) := \exp\left(\frac{1}{|D|} \sum_{\mathbf{b}_I^{(i,j)} \in D} \ln p\left(\mathbf{b}_I^{(i,j)}\right)\right), \tag{12}$$

where $|D|$ denotes the total number of the defective blocks inside D. To recap, as explained above, through block-wise comparing the SVM multi-class distributions between the inspection and reference image blocks, the defective blocks are identified. These defective blocks are then grouped into a defective region with the defectiveness $R_{defect}(D)$. For better illustration, the intermediate results of each stage are visualized in Fig. 4. The performance of the proposed inspection system is evaluated in the next section.

4 Experimental Results

4.1 Imaging System

The image acquisition unit of the inspection system was composed of a monochrome area CCD, lens, and a 3-color LED light source. The lighting mode was switched rapidly among R, G, B by the LED light source, and inspection images of the corresponding color channels were obtained by the simultaneously triggered CCD. Before taking the

Table 2 The confusion matrix of the defect classification results.

	Under-etch	Mousebite	Short	Open	Pinhole
Underetch	72	1	7	0	0
Mousebite	0	72	0	8	0
Short	6	0	74	0	0
Open	0	10	0	69	1

Table 3 Average system execution time.

Phase	Edge Detection	Defect Detection	Feature Extraction	Defect Recognition
Time	65.56 ms	58.45 ms	<1 ms	<1 ms

PCB images, the CCD was calibrated with the flat field correction technique to ensure the uniformity of the obtained images. The imaging acquisition component is shown in Fig. 5a, together with a snapshot of the system output in Fig. 5b. Some examples of the observed metal as well as the substrate are shown in Fig. 5c. The images acquired from this imaging system were subsequently processed by the proposed inspection algorithms to test their performance.

4.2 Pattern Defects

As previously reported in the literature, most common defects on PCBs are caused by errors of thermal expansion of the artwork during printing, dirt on board, air bubbles from electrolysis, incorrect electrolysis timing, or mechanical mis-registrations, which appear on the finished PCB products and cause defects as: open, short, pinhole, under-etch, or mousebite [1, 28]. The experiments of this work focused on these five common defects. In the experiments, the defective examples in the database were from the failed PCBs during the manufacturing process and acquired by the imaging platform. Eighty images for each error type were obtained from real defect images or by simulating defect images from non-defective images, thus yielding a total of four hundred PCB images in the database. Some examples of defect detection results are given in Fig. 6, where the detected defects are marked by rectangles. In the experiments, 100% detection rate was achieved using the proposed partial Hausdorff-distance based method. The detected patches were subsequently fed into SVM to further classify their defect types.

The system performance for defect classification was evaluated using five-fold cross validation. That is, the detected image patches were randomly divided into five portions. Each portion was then applied once for assessing the performance of the classifier by using the rest four portions as the training data. In Table 1, the average classification accuracy and total numbers of support vectors for SVMs in conjunction with various kernels (i.e. polynomial and RBF kernels) are compared. The classification accuracy of a single layer perceptron (SLP), which consists of ten hidden nodes associated with weights trained by the LMS method [29], is also provided for benchmarking. From this table, we see the SVMs incorporated with the polynomial kernels (with different degrees d) performed properly in terms of the classification accuracy. On the other hand, the RBF kernel SVM with an appropriate kernel parameter setting achieved the highest classification accuracy, but it requires the highest computational complexity and the largest number of support vectors (SVs) compared to other kernels in the comparison. Applying more complex kernels (i.e. higher order polynomial kernels) here to map the data to higher dimensions may lead to smaller training errors; however, it may cause the system to overfit the training data and thus degrades the accuracy of the trained system during testing. The overfitting phenomenon occurs probably because the available PCB defect samples are limited. As a result, the linear kernel was adopted for its computational efficiency and satisfactory performance with this limited experimental dataset. However, it may need to use a higher-order polynomial or a more sophisticated kernel for more complicated PCB classification tasks.

