2022

Printed Circuit Board Defect Detection Using Image Processing

Nandeera Kamalaja De Silva Munaweera Tanthirige
Minnesota State University, Mankato

Printed Circuit Board Defect Detection Using Image Processing

By

Nandeera Kamalaja De Silva Munaweera Tanthirige

A Thesis Submitted in Partial Fulfillment of the

Requirements for the Degree of

Manufacturing Engineering Technology

In

Master of Science in Manufacturing Engineering Technology
Minnesota State University, Mankato.

Minnesota State University, Mankato

Mankato, Minnesota

(July 2022)

Date – July 5th 2022

Printed Circuit Board Defect Detection Using Image Processing

Nandeera Kamalaja De Silva Munaweera Tanthirige

This thesis has been examined and approved by the following members of the student's committee.

____
Advisor Dr. Kuldeep Agarwal PhD, PMP

____
Committee Member Dr. Shaheen Ahmed PhD, PE, CPE, CQE

____
Committee Member

Acknowledgment

I would like to thank all people who have helped me to successfully complete this work.

The never ending guidance and support from my academic advisor Dr. Kuldeep Agarwal Ph.D, PMP, faculty Associate Professor Dr. Shaheen Ahmed PhD, PE CPE, CQE and faculty mentor Dr. Winston Sealy, CD. CTM. Ph.D helped me tremendously in completing this project and rising through failures to continue the task.

Furthermore, I would like to thank the Creation technologies company, namely Ferdinand Mabeya, for assisting in acquiring the circuit boards for system testing purposes. I would like to thank Allan Wodtke for assisting in building the test fixtures.

A special thanks go out to the Minnesota State University, Mankato Graduate Research Grant program as this project wouldn't have been possible without the funding provided from this program. I would like to thank Minnesota State University, Mankato for supporting me and other students in their research.

Contents

PRINTED CIRCUIT BOARD DEFECT DETECTION USING IMAGE PROCESSING

NANDEERA KAMALAJA DE SILVA MUNAWEERA TANTHIRIGE

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE IN MANUFACTURING ENGINEERING TECHNOLOGY

MINNESOTA STATE UNIVERSITY, MANKATO
MANKATO, MINNESOTA
[JULY 2022]

ABSTRACT

This study was conducted to verify the hypothesis of the possibility to design an Automated Defect Detection system at a budget using image processing software. Focusing mainly on simplicity of integration with the capability to inspect a high variation of PCB with less user input. Reference comparison method was utilized to construct the defect detection algorithm where a defect free reference PCB gets compared with an inspection image to identify defects and anomalies. The paper discusses the range of possible defects for inspection on non-assembled PCBs, suggests methods for image processing and presents a final inspection algorithm, including their testing. The defect detections system showed high accuracy in detecting defects under ideal testing conditions and was unreliable in detecting defects in real-life testing conditions. Even though the current system may be sufficient for an experimental prototype system more improvements need to be done to be used in the industry.

1.0 INTRODUCTION

1.1 Introduction to Printed Circuit Board

Printed circuit boards (PCB) acts as the linchpin for almost all of today's electronics, from operating simple equipment such as kid's toys to being the hosts of all computing electronics for national defense equipment the uses of PCB are endless. With so much use cases in critical equipment and operations, quality inspection in the PCB manufacturing industry is paramount in producing highly reliable components.

The PCB production process contains several steps: Stating with a Raw material preparation step, Exposure and development of conductor step, material removal step through chemical or mechanical processors, Layering step and Masking step for protection. Different manufacturers would have variations for this process, but the primary steps would remain the same. Furthermore, many manufacturers would have inspection steps in-between critical steps to eliminate nonconformities moving forward in production lines, a single undetected defect that passes through any of one of these steps could make an entire PCB obsolete. The following flowchart (figure 1) represents the PCB manufacturing process excluding, the design phase [1].

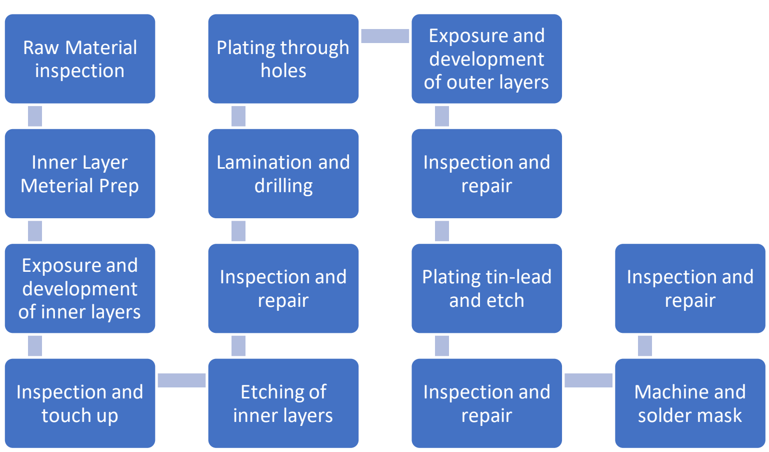

Figure 1 (Printed Circuit Board Manufacturing Flow Diagram)

1.2 Printed Circuit Board Defects

Imperfections on PCB can be categorized into two sectors, Cosmetic defects and Functional defects. It is critical to identify both of these deformities to produce a PCB with a 100% quality confirmation. Functional defects are considered to be fatal issues, which means the PCB does not attend the objective they are designed for; conductor breaking and short-circuit are some of the defects in this category. Cosmetic defects are imperfections that compromise the appearance of the PCB, for example, pinhole, breakout, over-etch, and under-etch. Cosmetic defects won't pose an immediate threat to the operation of the PCB but can jeopardize the performance of it in the long run due to abnormal heat dissipation and distribution of current.

Most of these defects or anomalies result from thermal expansion of artwork during printing, dust & dirt on board, air bubbles from electrolysis, and incorrect etching procedures. Thus, leaving unwanted conductive material or removing too much conductive material. Excessive conductive material results short, extra hole, protrusion, and island. And excessive removal of conductive material results in open, pinhole, nick (mouse bite), and thin patterns. Figure 2 depicts a PCB with a variety defect & Figure 3 depicts the same PCB with no defects.

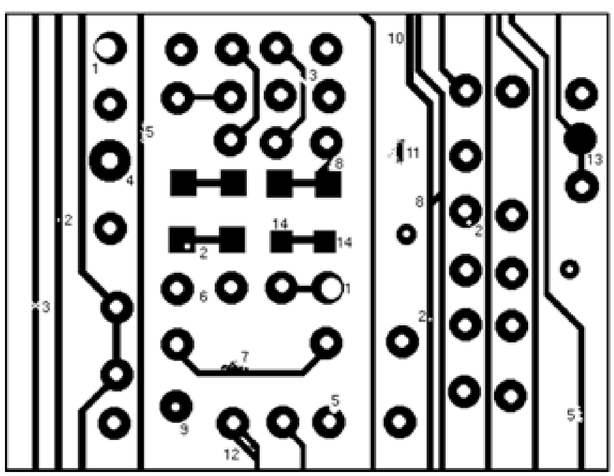

Figure 2(1. Breakouts 2. Pin hole 3. Open circuit 4. Under-etch 5. Mouse bite 6. Missing conductor 7. Spur 8. Short 9. Wrong size hole 10. Conductor too close 11. Spurious copper 12. Excessive short 13. Missing hole 14. Over-etch)

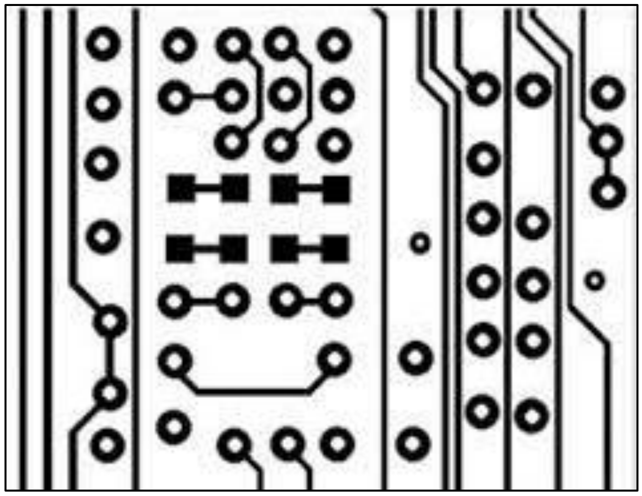

Figure 3 (PCB with No defects)

1.3 Printed Circuit Board Defect Detection Methodologies

There are two testing methods utilized to locate deformities on a PCB and they are 1. Electrical/contact and 2. Non-electrical/non-contact methods. The first method, electrical/contact is a reliable inspection strategy when it comes to testing design parameters that involves direct connectivity of a circuit. This method has its fair share of limitations. One of the major issues with electrical/contact methods is the test fixtures and setup procedures, with many new PCBs getting designed on grids that's smaller than 0.1 Inch's the testing fixtures becomes extremely complicated and expensive [2]. According to M. Moganti, C. Dagli, and S. Tsunekawa [1] even though electrical testing is less superior than visual testing in many ways it cannot be replaced. The second type of inspection method is Non-electrical/non-contact, human manual inspection and Automatic visual inspection (AVI) falls under this category [9]. With the use of an image acquisition system an image of a PCB would be captured and analyzed for defects.

Currently, large-scale PCB manufacturing companies utilize Automated Optical Inspection (AOI), In-Circuit Tests (ICT)/ Flying Probe Tester and, X-Rays (AXI) testing equipment to assist with the quality checking process [1]. The known downside of most of this commercial testing equipment would be the high price tag, making them unobtainable for small-scale manufacturers.

This becomes a great disadvantage when small-scale manufacturers try to expand their production and move from human manual inspection to automated product inspection methods. Both M. Moganti and Noor Khafifah Khalid echoed how important automatic visual inspection is in PCB manufacturing. They repeatably mention that moving to automatic visual inspection removes the subjective aspects of human manual inspection and provides fast and quantitative assessments and that it relieves the human operator from tedious, boring, and repetitive tasks of manual inspection. [2].

Computer vision technology and Image Processing can provide a solution to this problem. With the rise of the information technology era, most technologies that were expensive in the automatic visual inspection arena have become cheaper and more obtainable. Computer vision, image processing, deep learning, and open sources artificial intelligence technologies are readily available for users who show an interest in the subject. Many researchers have looked into ways to incorporate this computer vision technology to assist PCB manufacturers with automated quality inspection. During this project, an attempt would be made to design and develop a PCB inspection system utilizing readily available image processing software and inexpensive hardware to detect PCB defects.

2.0 LITERATURE REVIEW

A considerable amount of research has been conducted in the computer vision community to utilize image processing technology to detect defects in PCBs and the three main inspection methods more commonly used in achieving the desired results are the reference comparison approach, design rule checking (non-referential) approach, and the hybrid approach. The capabilities of these methods vary and in the following section, a brief introduction would be made about these three different approaches.

2.1 Printed Circuit Board Defect Detection using Reference Comparison Method

The reference comparison approach carries out a point-to-point (pixel-by-pixel) comparison with the inspection PCB image and the reference image to identify differences and locate areas of interest for further processing. One of the major advantages of the reference comparison method is that it is intuitive and easy to understand while having the capability of identifying almost all PCB defects. The referential method has a known set of drawbacks that needs to be addressed during implementation to avoid false defect detection, due to the differences between the inspection PCB images and templet images are sorted as defects any alignment inaccuracy or environment lighting differences may be identified as anomalies or a defect.

The focus of this study would be to utilize the referential comparison approach to detect defects of PCBs. As mentioned this approach is intuitive and less complex compared to the other approaches. In the following section, we'll be looking into some influential research projects that utilized the same approach to detect defects.

