01 Jan 1993

Visual Inspection Algorithms for Printed Circuit Board Patterns A SURVEY

Madhav Moganti

Fikret Erçal
Missouri University of Science and Technology, ercal@mst.edu

Cihan H. Dagli
Missouri University of Science and Technology, dagli@mst.edu

Visual Inspection Algorithms
for Printed Circuit Board Patterns
A SURVEY

M. Moganti, F. Ercal, and C. H. Dagli

CSc-93-27

Department of Computer Science
University of Missouri - Rolla
Rolla, MO 65401

Visual Inspection Algorithms for Printed Circuit Board Patterns: A SURVEY[†]

Madhav Moganti[1]‡ **Fikret Ercal**[2]‡
Department of Computer Science
University of Missouri - Rolla, Rolla, MO 65401

Cihan H. Dagli[3]
Department of Engineering Management
University of Missouri - Rolla, Rolla, MO 65401

Abstract

The importance of the inspection process has been magnified by the requirements of the modern manufacturing environment. In electronics mass-production manufacturing facilities, an attempt is often made to achieve 100 % quality assurance of all parts, subassemblies, and finished goods. A variety of approaches for automated visual inspection of printed circuits have been reported over the last two decades. In this survey, algorithms and techniques for the automated inspection of printed circuit boards are examined. A classification tree for these algorithms is presented and the algorithms are grouped according to this classification. This survey concentrates mainly on image analysis and fault detection strategies, these also include the state-of-the-art techniques. Finally, limitations of current inspection systems are summarized.

KEYWORDS : printed circuit board, reference comparison, graph matching, morphological processing, design-rule checking, model-based technique, feature matching.

[0]†This work was supported by the Intelligent Systems Center of the University of Missouri-Rolla
[0]‡Intelligent Systems Center, University of Missouri-Rolla, Rolla MO 65401
[01] mmoganti@cs.umr.edu; [2] ercal@cs.umr.edu; [3] c3260@umrvmb.umr.edu

Contents

List of Figures

1 Introduction

Many important applications of vision are found in the manufacturing and defense industries. In particular, the problems in manufacturing where vision continues to play a major role are inspection, measurements, and some assembly tasks. The order among these topics closely reflects the manufacturing needs. In most mass-production manufacturing facilities, an attempt is often made to achieve 100 % quality assurance of all parts, subassemblies, and finished products. One of the most difficult tasks in this process is that of inspecting for visual appearance - an inspection that seeks to identify both functional and cosmetic defects. With the advances in high speed, large memory and less expensive computers, image processing, pattern recognition, and artificial intelligence have resulted in better and cheaper industrial image analysis equipment. This made the electronics industry active in applying automated visual inspection in manufacturing/fabricating processes that include printed circuit boards, IC chips, photomasks, etc. Nello [1] gives a summary of the machine vision inspection applications in electronics industry.

Human operators monitor the results of the more than fifty process steps required to fabricate a printed circuit board (PCB). They simply inspect the work visually against prescribed standards. These decisions made by human inspectors often involve subjective judgment, in addition to its being labor intensive [2] and therefore costly. Whereas automatic inspection systems remove the subjective aspects and provide fast, quantitative dimensional assessments. These systems do not get tired, do not suffer burnouts and are consistent day in and day out. Applied at each appropriate step of the assembly process they can prevent value being added after a defect has occurred, reduce rework costs, and make electrical test more efficient. All of this means better quality at lower cost. [3, 4, 5, 6] have emphasized the importance of automatic inspection systems in the electronics industry.

The major PCB manufacturing stages and process steps involve bareboard fabrication, loaded board assembly, soldered board process. The increase in automated production line technology has initiated substitutes for human visual inspection rapidly. These systems have been produced with distinct and limited capabilities for covering the fault spectrum at each significant stage of PCB manufacture [5]. Even to date, automatic bare PCB inspection is considered to be the most matured industrial visual inspection application. The problem of loaded board and soldered board inspection have been addressed but the results are typically limited to detection of more noticeable discrepancies. Due to the following criteria, the sophistication in automated visual inspection has become a part of modern manufacturing environment [6, 7]:

- relieve human inspectors of the tedious jobs involved,
- industry set quality levels so high that sampling inspection is not applicable,
- production rates so high that manual inspection is not feasible,
- tolerances so tight that manual visual inspection is inadequate,

- configuration management and defect tracing require computer assistance,
- the reasons that circuit boards are becoming increasingly more complex while the circuit board features themselves are becoming smaller has made visual inspection by human operators impossible. Progress in surface-mount technology has resulted in a swift gain in the mounting density among PCBs. This in turn contributed to improved functions, enhanced performance and diminishing size among PCBs, which in turn has contributed to the complexity of the inspection problem, and
- as the packaging technologies become increasingly complex, substrates become more costly.

Most vision systems for automated industrial inspection are custom designed, so they are suitable only for one specific application. A variety of approaches for automated optical inspection of printed circuit boards (PCB's) have been reported over the last two decades. Earlier studies (surveys), [8, 9, 10] have a list of papers which have been published till 1987. Since then, there has been significant improvements in the field to justify a new survey study. In this survey, algorithms and techniques for the automated inspection of PCB's are examined. We concentrate mainly on image analysis and fault detection strategies, which include the state-of-the-art techniques. Limitations of current inspection systems are presented. One of the goals of this study is to collect most (if not all) of the articles in this field published to date, to classify and discuss them according to the methodologies employed. All of these will be discussed under a consistent set of terminologies (where variations will be mentioned) in the hope that such a unified treatment would be helpful.

1.1 Types of Inspection

Inspection procedures can be broadly divided into two classes: electrical/contact methods and non-electrical/non-contact methods. Electrical test can find flaws such as shorts and opens; the others require some other methods of detection. Some of the advantages and disadvantages of these methods are given in [11]. An image of a PCB can be acquired using visible or invisible light and then analyzed for defects. Most common and reliable methods reported in the literature have made use of light in the visible part of the spectrum. This section briefly lists some of the different inspection systems based on different imaging technologies. Some of the non-contact automatic inspection methods that are currently available in industries are [12, 13]:

- **Automatic Visual/Optical inspection** Optical testers can find defects other than shorts, and opens, such as line width errors, pad mouse bites, and trace misplacements. This paper focuses on this inspection method.
- **X-ray imaging** is used for rapid and precise measurements of multilayer PCBs. Based on the measurements of individual test pads or test coupons, the system supplies specific information on layer registration, distortion and the torsion of the layers. X-rays also reveal minute defects, such as

hairline cracks around the via, which escape other methods of inspection. SMD defects like heel cracking, voids, component misalignment, bridging, insufficient solder, excess solder, solder threads and balls, poor wetting and bent leads can be detected using X-rays.

- **Scanned-Beam Laminography** provide cross-sectional X-ray imaging which separates the top and bottom sides, or any other layer to the PCB, into cleanly separated images. The basic principle of laminography is to move the X-ray source and the X-ray image detector around on opposite sides of the object. As long as the X-ray beam always passes through the same points in the object and the same points in the detector simultaneously, a cross-sectional image is formed in real-time. By changing the size of the X-ray scanning circle, the field of view and magnification of the image can be varied on the fly. This enables inspection of fine-pitch components at high magnification and of other components at normal magnification to optimize throughput.
- **Thermal imaging systems** Indicate hot spots on operating PCBs pointing out shorts and overstressed components. Usually these systems find success in applications where automated measurement of heat is utilized to understand process performance or where temperature measurement and control are vital to process yield.

1.2 Defects

Printed circuit boards are inspected extensively before the insertion of components and the soldiering process to isolate defects (also called anomalies or faults). Even though automated approaches are used in the verification of artwork [14], before beginning actual etching process on the board, bareboard defects still exist. Wesley Hall [14] outlines the processing and post processing involved in the verification of artwork design. A variety of defects can afflict the copper pattern of PCBs; not all mean immediate rejection of the board from consideration. The types of faults range from hair-line (eg. size equal to 100 microns) breaks and bridges as small as 1 mm between conductor paths to unacceptable enlargements and reductions in line widths to poorly formed plated through holes. The anomalies looked at, for example are: *unetched copper, open (break or cut), partial open (mousebite or nicks), scratches or cracks, shorts, incipient short (fine wiring), overetching, underetching (abnormal wire width) , pad size violations, spurious (excess or residual) metal, spurs (protrusions or whiskers or smears), cracking of walls of holes, violations of spacing of holes, violation of spacing of conductor traces, etc.* A wide variety of terminology is used in naming these faults. Above list gives the commonly associated names used in naming the defects, followed by other popular/unpopular names in parenthesis. Figure 1 shows an artificial defect free PCB image pattern. This figure depicts through hole PCB patterns, printed wiring board patterns, and surface mount PCB patterns in the same image. This is because as most of the defects are common to all the three varieties of boards, the three different patterns are shown in one example image. Figure 2 shows the same image pat-

tern as in Figure 1 with a variety of defects shown in it. Though each defect shown in the figure is a representative example for that particular defect, the shape and size of the defect varies from one occurrence to the other. Smaller and smaller lines and spaces make these defects more serious, more likely and harder to detect. According to many quality studies, open/partial open, short, pinhole, spurious copper, overetch, breakout are the most frequent defects that occur. These defects are caused due to one or more of the following errors [11]:

- caused by thermal expansion of the artwork during printing, or by defective etching,
- dirt on board, air bubbles from electrolysis,
- incorrect electrolysis timing,
- mechanical misregistrations,
- distortions of the PCB due to warping, etc.

Thibadeau in [15] gives a good summary of some defects that occur in printed boards with causes that occur during fabrication. Inspection of bare PCBs demand:

- high-speed (about 30 seconds [5]),
- high data rate,
- high detection accuracy, and
- a low false-alarm rate.

The dimensional variations in the conductor spacings and widths due to seasonal temperature and humidity changes should be taken into account. Further, 1mm faults require at least 0.5 mm imaging resolution, therefore dust, hair, lint and fingerprints become unwanted noise sources for false alarms, making clean-room conditions necessary [5]. Although it is possible to detect initial defects such as conductor breaks and short circuits through conductor tests, these tests cannot reveal overetched conductors, limited conductor spacing, and other defects that can lead to deterioration with age [16].

