End term T12 - Project Phase II

on

Automated PCB Fault Detection System

[Comparative Study of Image Processing and Transfer Learning Techniques]

Submitted by

Project Members

Aneesh Pol (1032170079)

Muskaan Rajput (1032170239)

Sakshi Reddy (1032170358)

Swarali Paygude (1032170562)

Under the Internal Guidance of

Prof. Madhura Phatak

School of Computer Engineering and Technology

MIT World Peace University, Kothrud,

Pune 411 038, Maharashtra - India

2020-2021

SCHOOL OF COMPUTER ENGINEERING AND TECHNOLOGY

C E R T I F I C A T E

This is to certify that, Aneesh Pol, Muskaan Rajput, Sakshi Reddy, Swarali Paygude of BTech.(Computer Science & Engineering) have completed their project titled "Automated PCB Fault Detection System[Comparative Study of Image Processing and Transfer Learning Technique]" and have submitted this Capstone Project Report towards fulfillment of the requirement for the Degree-Bachelor of Computer Science & Engineering (BTech-CSE) for the academic year 2020-2021.

[Dr/ Prof. Madhura Phatak]	**[Dr. Vrushali Kulkarni]**
Project Guide	**Program Head**
School of CET	**School of CET**
MIT World Peace	**MIT World Peace**
University, Pune	**University, Pune**

Internal Examiner:

External Examiner:

Date:

Acknowledgement

We would like to take this opportunity to thank our internal guide Prof. Madhura Phatak ma'am for giving us all the help and guidance we needed. We are extremely grateful to her for her kind support and patience. Her valuable suggestions helped in shaping this project and giving us the direction to move ahead.

We are also grateful to Prof. Dr. Vrushali Kulkarni, Head, School of Computer Engineering & Technology for this opportunity and her encouragement.

Swarali Paygude,

Muskaan Rajput,

Sakshi Reddy,

Aneesh Pol

Abstract

Printed circuit boards (PCBs) are a crucial component in most electronic devices. Accurate PCB manufacturing is critical as manufacturers are required to produce PCBs in large quantities. Therefore, maintaining the quality of such large numbers of PCBs is challenging. Automated inspection systems can prove helpful in quality maintenance. Such systems overcome the limitations of manual inspection for a large number of PCBs. Automated visual PCB inspection can provide fast detection of defects and therefore can prove to be an asset in the manufacturing process. This project aims to achieve fault detection of bare PCBs through two different methods; one being the traditional algorithmic approach using image processing, which involves the use of image subtraction method and the other one being a transfer learning approach, which involves the pre-trained VGG16 model. A comparative study of both the methods is done.

List of Figures

List of Tables

Contents

Chapter 1
Introduction

1.1 Project statement

To develop an image based fault detection system to identify various defects on bare Printed Circuit Boards (PCBs)

[a comparative study of algorithmic computations of image processing and transfer learning approaches]

1.2 Area

PCB fault detection can be achieved using a variety of techniques. In our project we aim to use two techniques for pcb fault detection and classification. The first method is Transfer Learning. Transfer learning falls under the domain of machine learning. In transfer learning a model is developed for a particular task and is then reused as the starting point for another task. The pre-trained model which we would be using is VGG16.

The second method which we would be implementing in our project is a traditional handcrafted unsupervised algorithm. Image processing is an extremely large domain and consists of a number of methods and techniques within it. Image processing is basically a method adopted to perform certain operations on the image in order to extract useful information from it.

1.3 Project Introduction and Aim

Printed Circuit Boards are used to support and connect electrical and electronic components using conductive paths, pads and other features etched from one or more layers of copper laminated onto between sheet layers of a non conductive substrate. There are primarily two types of PCB boards on which defects can be detected. One is the mounted PCB and the other is bare PCB.