N. Khalid, [2] project titled "An Algorithm to Group Defects On Printed Circuit Board For Automated Visual Inspection" proposed an algorithm to recognize and classify bare PCB defects using Image Processing Toolbox, available in MATLAB 7.0 (R14SP1). The main methodologies that were used for defect detection were image difference technique and image subtraction. A reference image and the inspection image in the binary form would be passed down to the comparison step. The reason to have both images in the binary form would be to eliminate any effects of variation that might occur during the image preprocessing step, meaning that images would be compared in ideal environmental conditions. Using the mentioned image comparison technique, the PCB image would be compared pixel-by-pixel resulting in a sub-image containing defects. Further processing would be conducted using image adding, logical XOR, NOT, and flood fill operations to classify frequently found 14 PCB defects into five groups.

F. Raihan and W. Ce, [3] proposed a PCB defect detection system that used the same methodology "Image Subtraction" but had the capabilities to handle RGB (Red-Green-Blue) images. Ones a template image and the defective image in RGB format had been preloaded to the system it would conduct an image preprocessing step where the RBG images would be converted to a binary image using OpenCV's images processing techniques. The main reason for then binary conversion is that the algorithm is only equipped to handle 0 and 1. The two converted images would be compared using the Image Subtraction technique resulting in a new image that contains the defects of the PCBs' that's been inspected.

S. H. Indera Putera and Z. Ibrahim [4] in a later study made improvements to N. Khalid's initial defect detection system by introducing Mathematical morphology which segment the images into primitive patterns such as square-segment, hole-segment, thick-line segment, and thin-line segment. The template image and the defective image would have its own four segments. Each segment of the template image and the defective image would be compared with each other using the same image comparison algorithms N. Khalid proposed in his system to identify defects. This morphological technique increased the accuracy of the system. One of the main drawbacks both N. Khalid, [3], and S. H. Indera Putera and Z. Ibrahim [4] presented was that both systems were exclusively designed to handle binary images to reduce unwanted noise.

Up to this point, both F. Raihan and W. Ce, [3], and Indera Putera and Z. Ibrahim [4] systems were dealing with preloaded images. P. R. Masalkar [6] proposed a system that had the capability of capturing images in the RGB format and then perform defects detection. A captures image would be preprocessed to gain a more desired binary image. The binary images would be passed through an algorithm that combined F. Raihan's and Indera Putera's methods where the morphology technique separated square-segment, hole-segment, thick-line segment, and thin-line segments which would be individually compared with each other to detect PCB defects. The system was designed to detect and classify PCB defects.

The Images comparison operation has a few constraints that need to be addressed moving forward. Since the methodology, mainly realize making a pixel-by-pixel comparison any misaligned, resolution difference, image sizes difference, and lighting condition difference between the template image and the defective image can directly affect the accuracy and performance of the system [1,2,3,4,6]. To a certain extent, this issue was addressed by M. Baygin [7]. The following study proposed a system that could detect defects independently of alignment, but the defects that the system could detect are limited due to the algorithms been the studies utilized.

M. Baygin [7] proposed a system that uses feature extraction techniques such as Canny edge extraction to collect precise information about the edges in the templet image and, Hough Circle transformation to obtain information about the holes present in the templet PCB such as the count, the positions, and the diameters of each hole. This collected information from the templet image would then be used to feature match the inspection PCB images. Any difference between the data collected from the reference PCB and the inspection PCB would then be identified as a defect.

A similar approach was utilized by Kim, H.W., Yoo, S.I [11] for defect detection on non-repetitive pattern images. A modified corner detector was used for feature point extraction and defect were detected by finding minimum perfect matching of bipartite graph from a complete bipartite graph. According to the results both methods where less sensitive to alignment error and noise compared to pixel-based comparison technique, however M. Baygin [7] was only capable of detecting defects in holes and, Kim, H.W, Yoo, S.I [11] method was not equipped to locate defects but to merely identify a defective image.

Further improvements on the reference comparison method can be seen in V. Chaudhary [12], P. Wei, C. Liu, M [13,14] and hang, and, Y. Jin [15] work with the introduction of an image registration step. The feature points of the two images are extracted, matched and then a 2-D geometric transform will be estimated from matching points to transform inspection PCB image into the same orientation and position as the reference image, resulting in an accurate defect detection without having to consider misalignment, and image sizes difference.

2.2 Printed Circuit Board Defect Detection using Non-reference Method

The non-reference inspection approach also called the design-rule inspection technique, is a method that does not require any reference image or pattern to assist with defect detection. The methods operate by applying design-specifications knowledge to verify if the inspection PCBs are within the predefined design standards. For example, Min-Max trace width for different traces used, Min-Max circular pad diameter, Min-Max hole diameter, Min conductor trace, Min angular rings, trace termination rules, etc are some of the design character rules and feature dimensional tolerances that are used during this method [10]. Nevertheless, one of the more glaring issues with this method would be that any defects that do not violate the predefined design rules would pass through undetected. The following studies were conducted using non-referential methods.

J. F. Borba and J. Facon [16] (Add the reference at end) proposed a non-referential defect detection method using numerical rules with morphological operators to magnify the possible defects without altering the original image. The research succeeded in verifying vertical, horizontal, and 45 degrees-oriented traces. Later on, more developments in the image processing community lead to better performing techniques of defect detection techniques.

Du-Ming Tsai and Yan-Jheng Su [17] (Add the reference at end) introduced a non-referential, self-comparison machine vision scheme to detect defects on PCB substrate bond pads with rotated and deformed shapes. A Fourier shape reconstruction that is based on the contour of each individual bond pad was applied to detect local shape defects. Furthermore, discrimination features were extracted from the point-to-point distances between the original shape and the reconstructed shape, which then was used as quantitative measures to evaluate the anomalies on the contour. This method is best suited for anomalies in regular shapes such as circle, ellipse, rectangle & objects with simple geometric shapes. The performance of the system degrades as the object shape becomes highly irregular.

C. Benedek [18] (Add the reference at end) introduced a probabilistic approach for optical quality checking of solder pastes (SP) and detect a special soldering error, called scooping on PCB’s using a navel Hierarchical Marked Point Process (HMPP) framework. This method could handle the paste and scooping extraction problems simultaneously using unregistered images. The researcher conducted a quantitative and qualitative comparative evaluation between HMPP method and a morphological operations method and confirmed that the HMPP method is far more superior than the conventional morphological operations.

2.3 Printed Circuit Board Defect Detection using Hybrid Method

The hybrid inspection approach was aimed to increase the efficiency of defect detection by combining the strengths of both referential and non-referential methods and overcome any imperfections the systems might have individually. Furthermore, this method has the added advantage of detecting a higher number of defects compared to either method alone. The known downside of the hybrid method is the complexity in implementation. The following studies were conducted using the hybrid inspection method.

Kobayashi, H.H., Hara, Y., Doi, H., Takai, K., and Sumiya, A [19] introduced an optical system that was developed using a hybrid defect detection technique to inspect discriminated patterns such as copper, solder resists and silk-screen printings on PCBs'. the system utilized shape measurement and feature extraction for this process. Feature extraction, mainly aimed at detecting small defects, utilized shape deformation, corner shape, and isolated blob detection to defect detection such as scattered resist on a copper pattern, especially in SMT (surface-mount technology), devices such as soldering pad. The shape measurement method labeled the metal pattern and measures the shape features such as the center of mass (CM), size of the bounding rectangle, area, compactness, perimeter, and compared that data tables with the CAD data in order to evaluate the pattern defect. The system achieved a 100% defect detection rate with a very low false alarm rate of 0.06%.

Fikret Ercal, Filiz Bunyak. and Hao Feng [20] introduced a PCB inspection method that utilized RLE (run-length-encoding) for image representation and CAD artwork data to compare the test PCBs and filter non-defective areas on the test PCB image to reduce data storage and gain fast test results. Experimental results indicate that the method was very effective in locating functional defects and the researchers mentioned that none of the functional defects were missed during the test runs.

Chen, Tie, Zhang, Jianxin , Zhou, Youning and Murphey, Yi [21] proposed a PCB inspection system that consists of two modules, LIF (Learning Inspection Features) and OLI (On-Line Inspection) called Smart machine vision (SMV). The system took a similar approach to [20] where CAD data was utilized for the LIF phase to learn information about solder features and detect components, finding bounding boxes, and computing occupancy ratios. This information would then be used during the OLI phase to inspect solder paste, component placement, and post-reflow defects. The system was extensively tested, and the detection accuracy was above 97%.

Chang, P., Chen, L. and Fan, C [22] proposed an advanced PCB inspection system combining referential approach for case-based-reasoning (CRB) and rule-based approach to take the advantages and overcome the shortcomings of each approach. The system is comprised of two phases; phase 1 the pre-processing stage that the test PCB image is retrieved and segmented into basic pattern cases to compare with a concept space stored in a case base, and Phase 2 where the actual inspection/verification and online learning/training stage. According to the researcher the system can successfully identify open, short, indentation and particle.

In the present deep learning [23] and convolutional neural networks [24] are gaining more popularity in defect recognition and classification arena. Using a sample of segmented training images reference patterns would be stored to judge and test PCBs for defective and non-defective statues. These systems are much faster and more reliable compared to many of the traditional methods that we mentioned above.

Faghih-Roohi, S [25] and Soukup, D., & Huber-Mörk, R [26] introduced systems that were capable of inspecting defects on rail surfaces using deep learning of convolutional neural network (CNN). According to Faghih-Roohi, hours of automated video recordings of rail surfaces could be analyzed for defects using this system without relying on human manual inspection. Inspired from these research projects R. Ding, L. Dai, G. Li and H. Liu [27] proposed a Tiny Defect Detection Network (TDD-Net) that follows a Faster R-CNN [28] detection paradigm for PCB defect detection with three new changers. The resulting network could identify PCB defects such as missing holes, mouse bite, open circuit, short, spur at a mean Average, and Precision of 98.90%. which the researcher mentioned as a more superior performance compared with other state-of-the-arts defect detection networks.

Inconclusion it can be said that many researchers have identified successful methods to integrate computer vision to aid in PCB defect detection. During this proposed study I'll be focused on utilizing some of the above-mentioned methodologies to develop a functional, affordable, and accessible automated optical inspection system which could be beneficial either for small-sized companies, developers, or even enthusiasts and their home projects. Since most of the available AOI, ICT and, AXI systems are mainly focused on a large-sized serial production, the goal of this work is to use the exact opposite, with the technologies and components that are utilized to develop this system, any professional with basic knowledge of computer programming and hardware integration would be able to incorporate this system as a quality control measure for a fraction of the price of a commercial inspection system.

3.0 METHOD AND PROCEDURES

In this study, an Automated Optical Inspection was developed. The system consisted of a hardware portion and a software portion. The hardware portion includes the image capturing device and its support equipment and the lighting system. The software portion of the system includes image acquisition coding and image processing algorithms. In the following section, an overview of the system hardware and software would be given.

3.1 Image Processing Algorithm and Software

The image acquisition and processing portion of the system were designed using MATLABs' computer vision, and the image processing toolbox. The main goal of the proposed system was to identifying PCB cosmetic deformities, such as breakouts, mouse-bites, pinholes, open-circuits and, under/over etch. Figure 4 depicts the procedure map of the current PCB inspection system that was created using some of the methodologies that were utilized by researchers in the literature.