2 Components and Terminology Involved

This section briefly defines the most commonly used terminology in this field. The reader is not provided with any rigorous and complete definitions. Interested readers are advised to refer recent picture processing or machine vision text books to get a complete understanding of the individual subjects involved. This section also identifies the major components that an inspection system consists of.

Though there is distinction between printed circuit boards, printed wiring boards, and surface mount boards, here in this paper we use the generic term *printed circuit board* (PCB) to refer all of them. This is because most of the defects and defect analysis techniques are common for all of them.

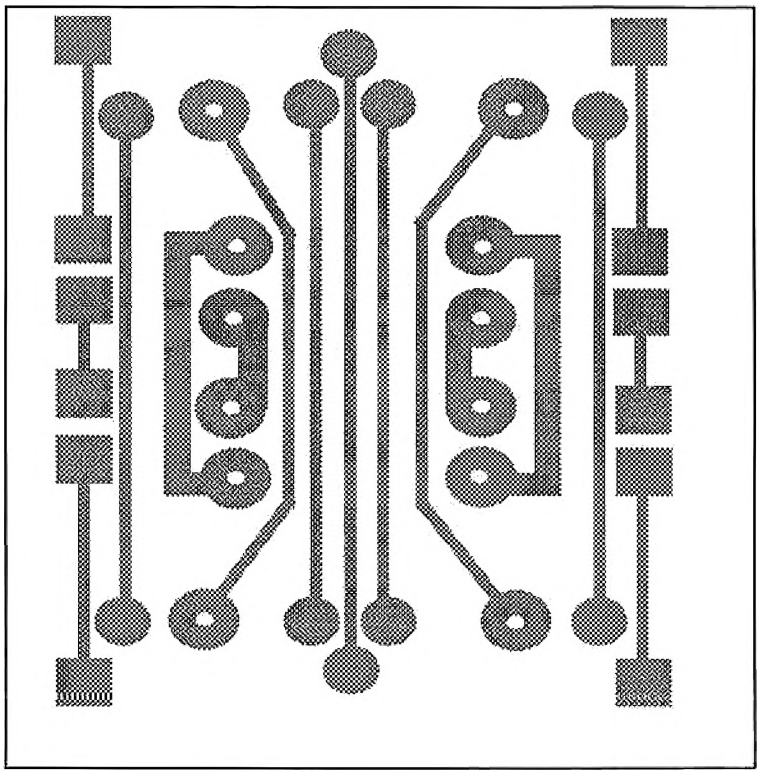
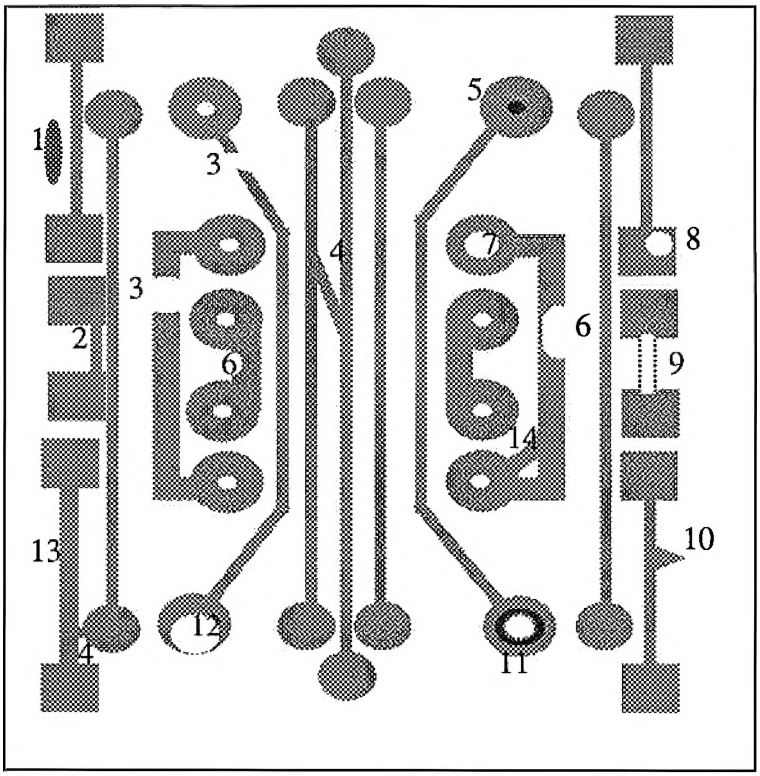


Figure 1: Example PCB pattern



1. Spurious Copper
2. Conductor Too Close
3. Open Circuit (break)
4. Short
5. Missing Hole
6. Mouse Bite (nick)
7. Wrong Size Hole
8. Pine Hole (void)
9. Missing Conductor
10. Spur (protrusion)
11. Overetch
12. Breakout
13. Underetch
14. Excessive Short

Figure 2: Example PCB pattern showing defects

A typical inspection process involves observing the same type of object repeatedly to detect anomalies. The process involves digitization of the object to be inspected for visual data and the analysis involves the processing of the imagery to enhance relevant features and the detection of defects. The inspection procedure of such systems is first to precompile a description of each of a known set of defects and then to use these models to detect defects in an image. Another procedure is to model the part by its normal, expected features and then to use the part model to verify in an image that the part under inspection has all the expected features. Foster *et al* [7] and Chin [17] outlined the tasks involved in inspection of printed circuit boards, and industrial inspection in general. The major components involved in automated visual inspection systems related to image processing are:

Hardware System The main hardware components of the inspection system are the illumination system, image acquisition system, and the processor.

Illumination System: The main parameters that characterize the suitability of an illumination system to acquire an image of good quality are: (a) *intensity*, (b) *uniformity*, (c) *directionality*, and (d) *spectral profile*. The relative importance of these parameters and the degree to which each one must be controlled are largely governed by the surface characteristics of a given PCB and the constraints imposed by the camera. Most of the systems that are built to date either require good lighting conditions or they employ different lighting techniques. Among the lighting techniques most commonly used are: standard light sources, indirect and back lighting, fluorescent lighting, reflected lighting, diffuse illumination [18], fiber-optic, quartz-halogen light sources [19], etc.

Image Acquisition System: Usually consists of a camera or a digitizer that acts as a sensor. There are a several type of cameras available and the determination of the appropriate type is dictated by use. Examples of different types are television camera, a charged coupled device camera, etc. AOI System Corp. developed the AOI-20 system that utilizes as many as 20 CCD cameras [16].

Processor: The processor system usually consists of a high speed computer system. A commercially available inspection systems, AOI-20, uses a high speed parallel processing system [16]. Usually most of the commercially available systems have special processors designed solely for inspection purposes.

Image Enhancement: Involves removal of noise, enhancement of edges, enhancement of contrast, etc. Thresholding (point processing operation), convolution (group processing operation), and picture processing (processing over the entire image) are some of the techniques used for enhancement of the images [20, 21].

Feature Extraction: The decision regarding what features to be considered is rather subjective and depends on practical situations. Features are less sensitive with respect to the encountered variations of the original noisy gray-scale images and provide data reduction while preserving the information required for the inspection. Most of the procedures used for feature extraction are simple edge-detection, line tracing, and object shape properties.

Model-Based system: The most common inspection technique is the model based process which perform inspection by matching the part under

inspection with a set of predefined models.

Modeling: Involves *training*, in which the user uses a model part to teach the system the features to be examined, their relations, and their acceptable tolerances.

Detection/Verification: Consists of matching the extracted features from the part under inspection with those of the model. A typical detection procedure involves simple comparison operation. These methods are computationally intensive if computed digitally. The detection process becomes very ineffective if the part to be inspected is noisy and is located at random positions. Detection using representative features and their relationships provide a way to inspect a part and locate defects on the basis of measurements taken from key features. This approach is more robust and effective.

Boundary Analysis: in which models of good boundaries are compared with those of the board being inspected [22, 23].

Thinning, Contraction and Expansion: These are image-to-image transformation operations [24, 25]. These operations are defined using neighborhood connectivity relations. An expansion sets all background pixels in an image to foreground pixel value, if any one of the neighboring pixel value is equal to foreground pixel value. Contraction is realized by first expanding the complement of an image and then taking the complement of the result. Thinning reduces an entity to its skeleton, a simplified version contained in the original entity that retains the basic shape of an entity. Unlike expansion or contraction, thinning maintains the connectivity [26] of an entity and preserves its holes (none are removed or added). Different definitions and implementations of these operations can be found in [27, 28, 29, 30].

Morphology: Refers to a branch of nonlinear image processing and analysis. The basic idea is to probe an image with a structuring element and to quantify the manner in which the structuring element fits (or does not fit) within the image. The operations of dilation, erosion, opening, closing, etc are used in this type of image processing. A complete treatment on this subject can be found in [31, 32].

Algorithms

A large number of PCB inspection algorithms have been proposed in the literature, Figure 3 shows the classification of these algorithms. In general, they fall into one of three categories: reference comparison (or referential approach), non-referential approach, and hybrid approaches - which involve a combination of more than one of the methods. The reference comparison approach uses complete knowledge of the circuit under test, whereas the design-rule verification approach uses knowledge of properties common to a circuit family but not knowledge of the specific circuit under test. There are two types of reference comparison methods: the simpler approaches involve some kind of direct image comparison, between pixels in the test image and in an idealized reference image. Somewhat more sophisticated approaches involve recognition of circuit features in the test image followed by a comparison against a set of reference features. The non-referential approaches either work on the assumption that

features are simple geometric shapes and the defects are unexpected irregular features or on directly verifying the design rules. Basically these methods, use local neighborhood processing techniques over the image to be inspected. In these methods the task is to determine whether each feature falls within the required dimensions. This approach does not require precise alignment, but might miss large flaws and distorted features.