During the manufacturing of PCBs, it can encounter a number of defects. The first step is to produce the manufacturing data followed by type setting. The board is then cut. The board used could be made up of glass or copper. CNC machine drilling is then performed depending on the design. This is then followed by track painting and cleaned with water so that the track is visible. Tracks are printed using ink and then the etching process takes place followed by masking. This forms our bare printed circuit board. The errors that can occur during this stage are mouse bite, open circuit, missing hole etc. Detecting these defects at this stage is very essential. It can save a lot of money, time and effort in the long run. Further, the electronic components are mounted on the bare PCB. This gives rise to a number of soldering defects. In our project we are focusing on detection of faults occurring on bare pcbs. In the past a number of supervised as well as unsupervised methods were adopted for detection of faults on bare PCB.

In this project we will be using two methods viz transfer learning which is a supervised method and image processing techniques which is an unsupervised method and then compare the two on the basis of accuracy and time. Accurate PCB manufacturing is critical as manufacturers are required to produce PCBs in large quantities. Maintaining the quality of such large numbers of PCBs is challenging. Automated inspection systems can prove helpful in quality maintenance. Such systems overcome the limitations of manual inspection for a large number of PCBs. Automated visual PCB inspection can provide fast detection of defects and therefore can prove to be an asset in the manufacturing process.

Chapter 2
Literature Survey

In the development of technology, image processing and computer vision have experienced rapid development in recent years. Rather than the manual inspection computer vision technique is playing an important role in the advanced automatic industry. In order to reduce the cost in manufacturing of electronic boards, defects in the bare PCB must be identified. Many researchers have done a lot of work regarding PCB inspection.

Numerous methods are suggested by the researchers on PCB defect classification. In this section, brief evaluations of a few essential contributions to the existing works of literature are presented.

Sr. No.	Paper Title	Publication Journal & Year	Summary	Research gap/Future scope	Author
1.	Study of the Image Processing algorithms for defect detection of PCBs	International Journal of Engineering Technology Science and Research (IJETSR) ,2017	3 approaches: Template matching, Image subtraction, Image morphology to detect defects and classify them into groups. Dataset was created using Pi camera for capturing images.	Some of the defects cannot be addressed individually. During grayscale to binary transformation, undesirable images may be produced by the noise. Future scope includes inspection and analysis of a PCB with Surface Mounted Devices.	Pratiksha R. Masalkar, Prabha S. Kasliwal
2	Components Free Electronic Board Defect Detection and Classification Using Image Processing Technique	International Journal of Engineering Research and Technology (IJERT), 2018	Using the subtraction method the defects are identified A different algorithm is introduced to classify the 7 defects.	-	Harshitha R, Apoorva G C, Ashwini M C, Kusuma T S

3	Automatic PCB Defects Detection and Classification using Matlab	International Journal of Current Engineering and Technology 2014	Detect and classify all the known 14 defects Used MATLAB Image subtraction method for defect detection KNN classification algorithm for classification of defects	Further study on robust technique to differentiate the defects on single PCB image. Study further defect detection and classification on loaded PCB	Prachi P. Londe, S. A. Chavan
4	Very deep convolutional networks for large-scale image recognition	Published as a conference paper at ICLR 2015	Input:224*224*3. Pre-processing: Subtracting the mean RGB value, computed on the training set from each pixel. Filter: 3x3 Stride: fixed to 1 pixel. Spatial pooling: 5 max pooling layers. Fully connected layers: 3 FC. Final layer: softmax layer All hidden layers are equipped with ReLu.	-	Karen Simonyan and Andrew Zisserman

Table 1 Literature Survey

Chapter 3
Problem Statement

3.1 Project Scope

Our project aims to prepare a comparative study between transfer learning, with the help of a pre-trained object detection model (VGG-16) and traditional handcrafted algorithm using Image Processing Algorithms such as Image subtraction, to detect defects in bare PCBs.
The dataset contains labelled images of 12 different Printed Circuit Boards having six types of defects (Missing hole, Mouse bite, Open circuit, Short, Spur, Spurious copper). Using these images the model will be trained to detect these defects in real-time.
While the project uses a pre-trained model, it has to be modified to fit our purpose. The output layers are changed to detect the faults and classify them accordingly.
The project aims to locate and classify multiple defects on a single PCB simultaneously and make bounding boxes to visualize them.
Image processing algorithms such as Image subtraction are considered to be generally less effective than CNN based algorithms but considering the rather small size of the dataset they might return a better accuracy. Hence the comparative study might prove effective to understand the importance of choosing the right algorithm, with regard to the size of the dataset, in the accuracy of the model.