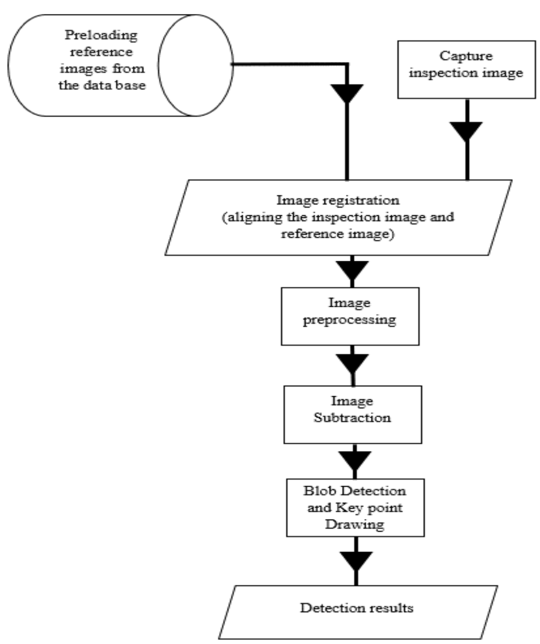

Figure 4 (PCB Defect Detection System Process Flow)

3.1.1 Reference image and Inspection Image aquation

The first step in the process is to preload a reference image (AKA templet image) to the system. This can be done by capturing the image during inspection or by uploading an image that was saved on the computer. The reference image would be an image of a defect free PCB that the system would compare the inspection images against. Caution needs to taken when acquiring images for detection as the differences in lighting conditions can lead to deferent contrast values which would result in false failures. The following pictures Figure 5 display the contrast differences according to different light conditions in the system.

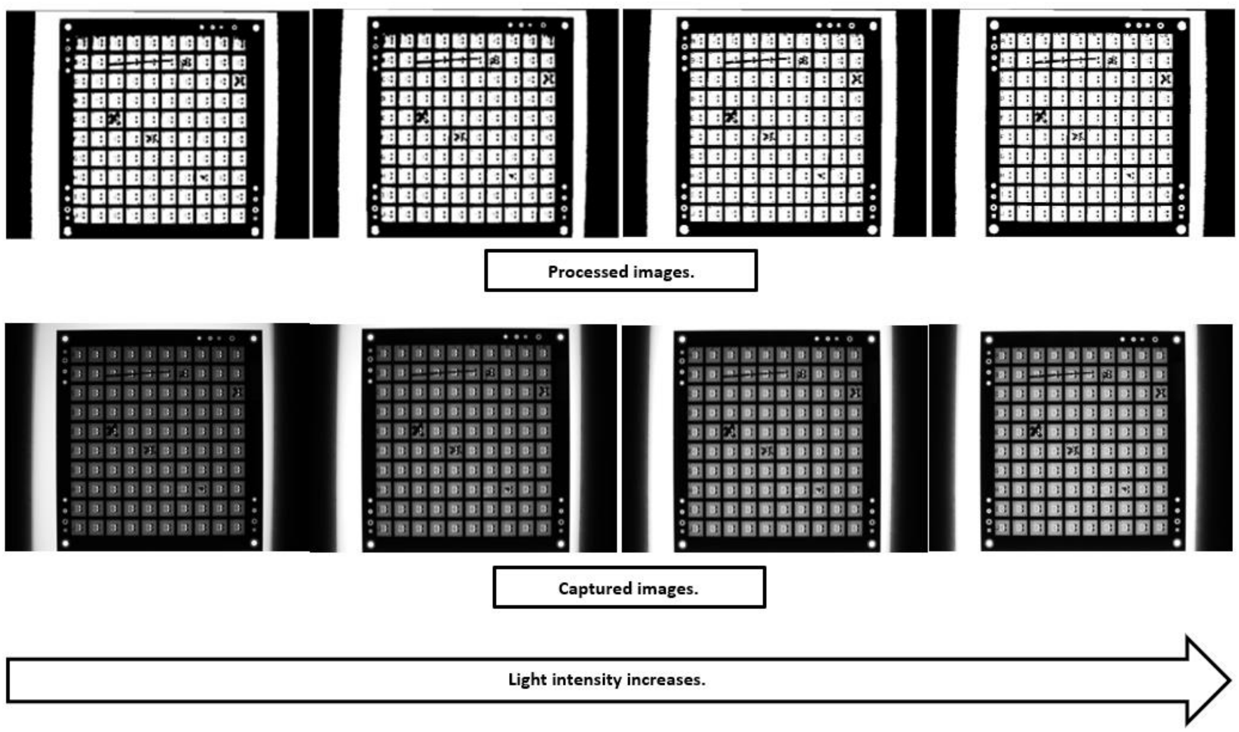

Figure 5 (Contrast differences according to different light conditions)

Once the templet image is preloaded to the system the rest of the process would require minimal operator input.

During the same stage of the process, the automatic image acquisition process would occur. The MATLAB software allows for different image acquisition methods, the softwares' built-in "image acquisition toolbox" allows for interactive image detection and hardware property configuration, furthermore, it generates equivalent MATLAB code to automate the image acquisition process. However, to keep the programing code simple, MATLABs' generate snapshots function was utilized to acquire and save an image to the computer memory.

3.1.2 Image Registration

Once both the images are in place the third step would commence. The image registration step is used to align the inspection image with the reference image. There are several image registration methods available in MATLAB: Feature-based registration, intensity-based registration, and Non-rigid registration. Each method has its pros and cons, for the following system, a feature-based image registration method called Speeded-Up Robust Features (SURF) was utilized because the local structure information of PCBs is more significant than information that can be obtained from image intensity or any other registration method.

The feature-based method operates by processing the image to extract anatomical structures from images such as points, curves, or edges, furthermore, this method can handle complex between-image distortions and can be faster because it doesn't evaluate matching criterion on every single voxel in the image. This data gets saved in a matrix (Geometric Transformation Matrix). The following picture Figure 6 show the points which the system was identified as matching points.

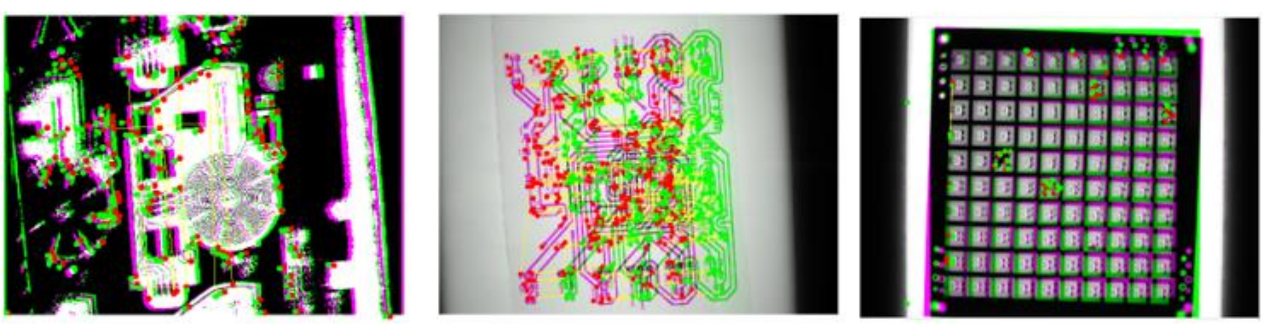

Figure 6 (Matching point of the reference image identified in red dots and matching points of the Inspection Image identified in green)

The data that was saved in the geometric transformation matrix can then be used to perform a global transformation on the inspection image to align the inspection image with the reference image. The following images Figure 7 are the aligned images from Figure 6 after the transformation.

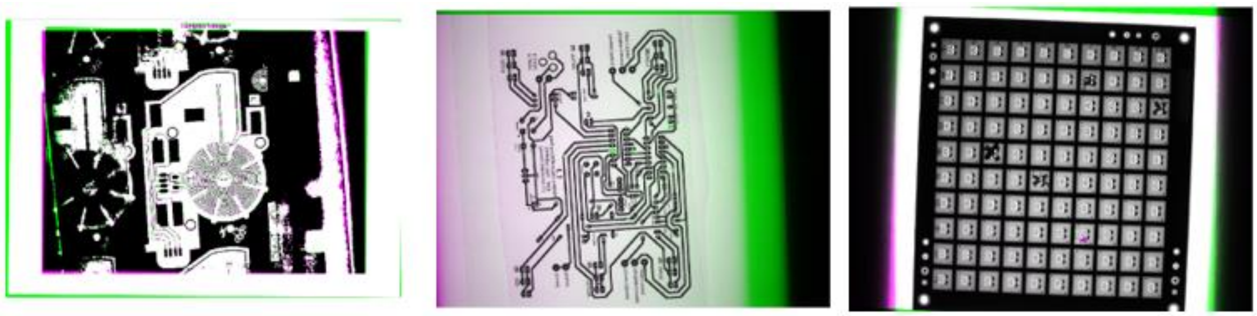

Figure 7 (Reference image and inspection image after completing the geometry transformation)

At the end of the registration step, both the inspection image and a reference image are similar in size, and alignment.

3.1.3 Image Pre-Processing

The next step of the process would be the preprocessing stage, where the images goes through a few image processing algorithms to gain a desirable binary image. The following section gives a brief description of the algorithms used in the image preprocessing stage.

Smoothening

The first step in the image pre-processing stage is smoothening the image this procedure is also referred to as blurring. Smoothening an image assist in reducing noise and camera artifacts in images. Gaussian blurring was utilized during this step. The Gaussian filtering is done by convolving each point in the input array with a Gaussian kernel and then summing to produce the output array. The calculation of the 2D convolution and can be described by the function.

$$f(i,j) = \sum_{(x,y)} \sum_{\epsilon 0} h(i-x, j-y) g(x,y),$$

where the pixels are weighted by the coefficient h, kernel and the Convolution kernel $h\ (x,\ y)$ is created according to the normalized Gaussian distribution formula.

$$h(x,y) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2},$$

Intensity adjustment

The second step in the image preprocessing stage would be the image intensity adjustment. The numerical data of an image can be observed to the histogram function on MATLAB. The *"x"* axis shows the pixel tonal variations from darkest (black or 0) to brightest (white or 255) and the *"y"* axis shows the number of pixels in under each tonal value. For many images the histogram would revile that the tonal variations tend to be focus on a relatively narrow spectrum. This can be adjusted using a MATLAB function. The following figures show an image and its tonal histogram before any adjustments were made.

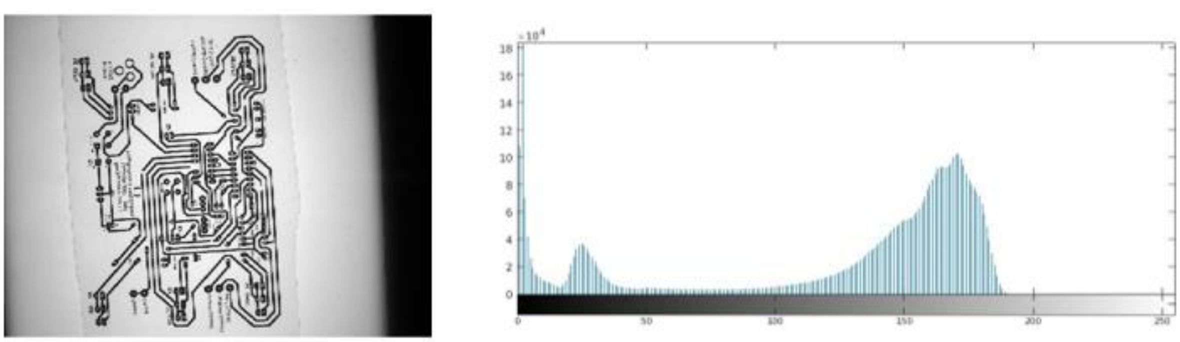

Figure 8 (Captured image from the image acquisition system on the left, Image tonal variation Histogram on the right; Before color intensity adjustments)

By running the image through a color intensity adjustment algorithm, a new image can be created which has a tonal frequency histogram that covers the whole pixel tonal variations. The algorithm saturates the bottom 1% and the top 1% of all pixel values, resulting an image that has better contrast and an image with better brightness distribution. This makes detail extraction easier on future steps. The following Figure 9 shows the earlier image after going through the color intensity adjustment.