3 Referential Modeling

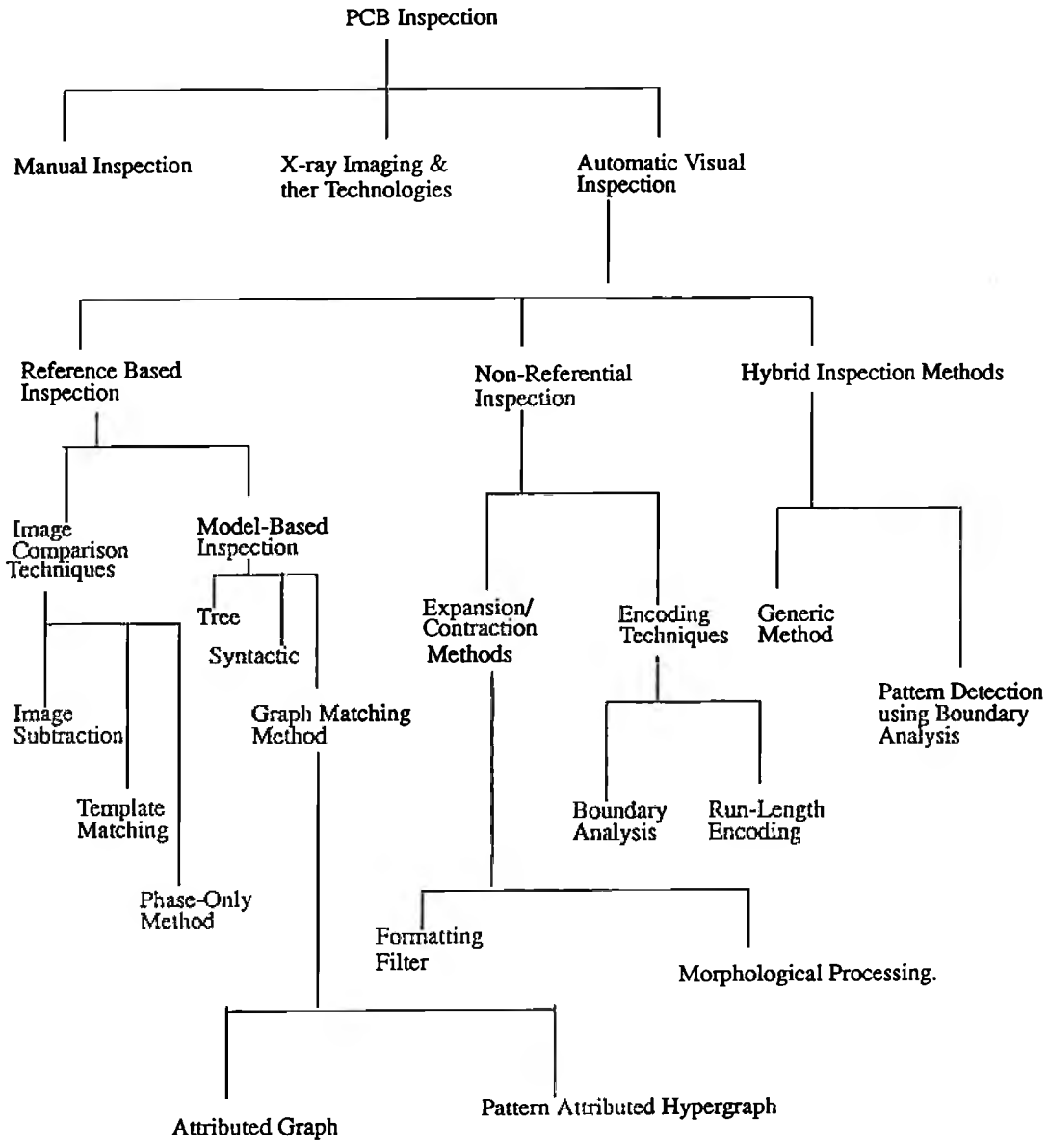
3.1 Image Comparison Techniques

3.1.1 Image Subtraction

Image subtraction is the most simple and direct approach to the PCB inspection problem. The board to be inspected is scanned and its image is compared against the image of an ideal part. The subtracted image, showing defects, can subsequently be displayed and analyzed. Figure 4 shows this direct subtraction process as a logical XOR operation on the subimage patterns of the PCB. This technique suffers from many practical problems, including registration, color variation, reflectivity variation, and lighting sensitivity. One other problem is that statistical analysis must be performed to determine if differences are due to nonconformities or due to alignment. Hara *et al* [33] has experimented with the image subtraction technique by comparing fluorescent light images. The paper shows the superiority of this method over reflected light inspection system [34].

3.1.2 Feature Matching

Feature matching is an improved form of the image subtraction, where the extracted features from the object and those defined by the model are compared. The advantage of this matching is that it greatly compresses the data for storage, and at the same time reduces the sensitivity of the input data and enhances the robustness of the system. This matching process is called *template matching.* One of, the major limitations of template matching for inspection is that an enormous number of templates must often be used, making the procedure computationally expensive. This problem can be eliminated if the features to be matched are invariant of size, location and rotation. Example template matching procedure is cross-correlation matching followed by a scalar subtraction measurements. The disadvantages of this method are that it requires a large data storage for the ideal PCB patterns, precise registration is necessary for comparison. It is sensitive to illumination and digitization conditions, and the method lacks flexibility. Hara *et al* [33, 35, 36] uses a defect detection method based on feature extraction and comparison. Large defects are detected by extraction of boundaries in a direction different from that of the boundaries of the reference pattern using H_{KX}, H_{KY}, H_{K45}, and H_{K-45} operator templates in the four directions $(0°, 90°, +45°, -45°)$ as shown in Figure 5(a). This method is used for detection of all defects of width greater than a fixed value and for isolated defects. Narrow defects, like fine wiring and whiskers are detected


PCB Inspection
Manual Inspection
X-ray Imaging & ther Technologies
Automatic Visual Inspection
Reference Based Inspection
Non-Referential Inspection
Hybrid Inspection Methods
Image Comparison Techniques
Model-Based Inspection
Tree
Syntactic
Graph Matching Method
Image Subtraction
Template Matching
Phase-Only Method
Expansion/ Contraction Methods
Encoding Techniques
Generic Method
Pattern Detection using Boundary Analysis
Boundary Analysis
Run-Length Encoding
Formatting Filter
Morphological Processing.
Attributed Graph
Pattern Attributed Hypergraph

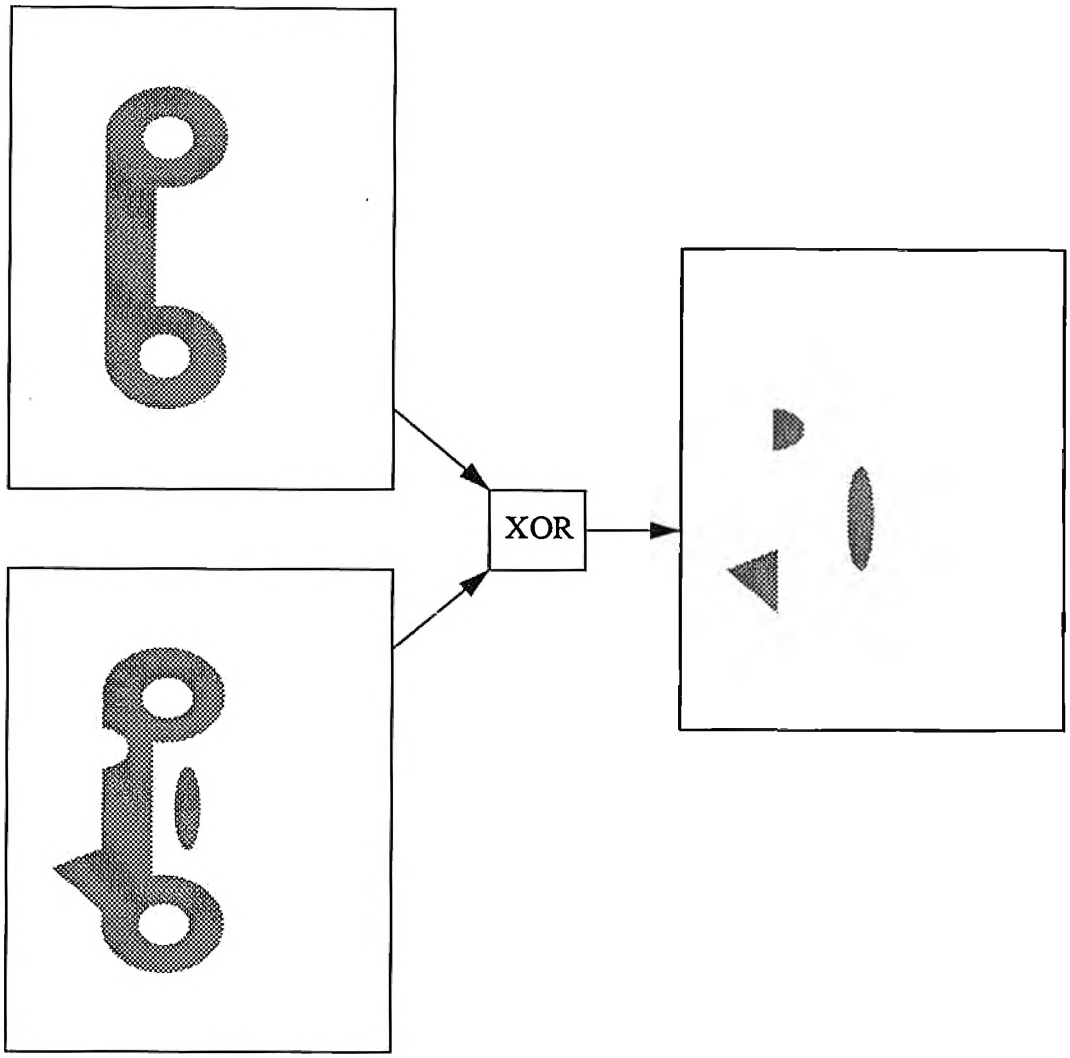


Figure 4: Image Subtraction

by extracting the fine patterns using $h^{\phi}_{BY}, h^{\phi}_{BX}, h^{\phi}_{B45}$, and h^{ϕ}_{B-45} operator templates, shown in Figure 5(b), searching in four directions $(0^o, 90^o, +45^o, -45^o)$. The final result of extraction is a logical AND of the four direction features extracted. The sizes of the templates H_{**} are not fixed and can be regulated by setting limits on the length, inclination and widths of the patterns. These different sizes are necessary to precisely identify the boundaries, as the trace pattern widths may change and also big hops can be made using larger template sizes in the uninteresting regions (eg. which do not have trace pixels), thus reducing unnecessary computation time. The h^{ϕ}_{**} operators detect narrow (fine) defects not extracted as boundary lines using H_{**} operators. The sizes of these templates depend on the widths of the flaws to be detected. Figure 5(c) shows a PCB sub-image pattern with its boundary extracted using H_{**} and h^{ϕ}_{*} feature extraction operators. Figure 5(d) shows a defective PCB pattern and its boundary. Figure 5(e) and 5(f) show the application of H_{KY} and h^{ϕ}_{BX} operators respectively. The comparison step involves the comparison of the extracted features of the reference image with that of the extracted features of the test image.