The classification result was further evaluated to study the confusions among various defect types. In Table 2, a confusion matrix of the accumulated defect classification results over five rounds is given, where each row represents the instances in an actual class and each column represents the instances in a predicted class. From the table, there are more misclassification cases between “open” and “mousebite” defect types, which are quite similar in appearance. The average system execution time is reported in Table 3 for each phase of the proposed inspection system. The computer platform is equipped with 1.86 GHz CPU and 1G RAM onboard. Compared to the state-of-the-art approach of Chang et al. [17], the proposed system inspects defects

Table 4 Comparison of the proposed method and CBR evolutionary model [17].

	Image size & field of view	per pixel resolution	Average detection time	Defect detection	Image alignment	Defect classification
The proposed method	480×640 3 mm×4 mm	6 μm	124 ms on 1.86 G CPU	Partial Hausdorff distance	Fast Hausdorff distance computation	Statistical optimal
CBR evolutionary model [17]	480×480 8 mm×8 mm	16.7 μm	530 ms on 2.4 G CPU	XOR logic operator	N/A	Rule-based

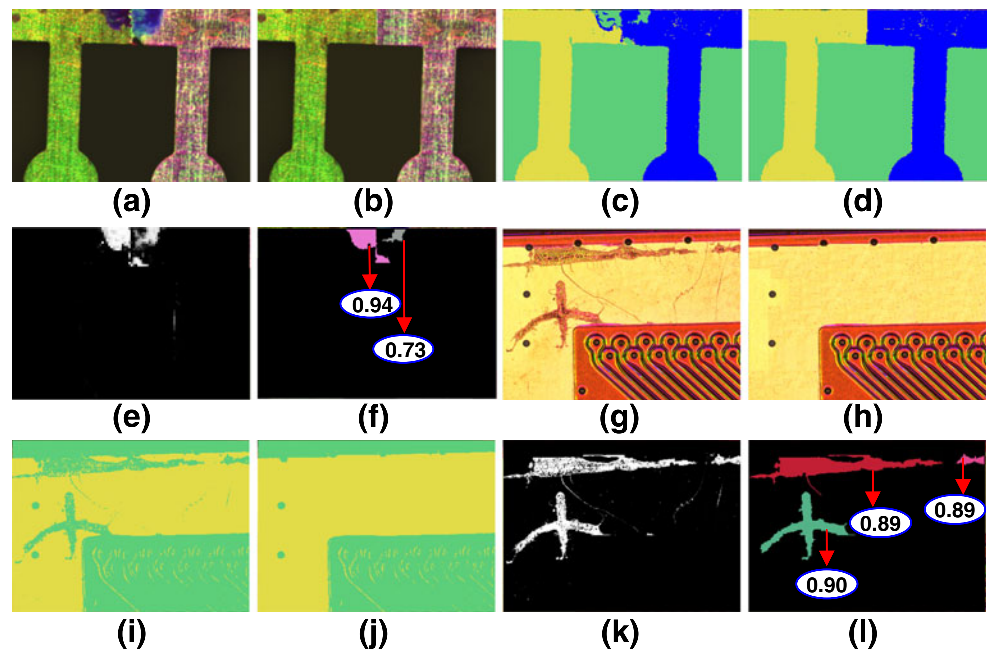

Figure 7 Experimental results for material defect inspection: **a**–**f** Ni-Cu image with foreign oxidized material, and **g**–**l** PCB image with copper exposure.

faster, and it is more reliable in defect detection and more accurate for defect classification. Besides, the proposed learning-based inspection system is easy to adapt for new defects, while the rule-based system [17] needs to design additional rules from the new defect observations and requires checking the dependency between rules. Details of the comparison were summarized in Table 4. Note that, in real situations, the defect patterns may not belong to any of the categories covered by the database. This will lead to mis-classification. A verification procedure (e.g. non-reference verification) can be employed to give warning of possible mis-classifications. By collecting new samples and training the corresponding sub-classifiers, the proposed learning-based inspection system can be easily extended to recognize new types of defects.