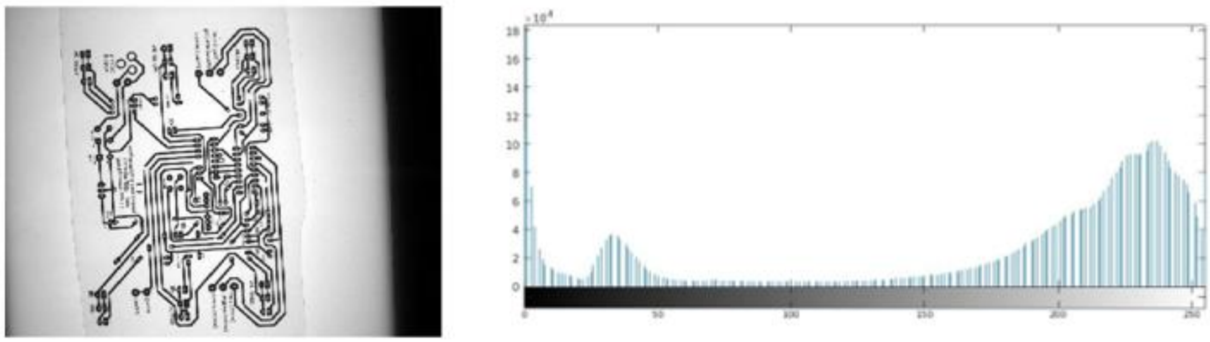

Figure 9 (Captured image from the image acquisition system on the left, Image tonal variation Histogram on the right; After color intensity adjustments)

Thresholding

During the next step of the image pre-processing stage, the grayscale image would be turned into a binary image. Using the threshold operation, a final decision would be made about the pixels in an image to categorically reject those pixels below or above some value while keeping the others. the resulting image would only contain pixels with vales of either 0 (black pixel) or 1 (white pixel). The following Figure 10 shows the results of the process.

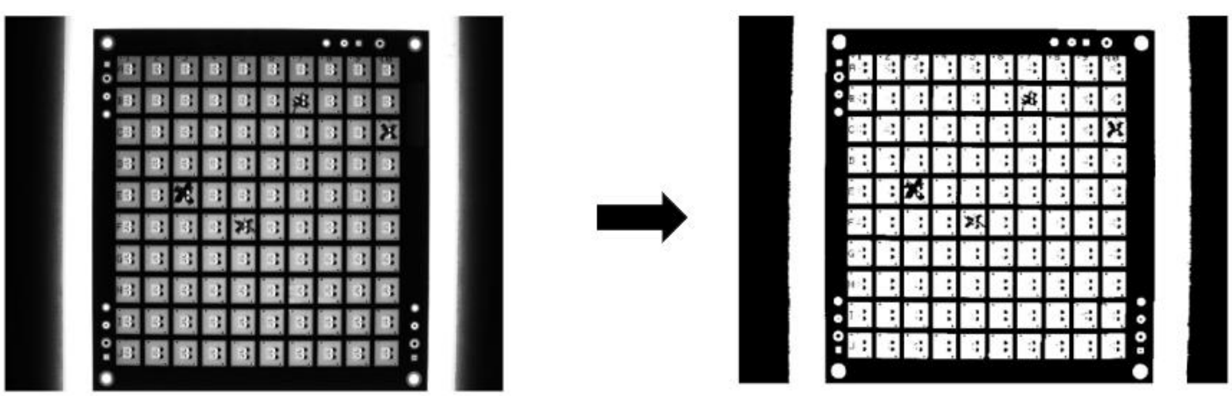

Figure 10 (Gray scale image on left, Binary image on the right after thresholding)

3.1.4 Image Subtraction

At the fifth step of the system, the preprocessed inspection and reference images would be compared using an absolute difference operation. A XOR operation would be carried out between the inspection image and the reference image resulting in a temporary image that contains anomaly and/or defect. A pixel-to-pixel comparison would take place, pixel in the reference image (x_r, y_r) with the pixel of the inspection image (x_i, y_i). The following Table 1 depicts the logical operation of the XOR function.

Pixel (x_r, y_r) Reference images	Pixel (x_i, y_i) Inspection image	Output
0	0	0
0	1	1
1	0	1
1	1	0

Table 1 (Logical XOR operation)

According to the logic,

If both pixels being compared are similar in value.
(x_r, y_r) = 0 (black) and (x_i, y_i) = 0 (black) or (x_r, y_r) = 1 (white) and (x_i, y_i) = 1 (white) the resulting output value would be a 0/black pixel. Which is not considered to be an anomaly or a defect.

If the pixels being compared are different in value.
(x_r, y_r) = 0 (black) and (x_i, y_i) = 1 (white) or (x_r, y_r) = 1 (white) and (x_i, y_i) = 0 (black) the resulting output value would be a 1/white pixel. Which is considered an anomaly or a defect.

The following graphic Figure 11 shows the process mentioned above. The reference image is “A” and the inspection image is “B”. The defects in the inspection image are highlighted by the red rectangles. The output image with the anomaly/defect are shown in image “C” and are highlighted by the red rectangles

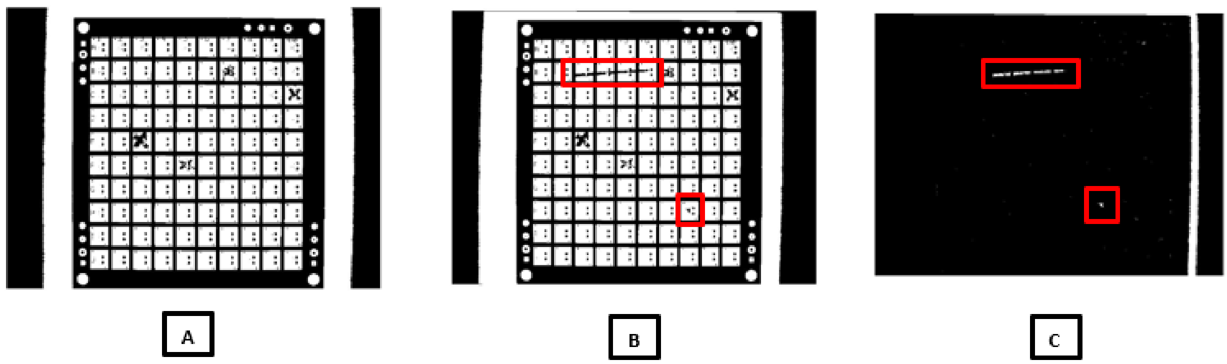

Figure 11 (Reference image "A", Inspection image "B", Output image of the XOR operation "C")

3.1.5 Blob Detection

The final step of the process is to portray the defects. The temporary image that is created from the previous step contains the differences between the reference image and the inspection image. This temporary image would then be passed through a technique called Blob Detection where an algorithm would identify and save the image coordinate values and size values of regions that differ in properties, such as brightness or color, compared to surrounding regions in a given image. When image “C” from Figure 11 passes through this algorithm the 1/white pixels gets picked up as regions of interest. The data that gets collected from the blob detection step would be passed to the Insert shape function in MATLAB, this function would create rectangles when the start point, height and width is given. These rectangles would pinpoint the anomaly/defect. The following Figure 12 showcases the final results of the system.

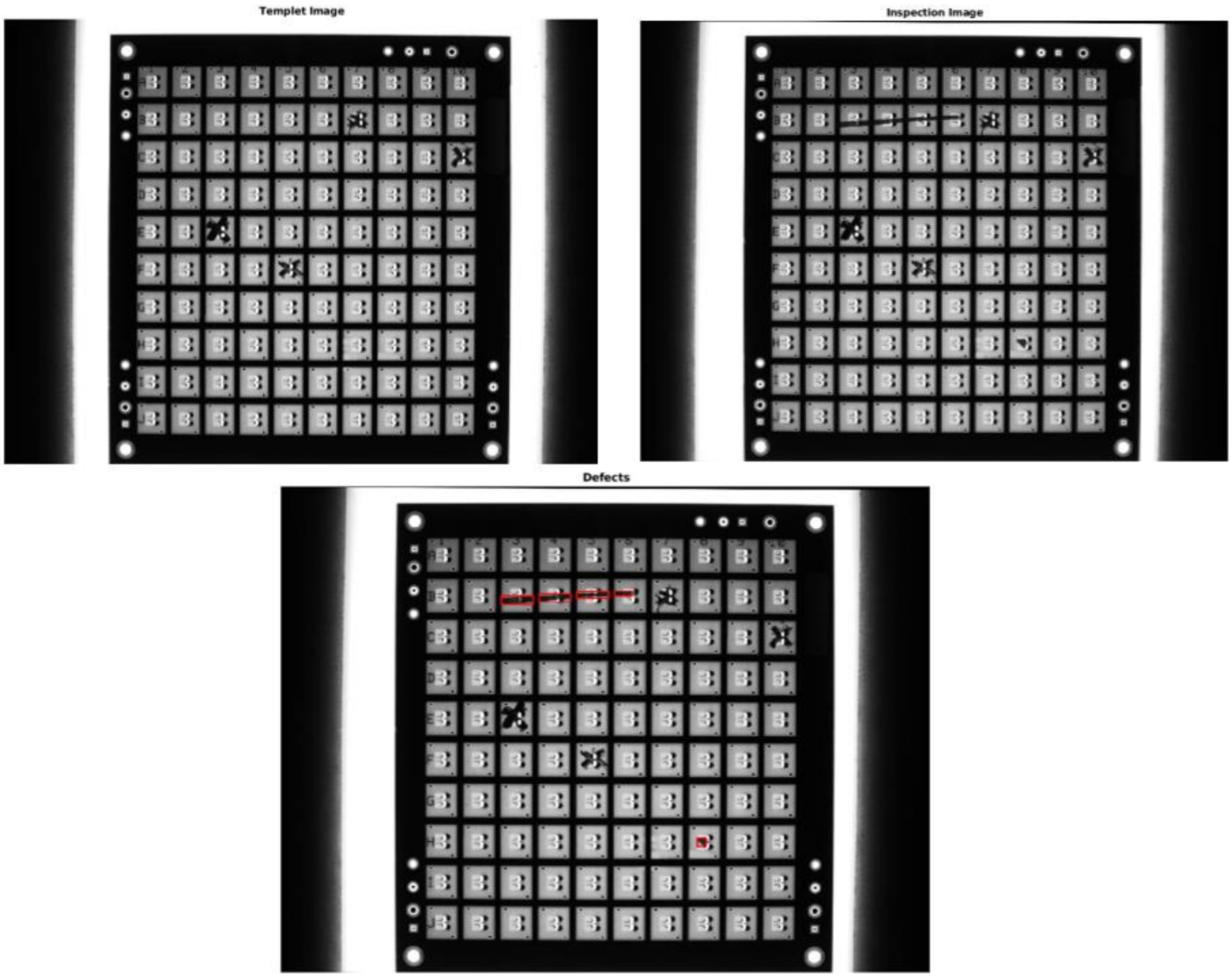

Figure 12 (Results from the defect detection system)

3.2 Hardware

Three major hardware components make up the PCB defect detection system (inspection system); the image acquisition system, the central processing unit, and the illumination system. The following hardware components were considered with potential further improvement in mind.

3.2.1 Image acquisition system

Camara

The inspection system uses an Allied vision color industrial camera module to capture images. The camera module is equipped with an ON Semi AR0521 CMOS sensor that enables high-quality imaging at 5.1 megapixels at 67 frames per second. The sensor size of the module is Type 1/2.5 with a resolution of 2592 (H) × 1944 (V) and a pixel size of 2.2 µm × 2.2 µm.