3.1.3 Phase-Only Method

David *et al* in their paper [37] discusses an alternative method to standard template matching technique which is based on phase-only imaging. A phase only image is an image which has unit power spectral density amplitude so that all information is contained in the phase. Phase-only image comparison has the properties of redundancy removal and edge enhancement. The method uses Fourier transform, whitens (normalizing the resultant image to spread over the entire grey scale range), and then inverse Fourier transforms an image pair to produce a map of significant image differences. Because the correlation of any pair of data points in the image are removed, all periodic components of the image gets suppressed. Two similar images can be compared by creating a composite image by placing them side-by-side and applying a phase-only transformation at once. If the two images are very similar, a strong periodic component with period equal to the subimage spacing appears in the spectrum of the composite image. By suppressing this component, all points which correspond to the two subimages will be suppressed, and only the differences remain. The paper presented examples of real and simulated images with different illumination levels, lighting gradients and board substrate colors, all compared with the same master reference.

This method has advantages over conventional template matching/comparison techniques because of its light intensity invariance, insensitivity to illumination gradients, tolerance to misregistration of the images to be compared, and invariance to translation. The method suffers from the disadvantage that it requires considerably large amount of computational time compared to simple template matching methods.

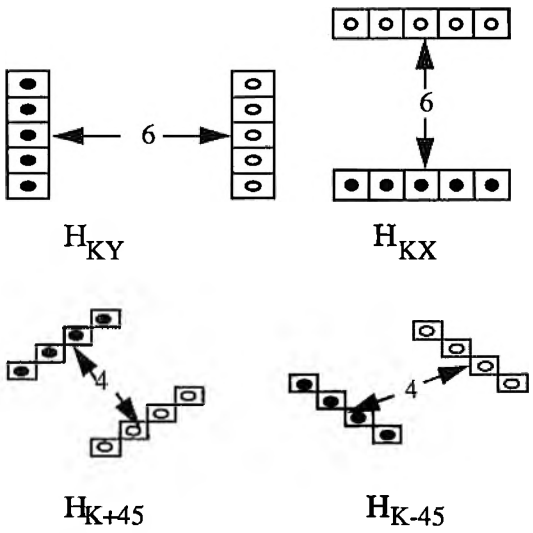


Figure 5 (a):Extraction operators for Boundary

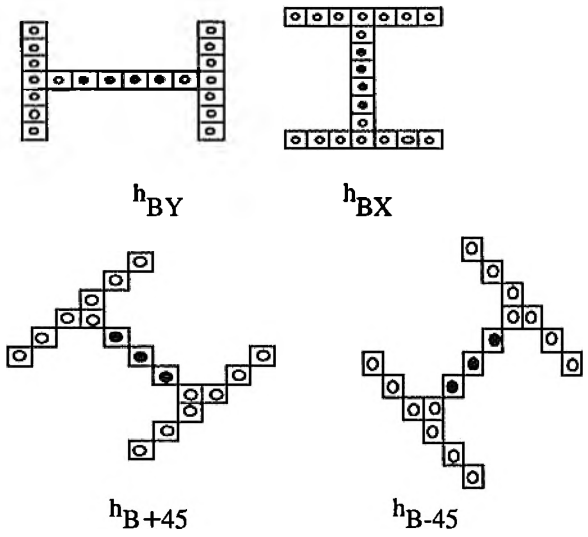


Figure 5(b): Narrow defects extraction operators

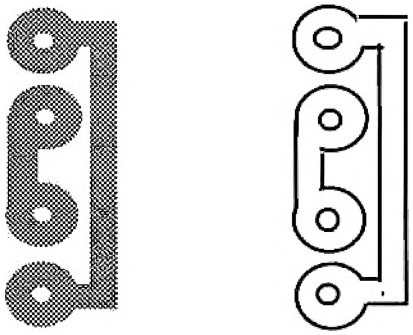


Figure 5(c): PCB sub-image pattern and its boundary obtained after applying H_{**} and h_{*} operators.

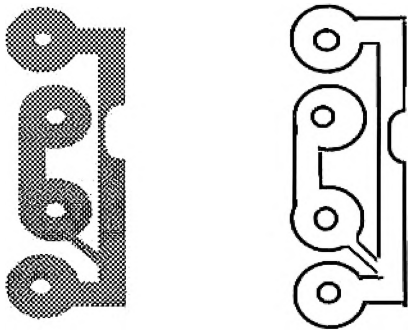


Figure 5(d): Defective PCB sub-image pattern and its boundary obtained after applying H_{**} and h_{*} operators.

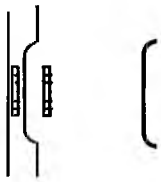


Figure 5(e): H_{KY} application

Figure 5(f): h_{BX} application

3.2 Model-Based Methods

The selection of a suitable model representational structure strongly effects the performance of the system. There are many structures used for model representation, such as the string, the tree, and the graph. One of the approaches that falls into model based techniques is the syntactic approach, also called string matching technique. In syntactic approach [38, 39] a PCB image is encoded into finite alphabets. The method involves tracing the boundary to produce an ordered list of boundary points, and analyzing the shape to produce syntactic description of the shape. The detection of defects then involves the detection of local defective features expressed in finite expressions. One major limitation of this approach is that the choice of primitives in quantifying the basic shape involved in the patterns is a difficult problem. This makes the approach not applicable for a real time application like this.

3.2.1 Graph Matching Methods

The graph matching methods are based on the structural, topological, and geometric properties of the image. The idea is based on the topological/structural comparison which compares the standard graph obtained from the conductors and insulator images of the reference PCB with those of inspection boards. For example topological information incorporates a weighted graph composed of several types of nodes, edges, connections, and their location [40].

3.2.1.1 Attributed Graph

Darwish *et al* [27] proposed a method that works in two main steps. In the first step, the image is transformed into a collection of nodes that describes the 2-D shape of the different objects in the image. These nodes are connected together depending on relational properties between primitives belonging to the same object and between different objects. Spatial relations are added to the graph in the form of directed attributes, which describes connectivity and neighborhood relationships. This graph is called an *attributed graph*(AG). The second step involves model verification process. This matching process between the inspected and model patterns is the most time-consuming step during inspection. A similarity evaluation function is used to measure how well the scene graph matches the model graph. The complexity of matching AGs is very large, since every node of an AG joins the coupling permutation at each iteration for every attributed relationship. This problem is overcome in [28] by reducing the large amount of unnecessary computations done in evaluating scores between impossible couples during the exhaustive permutations. The following section discusses the improvised method.

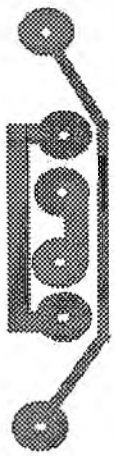
3.2.1.2 Pattern Attributed Hypergraph

Sun and Tsai [28] presents a representation called pattern attributed hypergraph (PAHG) and a structural inspection algorithm. The proposed graph, called PAHG, describes all segmented regions and the spatial relationship among

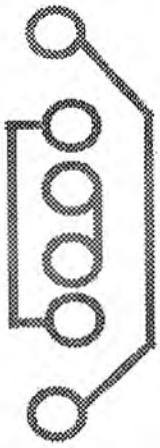
them. These segmented regions are represented by a regional attributed graph (RAG) that represents a set of primitive features connected to one another within a region, which is the bottom level of PAHG. The top level of PAHG contains regional features and the spatial relations among them. This representation allows to prune the search space by performing only selective matching operations during the matching phase, thereby reducing the inspection time. This new representation where the information is represented in two different levels is a major improvement over the attributed graph method. Figures 6(a), 6(b) and 6(c) show all the steps involved in the construction of the bottom level of PAHG. This step involves thinning of the binary image, then smoothing the thinned image using pruning operation in order to eliminate spurious effects in thinning and then labeling the pruned pattern. Figure 6(a) is thinned to obtain Figure 6(b). Figure 6(c) is the labeled graph obtained after pruning the Figure 6(b). Figure 6(d) shows the RAG constructed for the sub-pattern of the PCB pattern A. Figure 6(e) shows the PAHG for the complete PCB sub-pattern shown in Figure 6(a). The matching algorithm proposed works by (a) verifying the top level of PAHG on the scene model and reference model (Figures 6(e) and 6(g) are compared at this level, where the faults like open are easily caught), (b) finding the corresponding pairs of RAG's by evaluating the confidence scores between two PAHGs and the pair of RAGs, and (c) verifying each RAG of the scene model with the corresponding RAG of the reference model.