4.3 Cosmetic Defects

In the following experiments, the defective PCBs with cosmetic defects of skip plating, copper exposure, and those

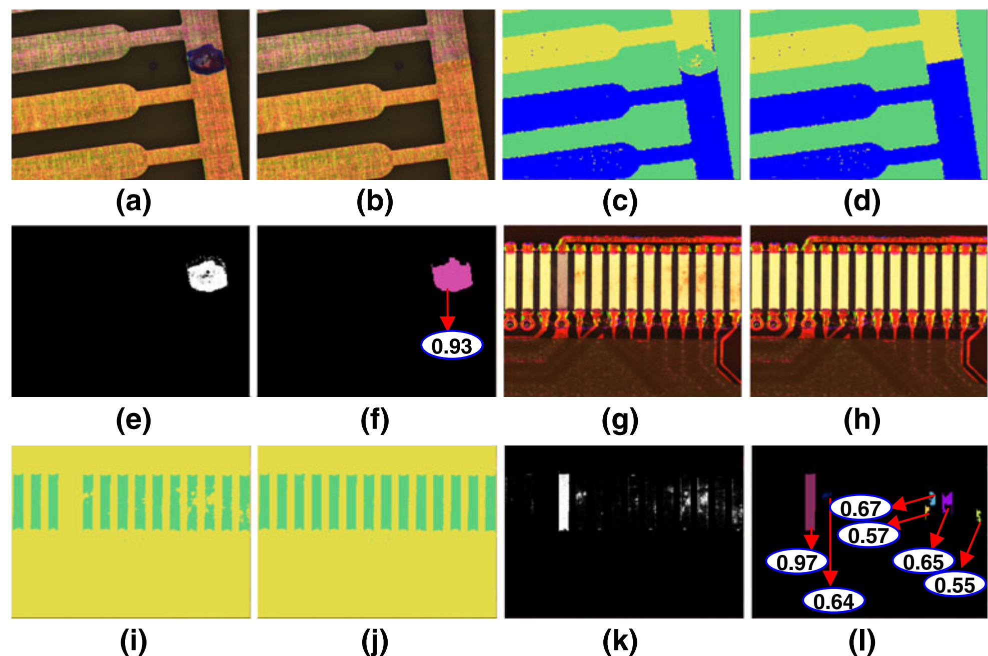

Figure 8 Experimental results for material defect inspection: **a**–**f** Au-Ni images with foreign oxidized material, and **g**–**l** pads of skip plating and copper exposure on the PCB.

with abnormal oxide on board were inspected by block-wise comparing their probability distributions. The multi-class probability distributions were estimated by the SVM with linear kernel. Figures 7a and 7g are two problematic PCB images being inspected, while 7b and 7h are the corresponding defect-free reference images. In Fig. 7a, an irregular region containing foreign oxidized material is seen on a Ni-Cu image, and in Fig. 7g a defective region of copper exposure is shown. Figures 7c, 7d and 7i, 7j are the SVM material classification results for the corresponding inspection and reference images. As for Figs. 7e and 7k, they are the intensity representation of the block defect probabilities for the inspection images in Figs. 7a and 7g, respectively. The final inspection results were shown in Figs. 7f and 7l depicting the detected regions with the values for defectiveness. Higher defectiveness value is shown if a region consists of blocks more dissimilar from the normal board, so the region is more certainly identified as a defect.

Figures 8a and 8g are another two problematic PCB images being inspected, while 8b and 8h are the corresponding defect-free reference images. The layout of the figures is similar to Fig. 7. As for the first problematic case shown in Fig. 8a, there is an irregular region with foreign oxidized material on an Au-Ni image. The proposed system successfully detected the defective region with a high confidence value because the defective region has a very different appearance from that of the corresponding reference image. On the other case shown in Fig. 8g, the PCB image concurrently has defects of skip plating and copper exposure. In Fig. 8k, the pad region of skip plating displayed with bright intensity was successfully detected with high probability. It is evident that the system also detects the regions with slight copper exposure. Yet, because copper is quite similar to gold in natural appearance, those blocks were not as firmly recognized to be defective and were associated with the lower regional defectiveness value in Fig. 8l. Thus, the reliability and the effectiveness of the proposed inspection system for identifying the cosmetic defects were validated through this experiment.