3.2 Project Limitations

Although CNN based models are widely used for object detection in images, due to the small size of the available dataset, the accuracy of the model could be affected adversely.
The lack of a standard dataset makes the comparison of our model with the accuracy of other models difficult.
The transfer learning model (VGG16) was originally designed to detect a fixed number of classes (objects) in images which did not include Printed circuit boards, hence the model has to be modified accordingly to achieve best results.

3.3 Project Objectives

- To create a dataset by using image augmentation on the available images
- To create a transfer learning based model to detect defects in PCBs
- To create a Image processing based algorithm to detect defects in PCBs
- To prepare a comparative study between the two approaches

Chapter 4
Project Requirements

4.1 Resources

4.1.1 Software & hardware requirements

Software:

- VGG16 model
- Python (Libraries like OpenCV, Keras)
- Jupyter Notebook
- GIMP (image editor)

Hardware:

Laptop/PC

4.2 Risk Management

4.2.1 Project Risk factors

Risks	Solutions
Low quality PCB images	Train model with different kinds of images consisting of noise. Take images with proper illumination.
Failure to detect the defect.	Train your model with a bigger and proper dataset.
Failure to correctly classify the defects	Choosing an appropriate classification algorithm.
Failure to procure a dataset with reasonable amounts of images.	Create your own dataset. Image augmentation, rotation, duplicating.

Chapter 5
System Analysis Proposed Architecture

5.1 Block Diagram

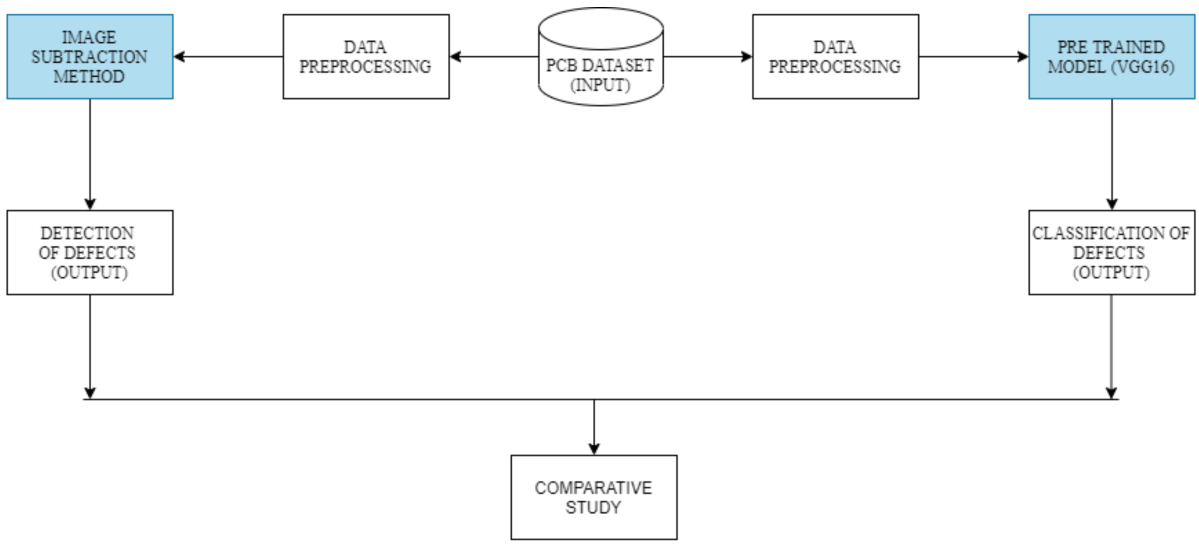


Fig1. System Block Diagram

5.2 Assumptions and Dependencies

- Lighting Conditions: The results obtained are specific to the current lighting conditions of the images present in our dataset.
- Size Constraints: For handcrafted methods, the size of the template image has to be the same as that of the test image. The image size has to be 224,224 when dealing with VGG 16.
- Large Dataset: By using a larger dataset, it is assumed that the VGG 16 model will give a better accuracy.
- Template Images: In order to detect faults using any handcrafted algorithms, a template image is required.