4 Non-Referential Inspection

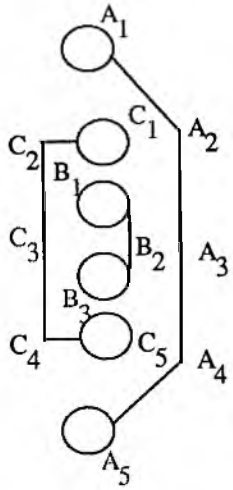
Non-Referential methods do not need any reference pattern to work with, they work on idea that a pattern is defective if it does not conform with the design specification standards. They basically use the design-specification knowledge in verifying the board to be inspected. Applying the design-rule verification process directly to the image patterns is a time consuming process, and hence the response time of the system decreases. These methods are also called design-rule verification methods, or generic property verification methods. Usually these methods process/transform the image into a form which reduces the verification time. Expansion-contraction methods employ pixel-neighborhood processing operations like, expansion, contraction, thinning, or morphological operations like erosion, dilation, etc in the pre-processing stage. The operators are designed in such a way that they embed the design specifications in them and the result of applying these operators directly reflects the discrepancies in the image patterns, if any exist. Design-specification information is embedded in these operators, such that the transformations generate images that could be easily interpreted for defects. The advantage of these operations is that they are simple to apply and easy to implement in hardware. Encoding techniques also transform the image patterns and the verification phase involves interpreting these transformed patterns: by extracting the topological features and imposing localized constraints such as minimum or maximum widths to detect anomalies. The disadvantage of these non-referential methods is that



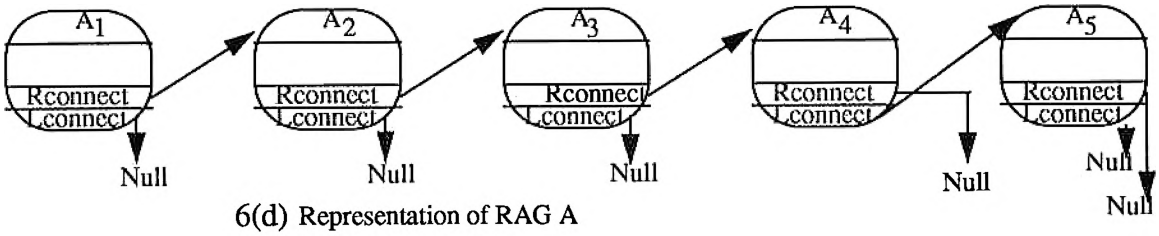
6(a) PCB Sub-Image Pattern



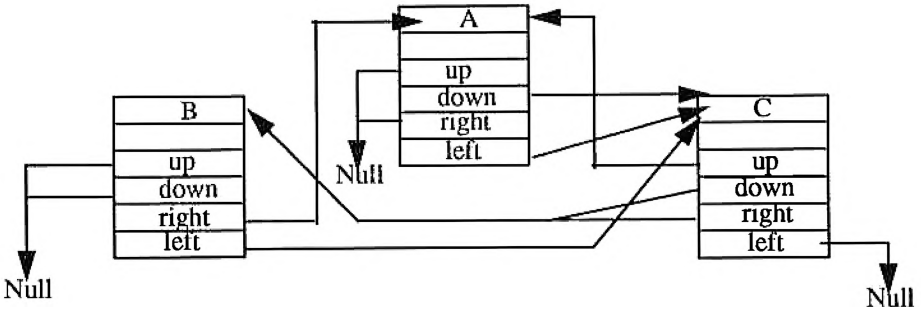
6(b) PCB Thinned Pattern



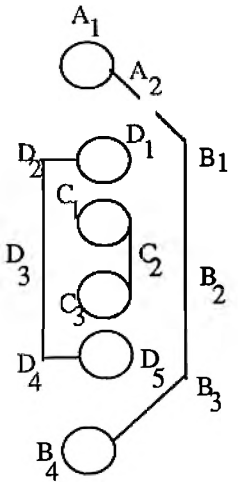
6(c) Pruned PCB pattern with lables



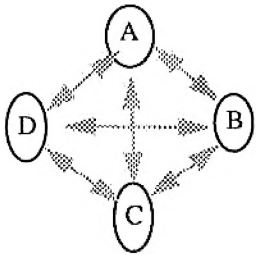
6(d) Representation of RAG A



6(e) Graphical Representation of PAHG for the Sub Image



6(f) Pruned defective PCB pattern with labeling



6(g) PAHG for Fig. 6(f).

work well in identifying only some kind of defects, such as in the verification of widths and spacing violations.

4.1 Expansion/Contraction Methods

The inspection involves the expansion-contraction process, which does not require any predefined model of perfect patterns. Ye and Danielson [26] presented an algorithm for verifying minimum conductor and insulator trace widths. The method iteratively applies shrinking (similar to contraction operation) and connectivity preserving shrinking (similar to thinning) operations on the image. After some number of iterations, the difference (logical AND) between the results gives the defects present in the patterns. The main advantage of non-referential methods is that the alignment problem is eliminated. But, the problem with these methods is that different pre-processing algorithms are to be applied to check different violations in the board, which automatically decreases the response time of the system.

4.1.1 Expansion and Contraction using Formatting Filter

Griffin *et al* discusses about a nonreferential inspection algorithm in [41, 42] which is a variation of shrinking method given by Mandeville [29]. In this method, the image is first enhanced by a formatting filter and then the connectivity through the circuit trace is checked. The formatting filter classifies each pixel of the observed circuit board into one of three types: trace type, board type or indeterminate type. A pixel is classified a trace (board) type if it is surrounded by a circle of trace (board) pixels with a minimum radius. If this radius is equal to specified minimum then at that point the trace (board) satisfies minimum trace (board) requirement. Pixels which are not classified as either trace type or board type are classified as indeterminate. Figure 7(a) shows a PCB sub-image whose output pattern would look like Figure 7(b) after format filtering. This classification provides a means to check for open/partial opens, minimum trace spacings and surface nonconformities on the circuit boards. Figure 7(c) shows a defective PCB sub-image, which has a mouse bite, wrong size hole and conductor too close defects, and whose output pattern looks like Figure 7(d) after format filtering. Opens/partial opens are identified by checking for connectivity along the trace, where failure of minimum width requirement indicates a break in the connectivity. Minimum spacing requirements are checked by verifying if there are any of the indeterminate pixels of one trace connected to indeterminate pixels of another trace, if exists, then the minimum spacing requirements are not satisfied. Surface nonconformities like scratches and dust are inspected after the algorithm for width and spacing requirements have been performed. These nonconformities are identified to be the areas of high intensity pixels by subtracting the metal trace pixels from the image whose lighting configuration is such that the source is at an acute angle to the board.

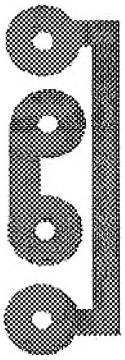


Figure 7(a): PCB Subimage

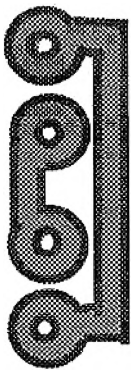


Figure 7(b):After Format Filtering

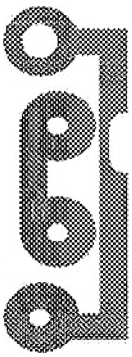


Figure 7(c): Defetcive PCB Pattern

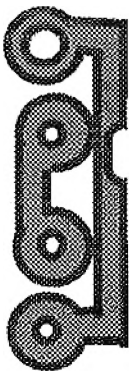


Figure 7(d): After Format Filtering

Figure 7: Expansion and Contraction Filtering

4.1.2 Morphological Processing

The system proposed by [43] makes use of defect detection algorithms which are derived using image transformations based on mathematical morphology. The system detects: violations of minimum land width requirement (MLW), violation of minimum conductor spacing requirement (MCS), and the violation of minimum conductor trace width requirement (MCTW). The fundamental operations used in the transformations are hit/miss transformation, erosion operation, dilation operation, and symmetrical thinning. The PCB images are supposed to be 3-level digital images as shown in Figure 8(a): substrate pixels with value 0, conducting structure pixel values with value 1, and holes with value 2. A segmentation algorithm which separates the conductor lands surrounding the holes from the conductor traces is employed. This enables the system to apply design rule checking easily and thus avoiding false alarms. The following steps depict the algorithm:

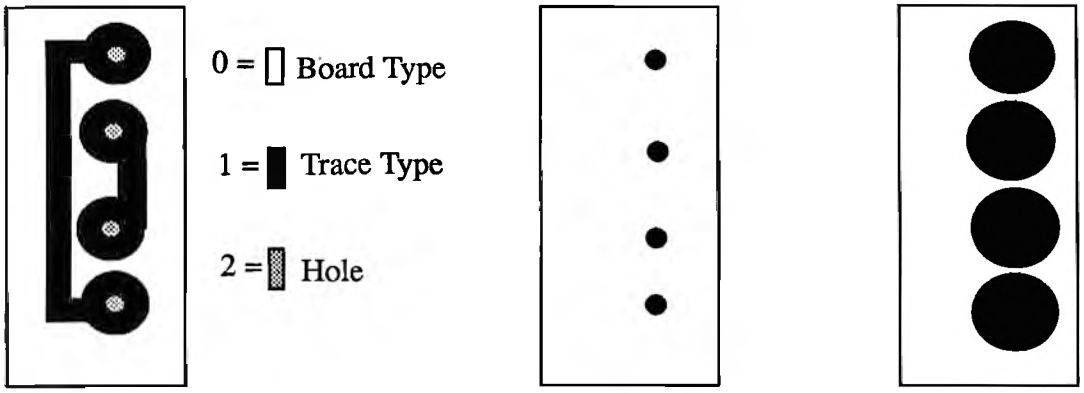
1. the original image is transformed using the following rule $0 -> 0$, $1 -> 0$ and $2 -> 1$. Figure 8(b) shows the resultant binary image.
2. the hole locations are enlarged, as shown in Figure 8(c), such that they cover the surrounding lands using dilation operation.
3. transform the original image by the rule $0 -> 0$, $1 -> 1$, and $2 -> 0$. Figure 8(d) shows the resultant binary image.
4. Images obtained in steps 2 and 3 are ANDed. The resultant image after this operation on Figures 8(c) and 8(d) is shown in Figure 8(e).
5. Images in step 3 and 4 are EXORed, resulting the conductor trace image, as shown in Figure 8(f).