5 Conclusions

This paper presents an innovative and automated learning-based PCB inspection system. It compares the inspection image to a reference image, corrects the misalignment errors, and has the ability to accurately assess defects before they propagate further down the assembly line, while providing very rapid data analysis. In the experiments, the effectiveness of the system was demonstrated for defective patterns (opens, shorts, pinholes, mousebite, under-etch), and for cosmetic defects of irregular appearance (copper exposure, skip plating and oxides on board). The experiments indicated that the learning-based AOI system provides a reliable and flexible solution for PCB inspection, and it can be easily extended to deal with unknown defects for future PCB inspection tasks.

Several future directions for extending the proposed AOI system are discussed in the following. A possible improvement is to include the feature selection step into the system for enhancing both the generalization capability and the computation speed of the learned model. Since the proposed algorithms are flexible, handling different diagnostic tasks with the proposed algorithms according to the positions on the PCB assembly line is also feasible, such as SMT component placement inspection. Finally, incorporating different imaging technologies by using visible or invisible light, e.g. X-ray, into the system may also help the inspection for multi-layer PCBs.

Acknowledgements This work was partially supported by the Center for Measurement Standards, Industrial Technology Research Institute, Hsinchu, Taiwan.

References

1. Moganti, M., Ercal, F., Dagli, C. H., & Tsunekawa, S. (1996). Automatic PCB inspection algorithms: a survey. *Computer Vision and Image Understanding, 63*(2), 287–313.
2. Chin, R. T., & Harlow, C. A. (1982). Automated visual inspection: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 4*(6), 557–573.
3. Chin, R. T. (1988). Automated visual inspection: 1981 to 1987. *Computer Vision, Graphics, and Image Processing, 41*(3), 346–381.
4. Newman, T. S., & Jain, A. K. (1995). A survey of automated visual inspection. *Computer Vision and Image Understanding, 61*(2), 231–262.
5. Akiyarnai, Y., Hara, N., & Karasaki, K. (1983). Automation inspection for printed circuit board. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 5*, 623–630.
6. Tominaga, S. (1991). Surface identification using the dichromatic reflection model. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 13*(7), 658–670.
7. Spence, H. F. (1993). Printed circuit board diagnosis using artificial neural networks and circuit magnetic fields. In *Proc. IEEE Systems Readiness Technology Conference*, pp. 41–45.
8. Hodges, S. E., & Richards, R. J. (1995). Fast multi-resolution image processing for PCB manufacture. *IEE Colloquium on Multi-resolution Modeling and Analysis in Image Processing and Computer Vision*, pp. 1–8.
9. Scaman, M. E., & Economikos, L. (1995). Computer vision for automatic inspection of complex metal patterns on multichip