When choosing a camera, the following factors had to be considered to gain the best results. The field of view and the smallest feature that needs to be detected. The field of view (FOV) is defined as the area under inspection that the camera needs to acquire, and the width or the length of the PCB would be considered as the FOV. Furthermore, to make an accurate measurement on the image, you need to use at least two pixels per smallest feature that you want to detect. With the mentioned information in mind the following equation can be used to calculate the minimum sensor resolution required for the system. The equation represents the relationship of the camera resolution with respect to the field of view and smallest feature, and the graphic in Figure 13 these dimensions.

$$Sensor\ Resolution = Image\ Resolution = 2\ \times \frac{Field\ of\ View\ (FOV)}{Smallest\ Feature}$$

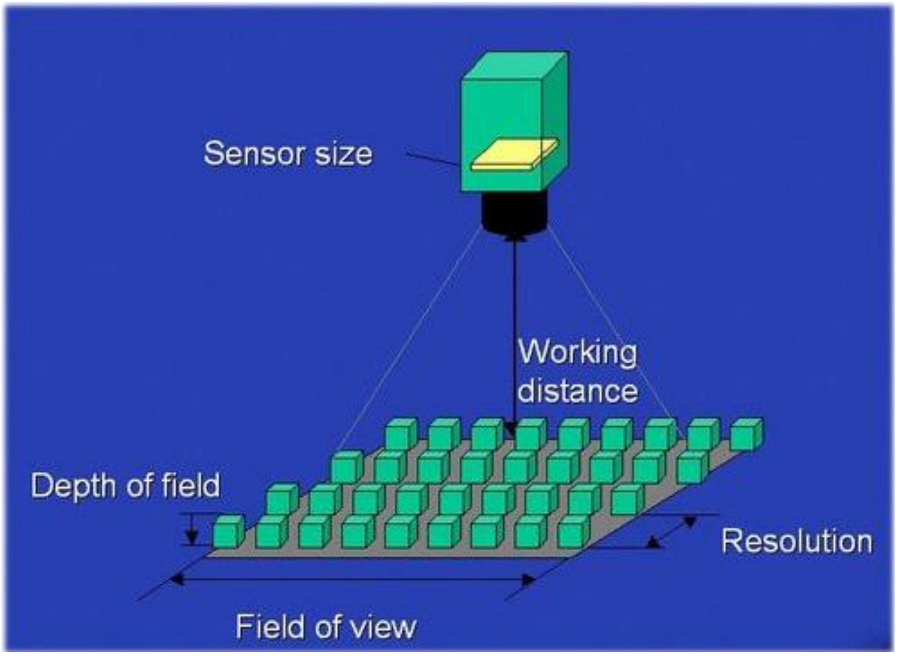

Figure 13 (Image acquired from nation instruments website explaining the Camera resolution calculation dimensions)

In order to keep the defect detection system within the budget the maximum field of view was determined to be 150mm, and the smallest feature to be detected was determined to be 0.5mm (500um), thus a camera with the minimum dimensional resolution (horizontal or vertical resolution) of more than 600 pixels would be adequate for the system. The camera that is used for the system has a resolution of 2592 pixels (Horizontal) × 1944 pixels (Vertical) exceeding the minimum requirements. furthermore, it offers two monochrome pixel formats, three YUV color pixel formats, and five RGB color pixel formats. These different image formats would allow for future experimentations of the system to yield better quality defect detection.

Lens

The second important component in the image acquisition system is the lens. Lenses are manufactured with a limited number of standard focal lengths and the lower the focal length the more the image gets distorted, the most common focal lengths include 6 mm, 8 mm, 12.5 mm, 25 mm, and 50 mm. Considering the sensor size, the field of view, and the working distance, a lens with a focal length of 25mm was selected. The following equation was utilized for the lens selection.

$$Focal\ Length = \frac{Sensor\ Horizontal\ Hight\ \times Working\ Distance}{Horizontal\ Field\ of\ view}$$

The horizontal high of a 1/2.5" sensor is 5.8mm and with an expected working distance of 600 mm and a maximum horizontal field of view of 150mm, the calculation would yield a focal length of 23.2 mm. The closest focal length on the market that would adhere with the systems requirements is 25mm focal length lens.

3.2.2 Central processing unit

The current defect detection system uses an Intel Xeon CPU running a 64-bit Ubuntu operating system. The central processing unit has adequate processing power for the inspection system to run smoothly without any issues. For future iteration of the defect detection system, an Nvidia Jetson TX2 developer board has been considered, the embedded computing board offers high processing power in a small package allowing the inspection system to be compact.

3.2.3 Illumination system

The final hardware component of the defect detection system is the illumination system, according to many researchers, this is one of the major components that get overlooked during inspection system designs. Having a stable illumination system could greatly reduce the amount of preprocessing that needs to be done to gain a desirable image. There are a few factors that affect the stability of an illumination system for an Automated optical inspection system; intensity, uniformity, directionally and spectral profile. Considering these factors before deciding on an illumination system can increase the contrast of the images that are being captured and reduce unwanted image preprocessing steps.

The current iteration of the defect detection system uses a Brightfield Backlight. This type of illumination generates instant contrast as it creates dark silhouettes against a bright background.

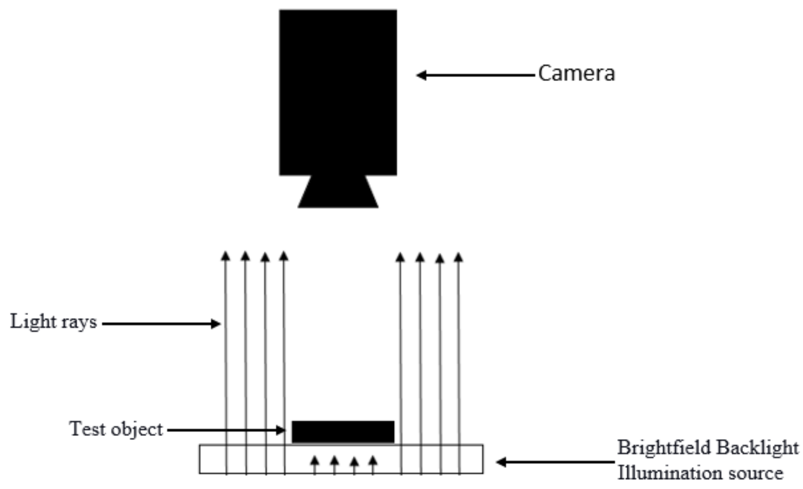

Figure 14 (Illumination system)

The following picture Figure 14 represents how the light rays emitted through the light source get blocked by the object and how the remaining lighting gets entered into the vision camera. The type of PCB that was intended to get tested using the system was the main reason to choose this lighting system. The focus of the system was to test Flexible PCBs and inner layers of multilayer PCB. The copper traces of the PCB block the light while the transparent substrate material of the PCB allows a portion of the light to pass through. This lighting geometry allows the highest contrast for this use case.

A backlight illumination system was built in-house using evenly spaced LED strips and a light diffuser. During the research into illumination system, it was made clear that LEDs are more suitable for lighting system as the life expectancy, application flexibility, output stability, continuous operation, and output intensity outperform most other lighting sources such as Quartz

Halogen, Fluorescent, and Xenon Strobe. The following Figure 15 show cases the performances of different lighting sources.

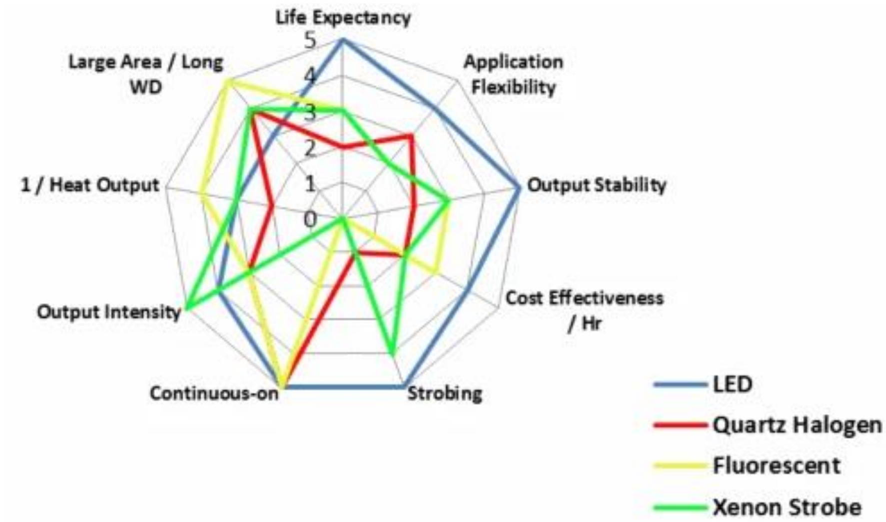

Figure 15 (Image obtained for National instruments' web page comparing common vision lighting sources)

The following Figure 16 showcases the current iteration of the PCB defect detection system. In order to isolate the system from uneven environmental lighting conditions, an enclosure was designed to prevent ambient light from entering the system and is shown in Figure 17.

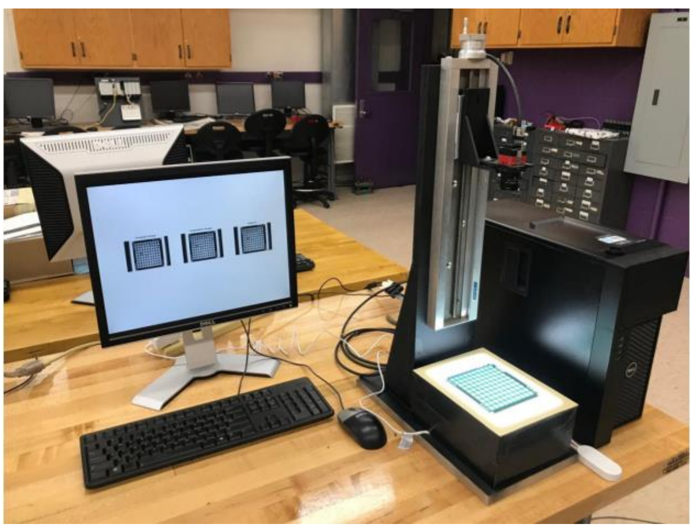

Figure 16 (The PCB defect detection system, Display to the left of the image, Backlight and the camera fixture at the center, and the central processing unit to the right)

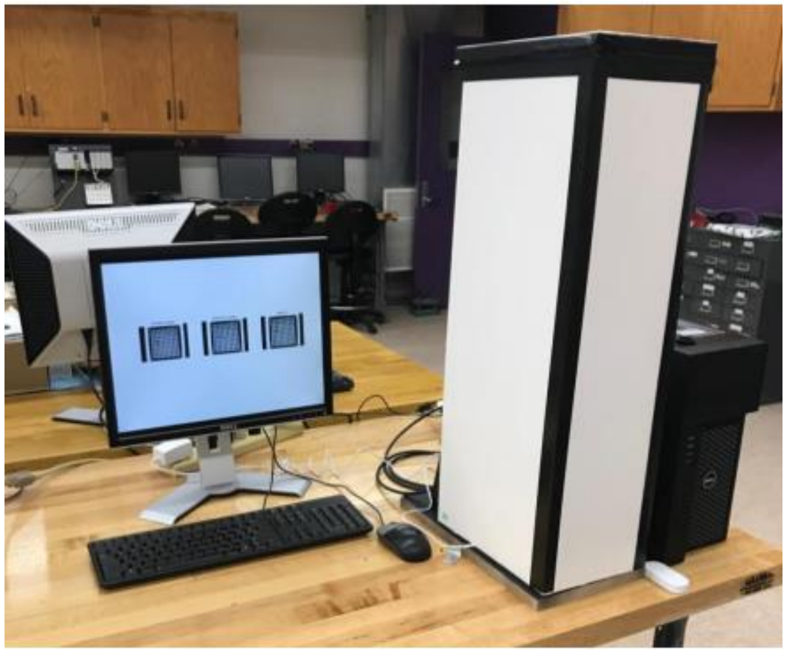

Figure 17 (The enclosure designed to isolate the system from ambient light)