Algorithm for verifying minimum conductor spacing (MCS) works as follows. The algorithm can be easily understood with the help of Figure 9, which depicts each step in the process.

Algorithm verifying MCS requirement

- dilate the original PCB image by an isotropic elliptical structuring element. The resultant image is shown in Figure 9(b).
- the above image is symmetrically thinned and pruned to remove hair like protrusions. The resultant image is ORed with the original image. Figure 9(c) shows the application of this step.
- the original image is EXORed with the image obtained in the previous step, thus obtaining defective patterns as shown in Figure 9(d).

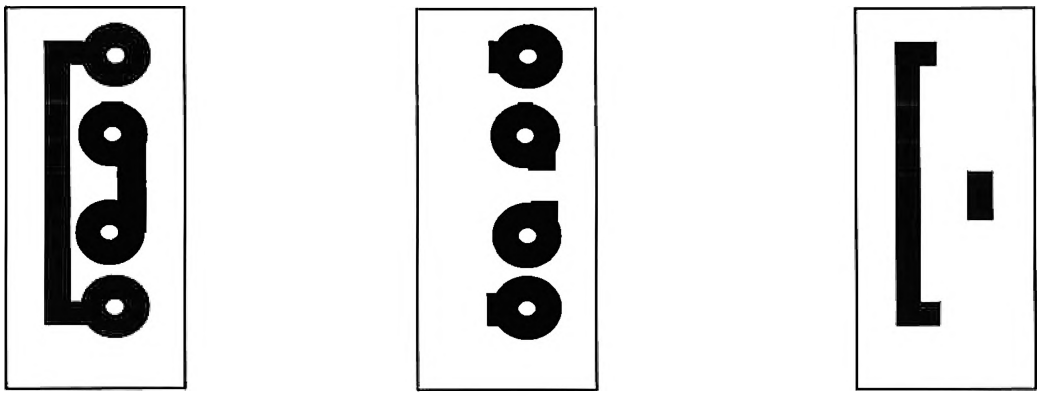
Similar algorithms are presented for verifying MLW and MCTW requirements. Also, a faster algorithm to speed-up the complete process is presented, which makes use of 2-D convolution and table look up operations as a means to implement morphological operations. The main advantage of morphological operations is that they are simple and easy to implement in hardware.



8(a):Original PCB pattern

8(b)

8(c)

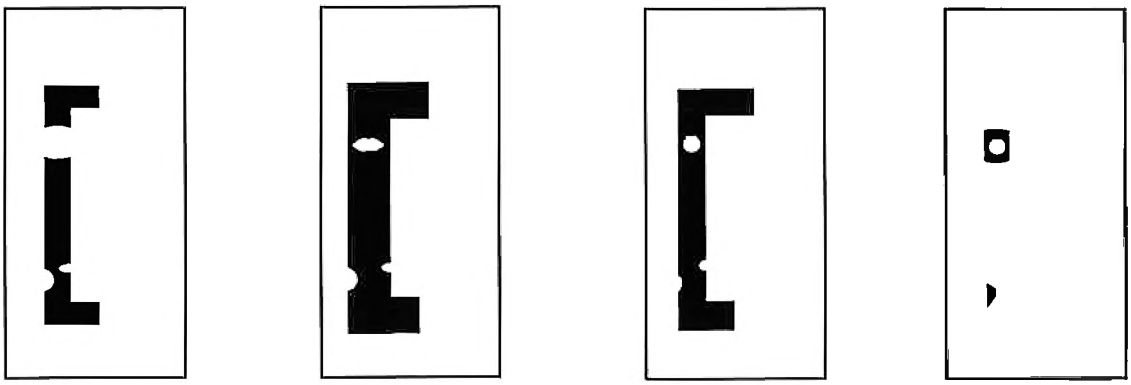


8(d)

8(e)

8(f)

Figure 8: Image Segmentation for separating conductor lands surrounding holes from other conductor traces



9(a)

9(b)

9(c)

9(d)

Figure 9: Verification of Minimum conductor spacing

4.2 Encoding Techniques

4.2.1 Boundary Analysis using Freeman Chain Coding

West *et al* [44, 45] gives a boundary analysis technique to detect faults by using Freeman chain coding [46] to describe the boundaries. For small faults, the method works in three stages: (i) compares the Euclidean distance and the boundary distance between two points on the boundary that are a constant number of chain code segments apart. The method works on the assumption that, for a normal boundary, the difference will be small, but for a defective boundary the difference will be large. (ii) Initially the adjacent curvature codes that have the same sign are combined making the sharp corners more visible in the processed corner data. This corner combination is sufficient to discriminate different faults, like nicks, bumps, etc, using the sign of the codes. These edge corners on the boundary are processed using three different corner fault models by traversing along the boundaries in a clockwise direction. Again the Euclidean distance and the boundary distance between two points on the corner models are calculated for filtering. (iii) In stage three the severity of the faults obtained in stage two is calculated. Large faults are detected by dividing the complete board into small squared regions and assigning the number of track pixels in each region to that region as the area count. Comparing this area count of each region with that of a reference board would reveal large defects.

4.2.2 Run-Length Encoding

Sterling's run length encoding method [47, 48] determines the positions of the edges of the conductor on each scan line, which provides a convenient means of linking the information on a scan line to the previous scan lines. The inspection process involves the tracking of regions from scan line to scan line, the extraction of topological features and the detection of anomalies by imposing localized constraints such as minimum and maximum conductor width. The run-length based technique developed by Thibadeau [15] analyses both vertical and horizontal histograms of run-length. The method counts continuous runs of trace pixels along every row and column of the PCB image and constructs a histogram. This histogram reflects very short horizontal runs along a horizontal edge or vertical runs along a vertical edge. Also line-width of the conductors gets reflected in the histogram which is useful to detect flaws. The conductor minimum width requirement is verified by checking if run-length of pixels is shorter than a threshold value. The main advantage of this technique is that it eliminates the need for precise alignment and enables the process to be implemented in hardware.

5 Hybrid Inspection Methods

The hybrid flaw-detection techniques increase the efficiency of the system by making use of both referential and design-rule techniques, exploiting the strengths and overcoming the weaknesses of each of the methods. These methods have

added advantage that they cover a large variety of defects compared to either referential or non-reference methods alone. For example, most of the design-rule verification methods are limited to verifying minimum conductor trace and land widths, spacing violations, etc. These methods can detect missing features or extraneous features like isolated blobs, etc.

5.1 Generic Method

The generic method is a combination of referential and non-referential inspection algorithms. As Mandeville explains in [29], it is a synthesis of reference-comparison and generic-property approaches. The method does not compare a reference image and the test image pixel-by-pixel, it eliminates the need for the storage requirement, generation, registration, and the comparison of a reference image with the test image. Instead, the method compares a small list of predicted feature types and locations with a list of detected features. This method is a major improvement over design rule approaches because it can detect missing features and extraneous circuitization that looks like good features. Unlike most design rule approaches, this method is not limited to verifying just minimum conductor trace width and spacing; it also verifies pads, various trace connections, isolated blobs, holes, etc. Most of the false-alarms that can occur in design-rule approaches are overcome in this technique.

The method makes use of image-to-image transform operations like contraction, thinning, expansion, etc. The observation that the local geometric and global topological correctness of typical circuit features can be inferred from the correctness of skeletal versions of the circuit features in a test image, is used in the analysis of the printed circuit patterns. The method works as follows:

- transform the image to obtain skeletal image from which defects and good circuit features can easily be detected.
- compare the detected feature list with a design feature list generated from circuit design data.
- conflicting features imply defects.

The fact that the presence of 0-, 1-, T- and blob-joins is sufficient to infer the existence of typical defects. Figure 10(a) shows these joins: where an *n-join* is a nonzero element with n nonzero 8-neighbors ($0 \leq n \leq 8$); a *T-join* is a 3-join whose 8-neighbors are skeletal elements; a blob-join is a skeletal element with an 8-neighbor that is not a skeletal element. In the Figure 10(a), X is blob-join, s is a skeletal element (a nonzero element necessary to maintain the connectivity of its 8-neighbors), and b is a boundary element (a nonzero element with a zero 8-neighbor).

The method can be used in: verifying minimum conductor trace width and detecting open circuits, detecting excessive trace width, verifying minimum spacing and detecting short circuits, and verifying pad position, area, shape, and trace-to-pad connections.

Algorithm for verifying minimum conductor trace width (MCTW) The algorithm works on the binary version of the test image as follows:

- alternately 4- and 8-thin the binary image ($\frac{w}{2}$) times. Figure 10(c) depicts the result of applying this operation on the original PCB sub-pattern in Figure 10(b).
- 8-thin ($\frac{W}{2} - \frac{w}{2}$) times the image obtained in previous step. Figure 10(d) depicts the 8-thinned output of Figure 10(c).
- detect 1- and blob-joins in thinned image obtained in previous step.
- compare the detected features in previous step with design list:
  - if 1-joins is not in design list, this implies trace width violations. The square boxes in Figure 10(e) are 1-joins, which implies the presence of defects (open).
  - if 1- and blob-joins in design list are not in detected features, then the image is missing these features.

Where W is the nominal trace width and w is the minimum acceptable trace width, less than W. Each of the algorithms presented in the paper use a different thinning process such that a particular class induces a known corresponding class of skeletal features that can easily and reliably be detected.

5.2 Pattern Detection using Boundary Analysis

The inspection system proposed by Benhabib *et al* [19] uses a hybrid flaw-detection technique based on pattern-detection and boundary-analysis techniques. For conductor flaws, the boundary-analysis algorithm locates areas that could have potential flaws, these are marked as non-standard edges, which are analyzed by a pattern-detection system to measure conductor widths. Thus this technique significantly increases the speed of the pattern-detection algorithm by isolating the conductor measurements only to those locations that could be flaws. Similarly, a pattern-detection algorithm measures land-widths for hole flaws, after locating the hole centers using an image subtraction technique.

Flaw alnalysis for conductors involves:

(a) *edge detection*, where four edge-pixel templates, shown in Figure 11(a), are used to determine whether the pixels in a window belong to an edge of a conductor in the image.