modules (MCD-M). *IEEE Transactions on Components, Packaging and Manufacturing Technology, 18*(4), 675–684.
10. Wu, W. Y., Wang, M.-J. J., & Liu, C. M. (1996). Automated inspection of printed circuit boards through machine vision. *Computers in Industry, 28*(2), 103–111.
11. Loh, H. H., & Lu, M. S. (1999). Printed circuit board inspection using image analysis. *IEEE Transactions on Industry Applications, 35*(2), 426–432.
12. Kim, N. H., Pyun, J. Y., Choi, K. S., Choi, B. D., & Ko, S. J. (2001). Real-time inspection system for printed circuit boards. In *Proc. 6th IEEE Int. Symp. Industrial Electronics*, pp. 12–16.
13. Tominaga, S., & Okamoto, S. (2003). Reflectance-based material classification for printed circuit boards. In *Proc. 12th Int. Conf. Image Analysis and Processing*, pp. 238–243.
14. Ibrahim, Z., & Al-Attas, S. A. R. (2004). Wavelet-based printed circuit board inspection system. *International Journal of Signal Processing, 1*(1), 73–79.
15. Garcia, H. C., Villalobos, J. R., & Runger, G. C. (2006). An automated feature selection method for visual inspection systems. *IEEE Transactions on Automation Science and Engineering, 3*(4), 394–406.
16. Acciani, G., Brunetti, G., & Fornarelli, G. (2006). Application of neural networks in optical inspection and classification of solder joints in surface mount technology. *IEEE Transactions on Industrial Informatics, 2*(3), 200–209.
17. Chang, P. C., Chen, L. Y., & Fan, C. Y. (2008). A case-based evolutionary model for defect feature segmentation of printed circuit board images. *Journal of Intelligent Manufacturing, 19*(2), 203–214.
18. Huttenlocher, D., Klanderman, G., & Rucklidge, W. (1993). Comparing images using the Hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 15*(9), 850–863.
19. Cortes, C., & Vapnik, V. (1995). Support vector networks. *Machine Learning, 20*, 273–297.
20. Vapnik, V. (1998). *Statistical learning theory*. Chichester: Wiley.
21. Schölkopf, B., & Smola, A. J. (2001). *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT Press.
22. Bazaraa, M., & Shetty, C. M. (1979). *Nonlinear programming*. New York: Wiley.
23. Platt, J. C., Cristianini, N., & Shawe-Taylor, J. (2000). Large margin DAG's for multiclass classification. *Adv. Neural Information Processing Systems*, vol. 12, pp. 547–553. MIT Press.
24. Hsu, C. W., & Lin, C. J. (2002). A comparison of methods for multi-class SVMs. *IEEE Transactions on Neural Networks, 13*(2), 415–425.
25. Wu, T. F., Lin, C. J., & Weng, R. C. (2004). Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine learning Research, 5*, 975–1005.
26. Cover, T. M., & Thomas, J. A. (1991). *Elements of information theory*. Wiley.
27. Gonzalez, R. C., & Woods, R. E. (2001). *Digital image processing* (2nd edn.). Prentice Hall.
28. Benhabib, B., Charette, C. R., Smith, K. C., & Yip, A. M. (1990). Automatic visual inspection of printed circuit boards: an experimental system. *International Journal of Robotics & Automation, 5*(2), 1034–1044.
29. Widrow, B., & Hoff, M. E. (1960). *Adaptive switching circuits. IRE WESCON Convention Record, Part 4* (pp. 96–104). New York: IRE.

Chia-Te Liao received his B.S. degree in computer science and information engineering from National Chi Nan University, and M.S. degree in computer science and information engineering from National Chung Cheng University, in 2001 and 2003, respectively. He is currently a Ph.D. candidate in the department of computer science at National Tsing Hua University. His research interests include pattern recognition and machine learning.

Wen-Hao Lee received his B.S. degree in computer science and information engineering from National Central University, and M.S. degree in computer science and information engineering from National Tsing Hua University, in 2005 and 2007, respectively. His research interests include image processing and computer vision.

Shang-Hong Lai received the B.S. and M.S. degrees in electrical engineering from National Tsing Hua University, Hsinchu, Taiwan, and the Ph.D. degree in electrical and computer engineering from

University of Florida, Gainesville, in 1986, 1988 and 1995, respectively. He joined Siemens Corporate Research in Princeton, New Jersey, as a member of technical staff in 1995. Since 1999, he became a faculty member in the Department of Computer Science, National Tsing Hua University, Taiwan. He is currently a professor in the same department. In 2004, he was a visiting scholar with Princeton University. Dr. Lai's research interests include computer vision, visual computing, pattern recognition, medical imaging, and multimedia signal processing. He has authored more than 150 papers published in the related international journals and conferences. He holds ten US patents for inventions related to computer vision and medical image analysis. He has been a member of program committee of several international conferences, including CVPR, ICCV, ECCV, ACCV, ICPR and ICME.