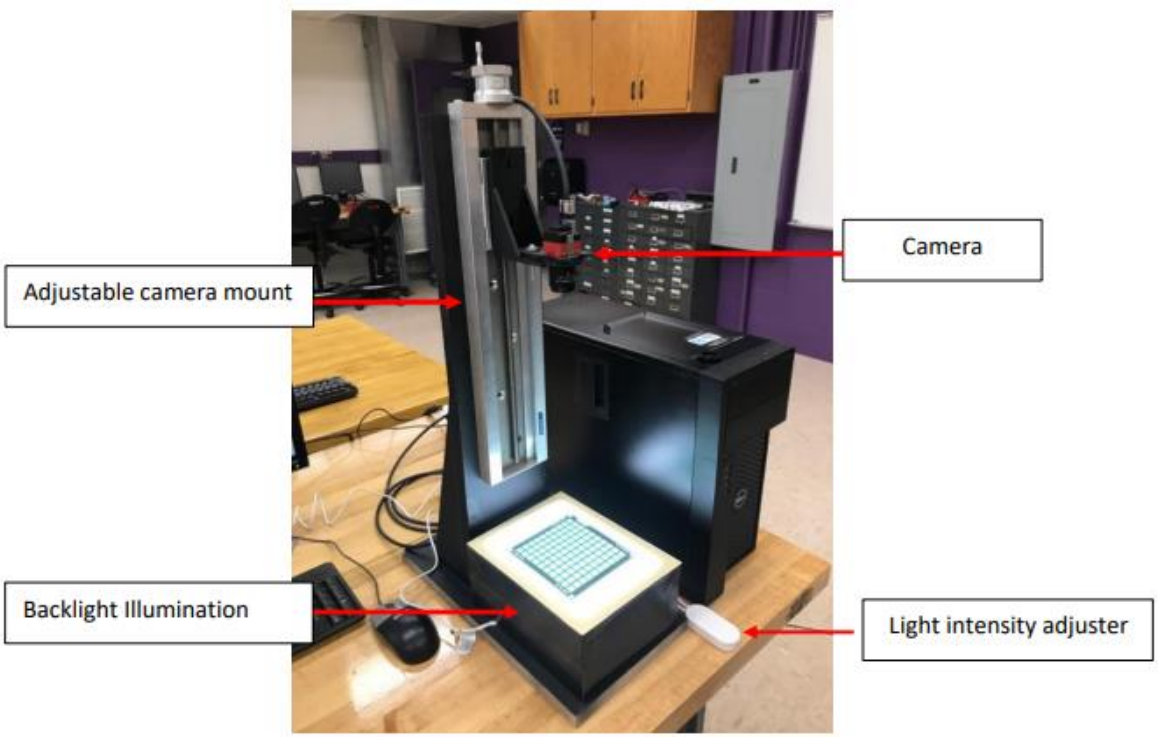

Figure 18 (Camera and light fixture component breakdown)

4.0 SYSTEM TEST AND RESULTS

The PCB defect detection system project was undertaken with the intent to build an inexpensive quality control tool to assist small-scale PCB manufacturers with low volume/ high mix production lines to relieve human operators from the tedious task of defect detection. With the use of a 5.1 Megapixel industrial camera, 25mm fixed focal length lens, brightfield backlight, and MATLABs' computer vision toolbox we were able to achieve this goal of designing an inexpensive PCB defect detection system. The system is still in its early stages where the full capabilities and limitations of the system have not been explored yet. The system has a working range of 152 mm to 406 mm with the ability to switch lenses (C-mount) depending on the test specimen. The lighting system has four adjustable lighting levels with the highest level of 6500K.

4.1 Defect detection algorithm test

The first test of the defect detection system was to identify if the system was cable of identifying defects on a PCB in ideal conditions. The ideal conditions in this case would be referred to having the reference image and inspection image in exact same lighting condition, same contrast level and same orientation. In order to perform the test, an image of a PCB patter was obtained by the internet and saved as the reference image, and copies of the same image was made, and on the copies of the image, artificial defects were made to mimic some of the common defects we encounter on PCBs and saved as the inspection images.

Seven copies of the original image were made, and the following defects were artificially created in each image using image editing software.

1. Under Etch
2. Wrong size hole & Missing hole

3. Over etch & Moues bite
4. Breakout & Pin holes
5. Shorts, Excessive Shorts, Spurious Copper & Spur
6. Missing conductor & Open circuit
7. Conductor too close

The following graphic Figure 19 depicts the original reference image and one of the artificially created defective PCB image.

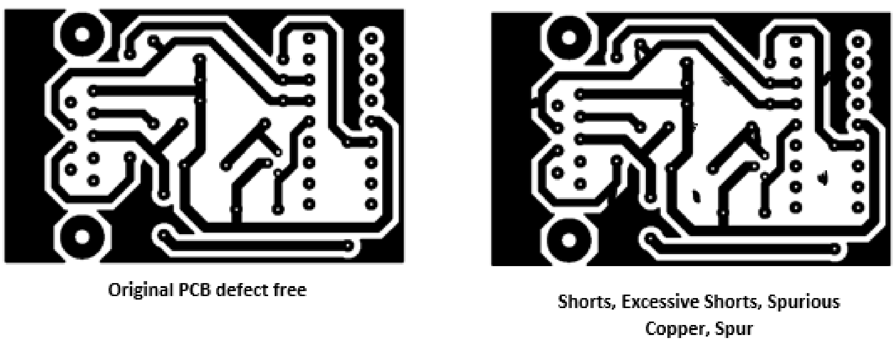

Figure 19 (Original PCB image on the left, Defective copy of the PCB in the right)

The following image Figure 20 is the results when the images from Figure 19 was passes through the system.

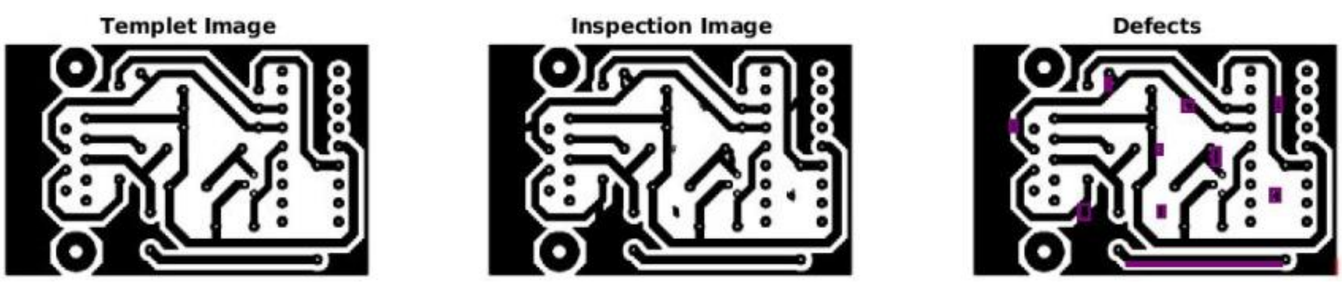

Figure 20 (Results from the defect detection of the Shorts, Excessive Shorts, Spurious Copper & Spur sample image)

All seven copies of the images were processed through the system to test if the defect detection algorithm was capable of detecting the defects. To quantify the success rate of the algorithm the following formular was utilized.

$$Success\ rate = \frac{True\ positive}{Errors\ detected} \times 100\ [\%]$$

True defect (TD) = actual number of defects/anomalies in the test image
True positive (TP) = number of defects/anomalies that was correctly detected by the system
False positive (FP) = number of incorrect defects/anomalies detection by the system.
Errors detected (ED) = the sum of both True positive (TP) and False positive (FP)

The following Table 2 lists the results from the test that was performed.

Defect group	TD	TP	FP	ED	Success rate
Under Etch	7	7	0	7	100%
Wrong size hole & Missing hole	7	7	2	9	77.7%
Over etch & Moues bite	11	11	2	13	84.6%
Breakout & Pin holes	12	12	0	12	100%
Shorts, Excessive Shorts, Spurious Copper & Spur	9	9	1	10	90%
Missing conductor & Open circuit	9	9	1	10	90%
Conductor too close	3	3	0	3	100%

Table 2 (Success rate of defect detection algorithm)

Three of the seven images that was tested successfully detected 100% of the defects/anomalies with 0 false positives. The lowest successes rate of 77.7% was recorded from the Wrong size hole & Missing hole test specimen where the system detected 2 locations of the PCB as defects/anomalies when they were actually not. Even though the algorithm had false positives it was capable of detecting all the true defects (TD) of the test specimens.

The following figures shows some of the false positives that was recorded during the test. Figure 21 is the image copy that had Wrong size hole & Missing hole. The system was able to detect all the defects that was present in the specimen, but some undesirable false positive was detected during the test as well. The red arrows point to the locations of the false failure.

Figure 21 (Undesirable false positive e on the Wrong size & hole Missing hole image copy)

Figure 22 show the image copy that had the Over etch & Moues bite defects mimicked. Once again, the system was able to identify the all the defects that was present in the image copy, but due to an alignment issue some undesirable false positive was detected.

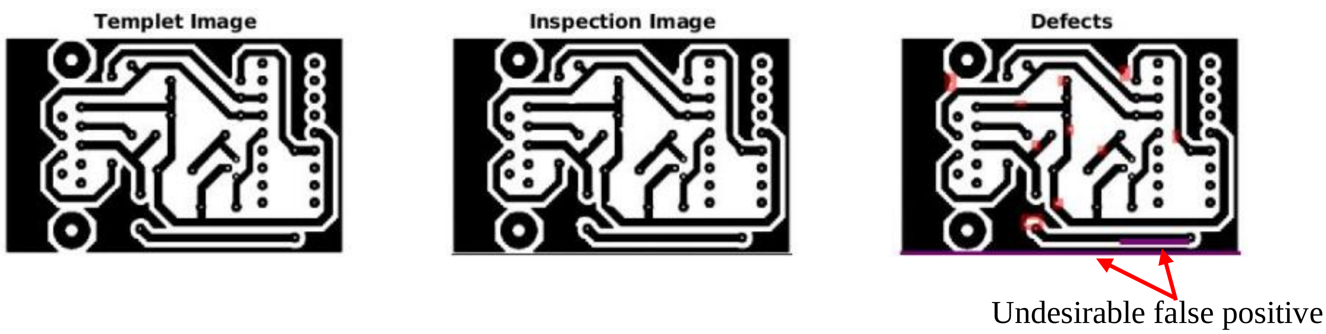

Figure 22 (Undesirable false positive on the Over etch & Mouse bite image copy)

Overall, the system was able to detect all the defects that was in the image copies with the mimicked defects showing that the system was capable of detecting defects in ideal conditions where reference image and inspection image are in exact same lighting condition, same contrast level and same orientation.

4.2 Illumination system test

The second test on the defect detection system was to identify how the system would perform under the different illumination levels. A proper registration of an image could show how the system preforms in each lighting level.