(b) *non-standard edge pixel determination* where edge-pixels are classified as either standard or non-standard based on a set of horizontal, vertical and diagonal edge-templates, as shown in Figure 11(b). An edge-pixel that does not match any of the templates is considered to be a potential flaw location, hence marked as non-standard.

(c) *edge-normal determination*, where three different operators (T, Y, I), shown in Figure 11(c), are used to determine the edge-normals of non-standard

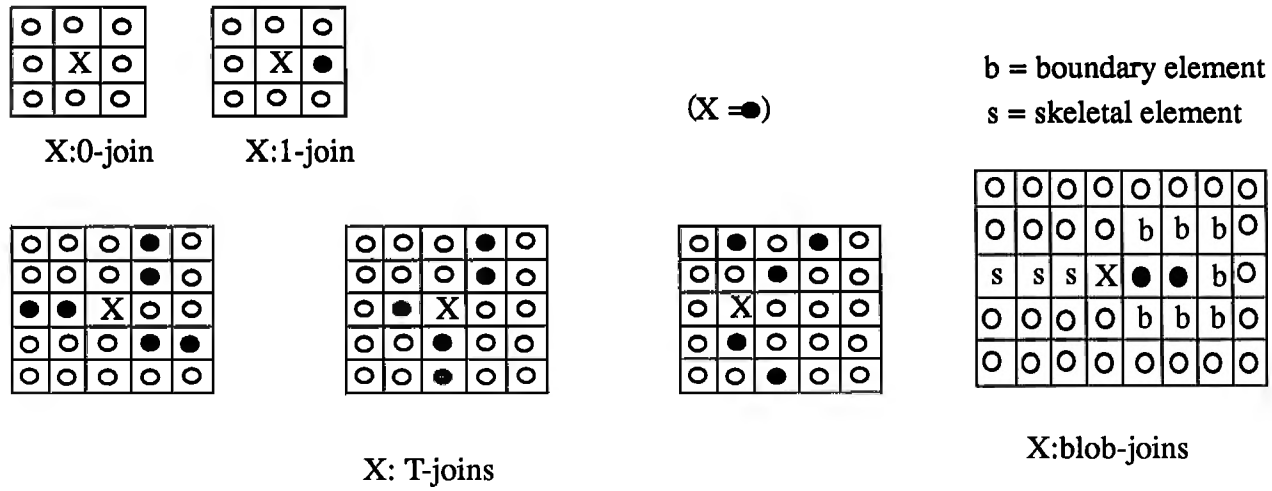


Figure 10(a): n-joins; T-joins and a blob-join

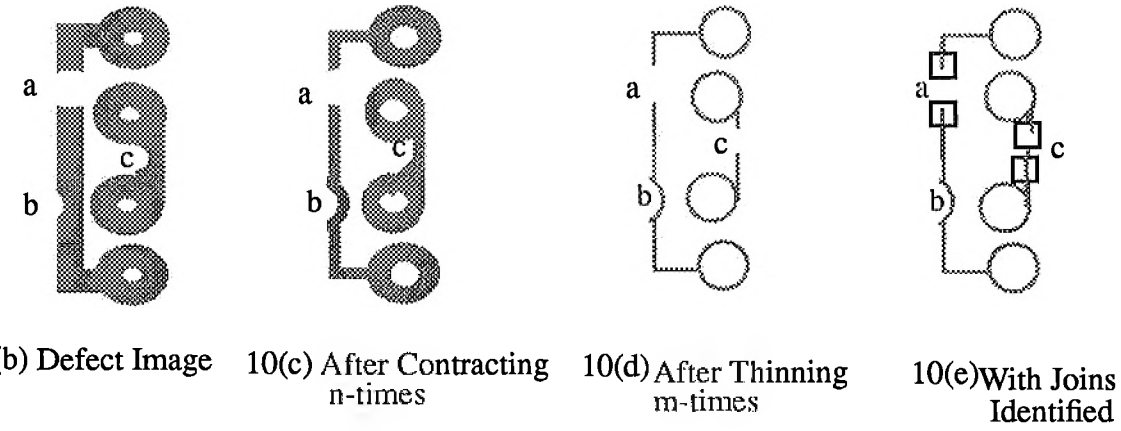


Figure 10:Verification of minimum conductor trace width

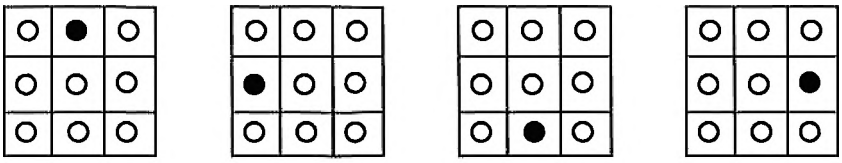
conductor edge-pixels. First the T-operator is applied and if each pixel under this operator is classified as substrate, then the edge-normal is in the direction indicated by the operator base. When this operator fails, usually at internal square corners of conductors, the Y-operator is next applied at these locations. When both operators fail, the I-operator is applied.

(d) *Flaw detection*, involves five different steps: (i) the non-standard edge-pixel and its counterpart on the opposite edge of the conductor are examined to determine whether they belong to a land or a conductor, (ii) the conductor width is compared with a specified minimum value to determine if there exists a flaw, (iii) the pin-hole size is compared, as a percentage with a specified maximum value to determine if there exists a flaw, (iv) the interconductor spacing is measured by counting substrate pixels in the opposite-normal direction until the first edge-pixel of the next conductor is located. This is compared with a minimum specified value to verify the existence of a flaw, and (v) a conductor-break-detection is performed by tracing from the current non-standard edge-pixel to the opposite edge-pixel along the edge of the conductor. If the trace succeeds within a specified number of edge-pixels, there exists a conductor break.

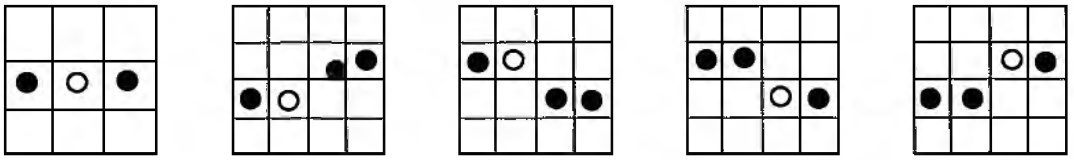
6 Summary

Back in the early 80's, machine vision was mostly smoke and mirrors - and a lot of credibility was lost because the technology was just not mature enough. Today, with the advances made over the last decade, we see machine vision answering the manufacturing industry's need to improve product quality and increase productivity. This study presented a survey of algorithms for visual inspection of printed circuit boards. The algorithms are not exhaustive in nature, but cover a broad variety of them conveying the main idea and the approach. A classification tree of the algorithms is presented. The classification divides the techniques into three basic classes: *reference comparison* in which production boards are compared with a database or golden board patterns, *design rule checking* provides for making measurements that are checked against predetermined quality rules, *hybrid techniques* combine both in selectively performing pattern matches as well as design rule measurements compared against a statistical model built from production board data. The major limitation of all the existing inspection systems is that all the algorithms need a special hardware platform in order to achieve the desired real-time speeds, which make the systems extremely expensive. Any improvements in speeding up the computation process algorithmically could reduce the cost of these systems drastically. However, they remain as a better option when deciding between increasingly error prone and slow manual inspection and higher productivity.

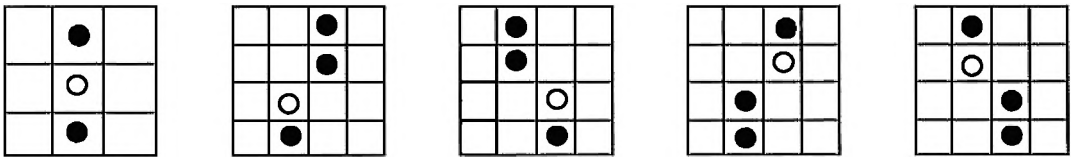
Acknowledgments - The authors would like to acknowledge Dr. Bruce McMillin, for his valuable suggestions in improving the quality of this report.



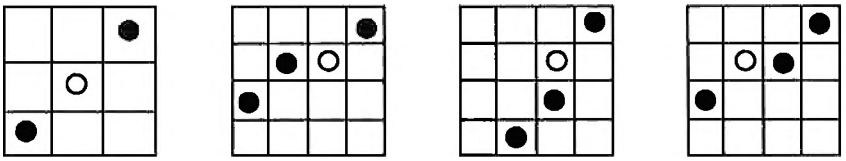
11(a)Edge Pixel Templates



Horizantal
Edge Templates

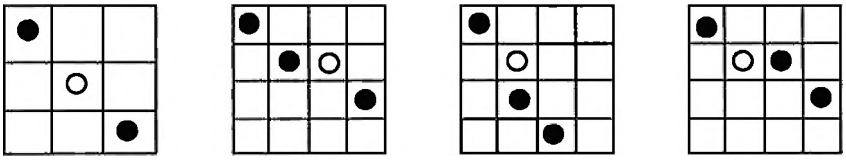


Vertical
Edge templates

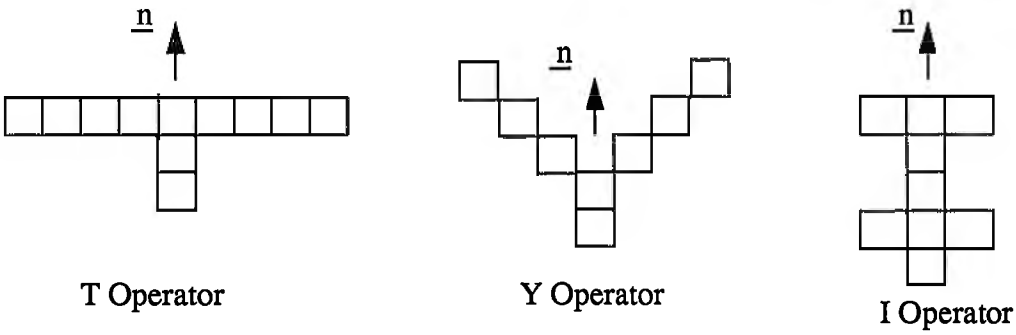


Diagonal

Edge Templates



11(b) Edge Templates



T Operator

Y Operator

I Operator

11(c) Edge normal determination operators

References

[1] Nello Zuech, "Introductory Thoughts on Machine Vision/AOI Applications in the Electronic Industry" *Proceedings of NEPCON '92*, Vol. 2, pp. 443-444, 1992.

[2] Walter H. Schwartz, "Vision Systems for PC Board Inspection", *Assembly Engineering*, Vol. 29, No. 8, pp. 18-21, 1986.

[3] Stephen T. Barnard, "Automatic Visual Inspection of Printed Circuit Boards", *Advanced Systems for Manufacturing : Conference on Production Research and Technology*, pp. 423-429, 1985.

[4] Ryan Hendricks, "On-Line Inspection Enables 6 Sigma Quality", *Circuits Assembly*, Vol. 1, No. 3, pp. 24-27, Dec. 1990.

[5] Frank J. Langley, "Imaging Systems for PCB Inspection", *Circuits Manufacturing*, Vol. 25, No. 1, pp. 50-54, 1985.

[6] Shin Mukai, "PCB Continuous Line System Proceeds from Manufacturing to Inspection", *Journal of Electronic Engineering*, Vol. 29, No. 305, pp. 34-39, May 1992.

[7] Joseph W. Foster III, Paul M. Griffin, Sherri L. Messimer, and J. Rene Villalobos, "Automated Visual Inspection: A Tutorial", *Computers in Industrial Engineering*, Vol. 18, No. 4, pp. 493-504, 1990.

[8] Ronald T. Chin, "Automated Visual Inspection: A Survey", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-4, No. 6, pp. 557-573, Nov. 1982.

[9] Ronald T. Chin, "Survey: Automated Visual Inspection : 1981 to 1987", *Computer Vision, Graphics, and Image Processing*, Vol. 41, pp. 346-381, 1988.

[10] Robert Thibadeau, "Printed Circuit Board Inspection", *Technical Report: CMU-RI-TR-81-8*, Carnegie-Mellon University, 1981.

[11] A. J. E. Goodall, and E. K. LO, "A Review of Inspection Techniques Applicable to PCB Manufacturing and Assembly, particularly with respect to SMT", *Advanced Manufacturing Engineering*, Vol. 3, Jan. 1991.

[12] Michael L. Martel, "Automated Inspection Roundup", *Circuits Manufacturing*, Vol. 29, No. 7, pp. 24-eoa, July 1989.

[13] Gerald Jacob, "Advances in Board Inspection", *Evaluation Engineering*, pp. 126-133, Sept. 1992.

[14] Wesley Hall, "Postprocessing in the '90s", *Printed Circuit Design*, pp. 12-16, August 1991.

[15] Robert H. Thibadeau, "Automated Visual Inspection as Skilled Perception", *TECON '85*, pp. 5.1-5.19, 1985.

[16] Hisashi Tsunekawa, "Latest Image Evaluation Systems Aid Efforts for Product Quality", *Journal of Electronic Engineering*, Vol. 29, No. 306, pp. 72-77, June 1992.

[17] Ronald T. Chin, Charles A. Harlow, and Samuel J. Dwyer III, "Automatic Visual Inspection of Printed Circuit Boards", *SPIE: Image Understanding Systems and Industrial Applications*, Vol. 155, pp. 199-213, 1978.

[18] Stephen Page, "Designing PC Boards for Optical Inspection", *Proceedings of NEPCON '92*, Vol. 2, pp. 1184-1187, 1992.

[19] B. Benhabib, C. R. Charette, K. C. Smith, and A. M. Yip, "Automatic Visual Inspection of Printed Circuit Boards: An Experimental System", *International Journal of Robotics and Automation*, Vol. 5, No. 2, 1990.

[20] Joseph W. Foster III, Paul M. Griffin, and J. D. Korry, "Automatic Visual Inspection of Bare Printed Circuit Boards using Parallel Processor Hardware", *The International Journal of Advanced Manufacturing Technology*, Vol. 2, No. 2, pp. 69-74, 1987.

[21] Joseph W. Foster, and Paul M. Griffin, "Automated Visual Inspection: Quality Control Techniques for the Modern Manufacturing Environment", *IEE Integrated Systems Conference Proceedings*, pp. 135-140, 1987.

[22] O. Silven, T. Westman, S. Huotari and H. Hakalahti, "A Defect Analysis Method for Visual Inspection", *Proc. of 8th IEEE Int. Conf. of Pattern Recognition*, pp. 868-870, 1986.

[23] J. Jarvis, "A Method of Automating the Visual Inspection of Printed Wiring Boards", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-2, No. 1, pp. 77-82, 1980.

[24] E. Abbott, M. Hegyi, R. Kelley, D. McCubbrey, and C. Morningstar, "Computer Algorithms for Visually Inspecting Thick Film Circuits", *Proceedings of RI/SME Conf. on Applied Machine Vision*, Memphis, TN, Feb. 1983.

[25] M. Ejiri, T. Uno, M. Mese, and S. Ikeda, "A Process for Detecting Defects in Complicated Patterns", *Computer Graphics and Image Processing*, Vol. 2, pp. 326-339, 1973.

[26] Qin-Zhong Ye, and Per E. Danielson, "Inspection of Printed Circuit Boards by Connectivity Preserving Shrinking", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-10, No. 5, pp. 737-742, Sept. 1988.

[27] Ahmed M. Darwish, and Anil K. Jain, "A Rule Based Approach for Visual Pattern Inspection", *IEEE Transactions of Pattern Analysis and Machine Intelligence*, Vol. PAMI-10, No. 1, pp. 56-68, Jan. 1988.

[28] Yung-Nien Sun, and Ching-Tsorng Tsai, "A New Model-Based Approach for Industrial Visual Inspection", *Pattern Recognition*, Vol. 25, No. 11, pp. 1327-1336, 1992.

[29] Jon R. Mandeville, "Novel Method for Analysis of Printed Circuit Images", *IBM Journal of Research and Development*, Vol. 29, No. 1, pp. 73-87, Jan. 1985.

[30] Louisa Lam, Seong-Whan Lee, and Ching Y. Suen, "Thinning Methodologies - A Comprehensive Survey", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, No. 9, pp. 869-885, Sept. 1992.

[31] Edward R. Dougherty, *Introduction to Morphological Image processing*, Bellingham, Washington, SPIE Optical Engineering Press, 1992.

[32] J. Serra, *Image Analysis and Mathematical Morphology*, Academic Press, 1982.

[33] Yasuhiko Hara, Hideaki Doi, Koichi Karasaki, and Tadashi Iida, "A System for PCB Automated Inspection Using Fluorescent Light", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-10, No. 1, Jan. 1988.

[34] M. A. West, S. M. DeFoster, E. C. Baldwin, and R. A. Ziegler, "Computer-Controlled Optical Testing of High-Density PCB", *IBM Journal of Research and Development*, Vol. 27, No. 1, pp. 50-58, 1983.

[35] Yasuhiko Hara, Nobuyuki Akiyama, and Koichi Karasaki, "Automatic Inspection System for Printed Circuit Boards", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-5, No. 6, pp. 623-630, Nov. 1983.

[36] Yasuo Nakagawa, Yasuhiko Hara, and Masayuki Hashimoto, "Automatic Visual Inspection using Digital Image Processing", *Hitachi Review*, Vol. 34, No. 1, pp. 55-60, 1985.

[37] E. B. David Lees and Philip D. Henshaw, "Printed Circuit Board Inspection - A Novel Approach", *SPIE - Automated Inspection and Measurement*, Vol. 730, 1986.

[38] T. Pavlidis, "A Minimum Storage Boundary Tracing Algorithm and its Application to Automatic Inspection", *Princeton University, Tech. Report 222*, Dec. 1976.

[39] C. M. Bjorkulund and T. Pavlidis, "On the Automatic Inspection and Description of Printed Wiring Boards", *Proc. of Int. Conf. Cybern. Soc., Princeton, NJ*, pp. 690-693, 1977.

[40] Masayasu Ito, and Yasuhiro Nikaido, "Pattern Inspection of a Printed Circuit Board using Graph Information", *Transactions of IEE Japan*, Vol. 112-C, No. 2, pp. 102-111, 1992.

[41] Paul M. Griffin, J. Rene Villalobos, Joseph W. Foster III, and Sherri L. Messimer, "Automated Visual Inspection of Bare Printed Circuit Boards", *Computers and Industrial Engineering*, Vol. 18, No. 4, pp. 505-509, 1990.

[42] Joseph W. Foster III, Paul M. Griffin, Sherri L. Messimer, and J. Rene Villalobos, "Automated Visual Inspection of Bare Printed Circuit Boards", *Computers and Industrial Engineering*, Vol. 18, No. 4, pp. 493-504, 1990.

[43] Seyfullah Halit OGUZ, and Levent ONURAL, "An Automated System for Design-Rule-Based Visual Inspection of Printed Circuit Boards", *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, pp. 2696-2701, April 1991.

[44] G.A.W. West, L. Norton-Wayne, and W. J. Hill, "The Automatic Visual Inspection of Printed Circuit Boards", *Circuit World*, Vol. 8, No. 2, pp. 50-56, 1982.

[45] G. A. W. West, "A System for the Automatic Visual Inspection of Bare-Printed Circuit Boards", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-14, No. 5, pp. 767-773, Sept./Oct. 1984.

[46] H. Freeman, "Computer Processing of Line-Drawing Images", *Computing Surveys*, Vol. 6, No. 1, Mar. 1974.

[47] W. M. Sterling, "Automatic Non-Reference Inspection of Printed Wiring Boards", *Proc. IEEE Comput. Soc. Conf. Pattern Recognition and Image Processing*, pp. 93-100, Aug. 1979.

[48] W. M. Sterling, "Nonreference optical inspection of complex and repetitive patterns", *SPIE Techniques and Applications of Image Understanding*, Vol. 281, pp. 182-190, 1981.