An image of a test specimen was captured in each lighting level (*1,2,3&4*) and was saved in the computer memory to be used as the reference image during the test. Then the same test specimen would be placed in 10 random orientations in the same lighting level and check the test results for any defects/anomalies. If the system was able to perform an accurate registration at that lighting level the result should not highlight any areas as defects/anomalies because the reference image and the inspection image are from the same test specimen. The test specimen that's been used for this testing was a 118 mm by 110 mm flexible PCB that has a 10×10 grid layout that goes from 1 to 10 from left to right, and A through J from top to bottom.

With the reference image loaded at each lighting level and the test specimen placed in random orientations the following results were gained.

At the first two lighting levels (*1 and 2*) or the lowest lighting levels, the system had a 100% false failure rate with defects/anomalies being highlighted on the results. On the 3rd lighting level, the system had a 30% failure rate, resulting only 3 false failures from 10 trials. The 4th or the heights lighting level had a 0% false failure rate, resulting in the best image registration out of all four lighting levels. The following Table 3 shows the results from the test that was performed.

Test number / Lighting level	1	2	3	4	5	6	7	8	9	10
1	Fail	Fail	Fail	Fail	Fail	Fail	Fail	Fail	Fail	Fail
2	Fail	Fail	Fail	Fail	Fail	Fail	Fail	Fail	Fail	Fail
3	Fail	Pass	Pass	Pass	Pass	Fail	Pass	Pass	Pass	Fail
4	Pass	Pass	Pass	Pass	Pass	Pass	Pass	Pass	Pass	Pass

Table 3(Illumination system test results)

4.3 Defect detection capabilities of the system

With these newfound test results from the Illumination system and the Defect detection algorithm test, the following test was performed to identify the capabilities of the defect detection system in real life conditions. Using an image of a defect free PCB that was captured from the defect detection systems' camara and comparing it with similar PCB with defects to find out how reliable the system is in locating defects were performed.

The same quantification method mentioned in defect detection algorithm *(section 4.1)* was utilized in this test, where the success rate gets calculated by dividing the "True positives (TP)" by the "Errors detected (ED)" and getting the parentage value of it.

Due to the limitation in test specimens the decision was made to print PCB patterns in transparency film, transparency films are more commonly used to print slides for overhead projectors. This allowed the opportunity to print conductor trace patters in a transparent background mimicking an inner layer of a PCB or a flexible PCB where the light would have less restrictions in passing through the subtract material. Furthermore, this method allowed to create artificial defects on a pattern and test it out on the system.

The same test samples that were used in the Defect detection algorithm test was used for this test. One image without defect was used as the reference image and seven copies of the same image with the deferent defect groups were printed on Transparency films. The following defects were artificially created in each image using image editing software.

1. Under Etch
2. Wrong size & hole Missing hole
3. Over etch & Moues bite

4. Breakout & Pin holes
5. Shorts, Excessive Shorts, Spurious Copper & Spur
6. Missing conductor & Open circuit
7. Conductor too close

The following graphic Figure 23 depicts the original reference image printed on a Transparency film captured by the system camara and one of the artificially created defective PCB image printed on a Transparency film captured by the system camara.

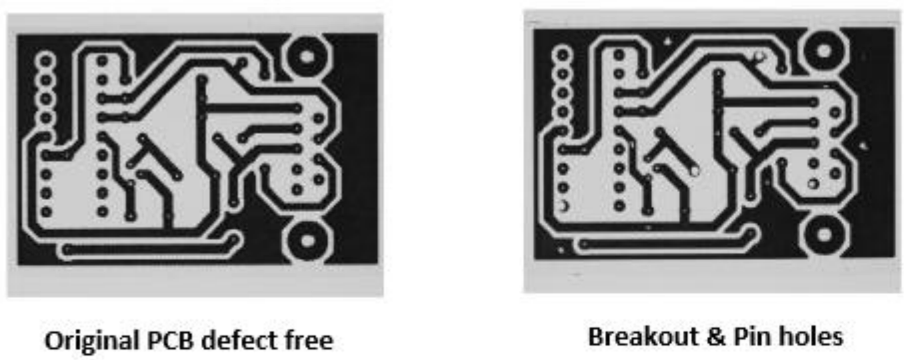

Figure 23 (Original defect free PCB on the left, defective PCB copy Breakout & Pinhole in the right)

All seven copies of the images were processed through the system to test if the defect detection system was capable of detecting defects/anomalies. The following image Figure 24 is the result when the images from Figure 23 passes through the system.

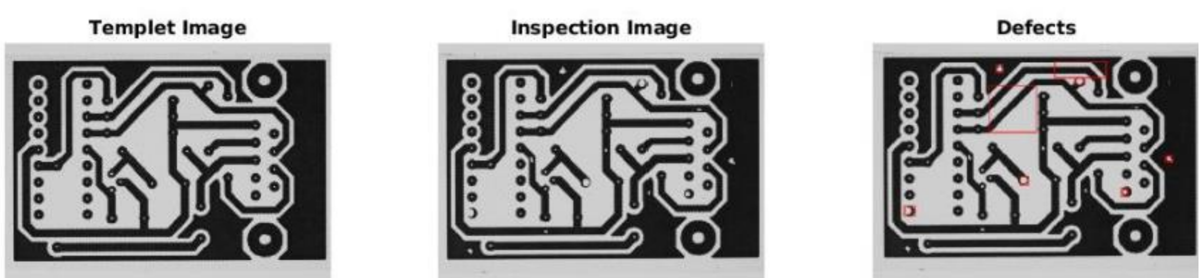

Figure 24 (Results from the defect detection of Breakout & Pin holes)

The following Table 3 lists the results from the test that was performed.

Defect group	TD	TP	FP	ED	Success rate
Under Etch	7	7	1	8	87.5%
Wrong size & hole Missing hole	7	7	13	20	35%
Over etch & Moues bite	11	10	8	18	55.5%
Breakout & Pin holes	12	6	2	8	75%
Shorts, Excessive Shorts, Spurious Copper & Spur	9	9	6	15	60%
Missing conductor & Open circuit	9	9	2	12	75%
Conductor too close	3	3	2	5	60%

Table 4 (Results of the defect detection system)

From the test results, it was made clear that the performance of the system has suffered a lot during the real-life conditions test. The maximum success rate was recorded to be 87.5% whit the system detecting all the defects and one false positive. The lowest success rate was 35% where the system identified more false positives than true positives and, in some cases, failed to identify all the true defects as well.

The following figures shows some of the results that was recorded during the test. Figure 25 is the specimen that had Missing conductors & Open circuits. The system was able to detect all the defects that was present in the specimen, but some undesirable false positive was detected during the test as well. The red arrows point to the locations of the false failure.

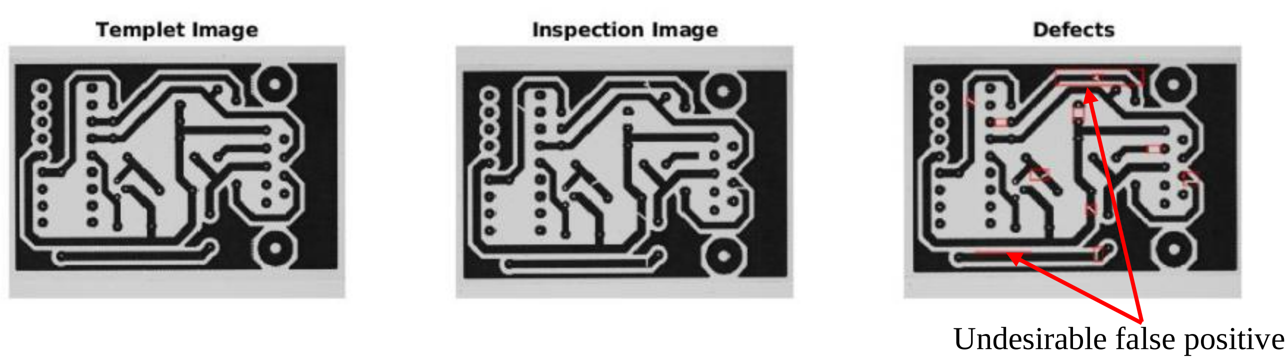

Figure 25 (Undesirable false positive on the Missing conductor & Open Circuits specimen)

Figure 26 show the test results from the Breakout & Pin holes test specimen. During this test, apart from having false positives the system was unable to detect all the true defects of the specimen, 6 true defects were missed out during the process. The red arrows show some of the false positives and blue arrows show some of the missed true defects.

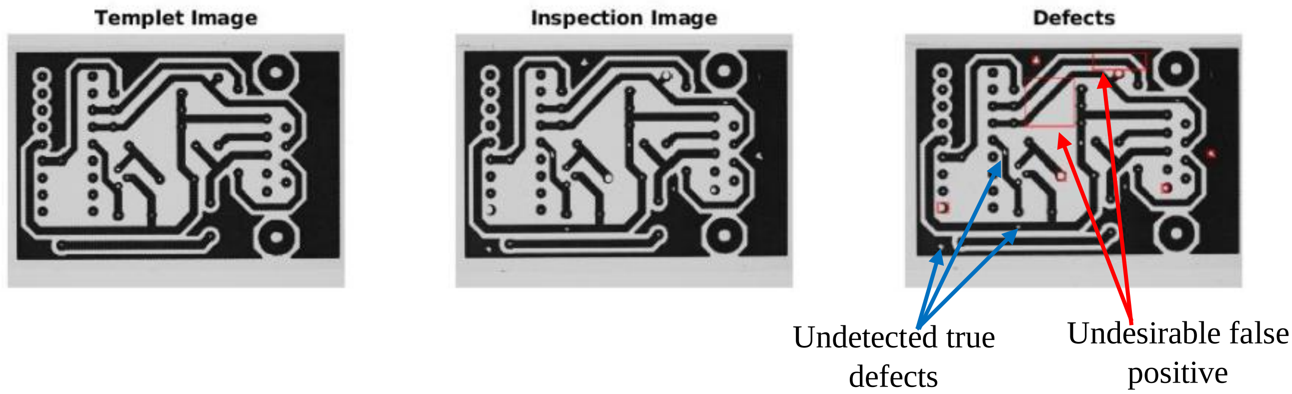

Figure 26 (Undesirable false positive and Undieted true defects on the Breakout & Pin holes test specimen)

Overall, the system at the real-life condition where the images were captured through the camara, the lighting was through the in house-built illumination system and the alignment of the inspection image was not a 100% to the reference image was able to perform the defect detection at a lower accuracy than it did during the ideal conditions. Five out of the seven specimens had all defect detected and two specimens “Breakout & Pin holes” and “Over etch & Moues bite” had defects that went undetected through the system.

5.0 DISCUSSION

There are several aspects of the system that needs to be discussed prior to reaching into a conclusion on the results of the PCB defect detection system. The main limiting factor of the system was the access to test samples. Attempts were made to acquire test samples from local PCB manufactures but due to perpetuity reasons the request were turned down. Due to the lack of test samples a more in-depth understanding about how the system might react to different martial were unable to be tested. In order to complete the testing phase of the system other means were used to create test specimens with similar properties to flexible PCBs and PCB inner layers. Nevertheless, it is still possible to draw conclusions from the results obtained and gain valuable conclusions about the functionality of the resulting system

5.1 Overall system performance

The PCB defect detection system was created using the reference comparison methodology, where a defect free reference PCB image gets compared with an inspection PCB image at a pixel level. Any deference between the pixels would be considered a defect or an anomaly. As mentioned in the literature review there are known drawbacks of the system that could produce false failures and during the testing of the system many of these issues were encountered.

During the test of the system under ideal conditions where the reference image and inspection image had exact same lighting condition, same contrast level and same orientation the detection of the true defects were at a 100%. The algorithm was capable of detecting all 14 defects and the highest false positive rate was at 22.2%. The reason for the high performance are the limitations of variables that could cause the images to be preprocessed differently.

When switching from an ideal testing condition to real-life testing condition the true capabilities of the image preprocessing stage gets tested. With the reference image and the inspection image getting captured through an industrial camara with the in-house-built illumination system, the possibilities of the lighting conditions varying depending on the placement of the specimen, the high noise level of the camara input, dust particles contaminating the test area, unaccounted anomalies in the test specimen can cause variation in the output of the preprocessed image. These mentioned reasons can cause false positive rates to be higher and undetected true defects to increase as well. The defect detection capability of the system was still at a desirable rate with 5 out of 7 specimens resulting true defect detection at a 100%. The remaining test specimens was not able to detect all defect resulting 9.09% and 50% of the true defects getting passed undetected. Furthermore, the real-life condition test resulted many false positives, and some were as high as 65%.

The defect detection system was tested in both ideal and real-life conditions to understand it's capabilities and limitations and it was evident that the reliability of the system gets affected significantly in real-life testing conditions, as mentioned above the image preprocessing algorithms been utilized can greatly affect the outcome of the defect detection system. For example, during the stages where the inspection image and reference image gets smoothened (blurred) and intensity adjusted in order to reduce undesirable image noise, details of defects can get eliminated, furthermore during the thresholding step even more details of the image can get distorted. These reasons can lead to defects getting passed undetected.

During testing it was further noticed that variable such as the threshold value, structuring elements size, or image registration method can behave differently with different test specimens being used.

6.0 CONCLUSION

With the improvements made in information technology and the accessibility to visual inspection technologies made easier, any PCB manufacturer or a hobbyist with robust algorithms and the right set of hardware can obtain a cheap and relatively reliable PCB defect detection system. During this study, the designing and development of a PCB defect detection system was undertaken with the intent to assist small-scale PCB manufacturers with low volume and high mix manufacturing systems to automate their PCB defect detection process. An algorithm was proposed using MATLABs' image processing toolbox that incorporated PCB defect detection using reference comparison approach. With the use of a 5.1 Megapixel industrial camera, 25mm fixed focal length lens, and an in-house-built brightfield backlight illumination system I was able to build a system that was capable of detecting cosmetic defects in PCBs.

The tested conducted during the system verification was helpful in identifying the ideal testing conditions for the system, during further studies a better understating of the system was obtained through identifying the limitation of it. The Current systems’ reliability weighs heavily on the quality of the images being captured, undesirable image noise, variations in the lighting intensity of the illumination system and the alignment differences of the image can cause the system to fail in detecting defects accurately.

The results gained through the testing of this system may be sufficient for an experimental prototype system, but in a real production environment the current iteration of the PCB defect detection system would not be usable. However, the following improvement can be suggested for future iterations of the system to obtained more reliable results.

To improve the performance of the PCB defect detection system, the future work has been suggested:

- Optimize image acquisition system to produce images with less noise
- Optimize the image preprocessing algorithm to operate in varying operating conditions
- Improve the illumination system to have more stable lighting intensity.

With the suggested improvement the accuracy of the system can be improved further, but even a system with a robust defect detection algorithm with all proper environmental conditions can still generate false alarms or let defects pass undetected. Only continues improvement and testing can creating a system that could match or outperform the current industrial automated optical inspection systems.

Appendix A:

PCB defect detection system MATLAB code.

```
%% Reading the templet image and the inspection image
% templet image = img_tmp
img_tep = imread('testog.bmp');

%% Acquiring the image from camera
% the inspection image = img_inp
vid = videoinput('gentl','1',"Mono8");
isrunning(vid);
triggerconfig(vid, 'immediate');
vid.FramesPerTrigger = 1;
frame = getsnapshot(vid);
image(frame);
imwrite(frame,'inp.bmp')
img_inp = imread('inp.bmp');

%% Aligning the inspection image with the template image
% tForm1PCB is the resulting structure and img_TFinp is registered image
[tForm1PCB] = registerImages(img_inp,img_tep);
img_inp = tForm1PCB.RegisteredImage;

%% Smoothing the images
img_tepm = imgaussfilt(img_tep);
img_TFinp = imgaussfilt(img_inp);

%% Adjusting image intensity values
img_tepm = imadjust(img_tepm);
```

```
img_TFinp = imadjust(img_TFinp);
%% Converting image to binary format
img_tepm = img_tepm > 60;
width = 2;
se = strel('square', width);
BW_tep = imopen(img_tepm, se);
img_TFinp = img_TFinp > 60;
BW_inp = imopen(img_TFinp, se);

%% Finding the difference
Disk_element = strel("disk",8); % introducing a disk element
Abb_diff = BW_tep - BW_inp;
Im_close = imclose(Abb_diff,Disk_element);
Im_close = im2bw(Im_close); % converting the results from Imclose to binary

%% Identifying anomalies in the absolute difference image
%Blob Analysis
Hblob = vision.BlobAnalysis('MinimumBlobArea',64,'MaximumBlobArea',5000);
[objArea,objCentroid,bboxOut] = step(Hblob,Im_close);
Ishape = insertShape (img_inp,'rectangle',bboxOut,'Color','r',"LineWidth",6); % drawing
bounding boxes
imshow(Im_close);

%% Displaying the results
subplot(3,1,1); imshow(img_tep)
title('Templet Image')
subplot(3,1,2); imshow(img_inp)
title('Inspection Image')
subplot(3,1,3); imshow(Ishape)
title('Inspection Image')
```

References

[1] Madhav Moganti, Fikret Ercal, Cihan H. Dagli, and Shou Tsunekawa. 1996. Automatic PCB Inspection Algorithms. Comput. Vis. Image Underst. 63, 2 (March 1996), 287–313. DOI:https://doi.org/10.1006/cviu.1996.0020

[2] N. Khalid, "An Image Processing Approach Towards Classification of Defect on Printed Circuit Board", Projek Sarjana Muda, Universiti Teknologi Malaysia, pp 87-91 2007

[3] F. Raihan and W. Ce, "PCB defect detection USING OPENCV with image subtraction method," 2017 International Conference on Information Management and Technology (ICIMTech), Yogyakarta, 2017, pp. 204-209.

[4] S. H. Indera Putera and Z. Ibrahim, "Printed circuit board defect detection using mathematical morphology and MATLAB image processing tools," 2010 2nd International Conference on Education Technology and Computer, Shanghai, 2010, pp. V5-359-V5-363, doi: 10.1109/ICETC.2010.5530052

[5] R. Heriansyah, S.A.R AI-Attas, and M.M. Ahmad Zabidi, Segmentation of PCB Images into Simple Generic Patterns using Mathematical Morphology and Windowing Technique', CoGRAMM Melaka, Malaysia 2002

[6] P. R. Masalkar, P. S. Kasliwal, "Study of the image processing algorithms for defect detection of pcbs", Int. J. Eng. Technol. Sci. Res, vol. 4, no. 6, pp. 606-612, 2017

[7] M. Baygin, M. Karakose, A. Sarimaden And E. Akin, (2017) "Machine vision based defect detection approach using image processing," 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), Malatya, 2017, pp. 1-5.

[8] Kim, H.W., Yoo, S.I. (2014). Defect detection using feature point matching for non-repetitive patterned images.Pattern Anal Applic 17, 415–429 (2014).

[9] Prof. Malge P. S., Nadaf R. S., 2014, A Survey : Automated Visual PCB Inspection Algorithm, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 03, Issue 01 (January 2014)

[10] Mahmoud taha, Eid & Emary, Eid & Moustafa, Khalid. (2014). Automatic Optical Inspection for PCB Manufacturing: a Survey. International Journal of Scientific and Engineering Research. Volume 5. 1095-1102.

[11] Kim, H.W., Yoo, S.I. Defect detection using feature point matching for non-repetitive patterned images.Pattern Anal Applic 17, 415–429 (2014).

[12] Chaudhary, Vikas & Dave, Ishan & Upla, Kishor. (2017) "Automatic visual inspection of printed circuit board for defect detection and classification". (2017). 732-737. 10.1109/WiSPNET.2017.8299858.

[13] P. Wei, C. Liu, M. Liu, Y. Gao and H. Liu, (2018)"CNN-based reference comparison method for classifying bare PCB defects," in The Journal of Engineering, vol. 2018, no. 16, pp. 1528-1533, 11 2018, doi: 10.1049/joe.2018.8271.

[14] Huang, Weibo and Peng Wei. (2019): "A PCB Dataset for Defects Detection and Classification." ArXiv abs/1901.08204 (2019):

[15] Zhang, Linlin & Jin, Yongqing & Yang, Xuesong & Li, Xia & Duan, Xiaodong & Sun, Yuan & Liu, Hong. (2018). Convolutional Neural Network Based Multi-label Classification of PCB Defects. The Journal of Engineering. 2018. 10.1049/joe.2018.8279.

[16] 20] Borba, J., & Facon, J. (1995). A printed circuit board automated inspection system. 38th Midwest Symposium on Circuits and Systems. Proceedings, 1, 69-72 vol.1.

[17] Du-Ming Tsai & Yan-Jheng Su (2009) Non-referential, self-compared shape defect inspection for bond pads with deformed shapes, International Journal of Production Research, 47:5, 1225-1244, DOI: 10.1080/00207540701549590

[18] Benedek, Csaba. (2011). Detection of soldering defects in Printed Circuit Boards with Hierarchical Marked Point Processes. Pattern Recognition Letters. 32. 1535-1543. 10.1016/j.patrec.2011.06.006.

[19] Kobayashi, H.H., Hara, Y., Doi, H., Takai, K., & Sumiya, A. (1998). Hybrid Defect Detection Method Based on Shape Measurement and Feature Extraction for Complex Patterns. MVA.

[20] Fikret Ercal, Filiz Bunyak, Hao Feng, "Context-sensitive filtering in RLE for PCB inspection," Proc. SPIE 3517, Intelligent Systems in Design and Manufacturing, (9 October 1998); https://doi.org/10.1117/12.326933

[21] Chen T.Q., Zhang J., Zhou Y., Murphey Y.L. (2001) A Smart Machine Vision System for PCB Inspection. In: Monostori L., Váncza J., Ali M. (eds) Engineering of Intelligent Systems. IEA/AIE 2001. Lecture Notes in Computer Science, vol 2070. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-45517-5_57

[22] Chang, P., Chen, L. & Fan, C. A case-based evolutionary model for defect classification of printed circuit board images. J Intell Manuf 19, 203–214 (2008). https://doi.org/10.1007/s10845-008-0074-8

[23] LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. Nature 521, 436–444 (2015). https://doi.org/10.1038/nature14539

[24] O'Shea, K., & Nash, R. (2015). An Introduction to Convolutional Neural Networks. arXiv.

[25] Faghih-Roohi, S., Hajizadeh, S., Núñez, A., Babuška, R., & Schutter, B.D. (2016). Deep convolutional neural networks for detection of rail surface defects. 2016 International Joint Conference on Neural Networks (IJCNN), 2584-2589.

[26] Soukup D., Huber-Mörk R. (2014) Convolutional Neural Networks for Steel Surface Defect Detection from Photometric Stereo Images. In: Bebis G. et al. (eds) Advances in Visual Computing. ISVC 2014. Lecture Notes in Computer Science, vol 8887. Springer, Cham. https://doi.org/10.1007/978-3-319-14249-4_64

[27] R. Ding, L. Dai, G. Li and H. Liu, (2019) "TDD-net: a tiny defect detection network for printed circuit boards," in CAAI Transactions on Intelligence Technology, vol. 4, no. 2, pp. 110-116, 6 2019, doi: 10.1049/trit.2019.0019.

[28] R. Girshick, "Fast R-CNN," 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, 2015, pp. 1440-1448, doi: 10.1109/ICCV.2015.169