5.3 General Constraints

- There are no standard datasets readily available. Thus additional efforts are required to build a new dataset from scratch.

- The images used for fault detection in pcb need to be clear inorder to achieve better results.
- The developed system is specific to our dataset, hence will not provide the desired accuracy on a different dataset.
- The dataset consists of more defective images as compared to non defective images.

Chapter 6

Project Plan

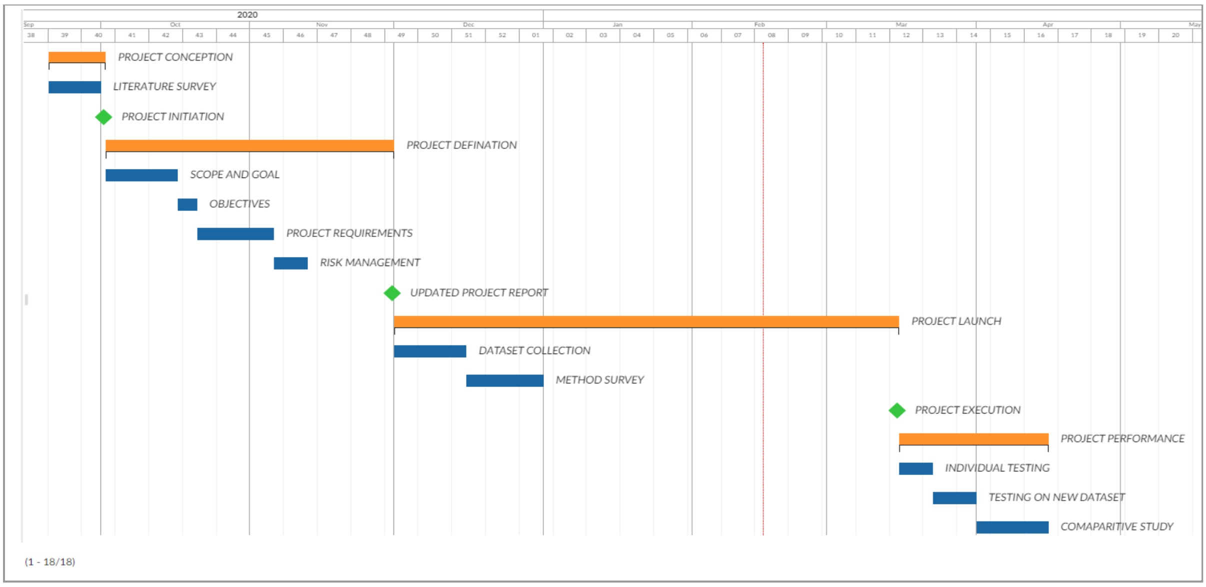


Fig2. Gantt Chart

All open

+Create | Filter 1 | Table

ID	SUBJECT	TYPE	STATUS	ASSIGNEE	PRIORITY
34	PROJECT CONCEPTION	PHASE	Scheduled	-	Normal
54	LITERATURE SURVEY	TASK	New	-	Normal
55	PROJECT INITIATION	MILESTONE	New	-	Normal
38	PROJECT DEFINATION	PHASE	New	-	Normal
39	SCOPE AND GOAL	TASK	New	-	Normal
40	OBJECTIVES	TASK	New	-	Normal
41	PROJECT REQUIREMENTS	TASK	New	-	Normal
42	RISK MANAGEMENT	TASK	New	-	Normal
43	UPDATED PROJECT REPORT	MILESTONE	New	-	Normal
44	PROJECT LAUNCH	PHASE	New	-	Normal
45	DATASET COLLECTION	TASK	New	-	Normal
46	METHOD SURVEY	TASK	New	-	Normal
47	PROJECT EXECUTION	MILESTONE	New	-	Normal
48	PROJECT PERFORMANCE	PHASE	New	-	Normal
49	INDIVIDUAL TESTING	TASK	New	-	Normal
50	TESTING ON NEW DATASET	TASK	New	-	Normal
51	COMAPARITIVE STUDY	TASK	New	-	Normal

(1 - 18/18)

Fig3. Project plan

Chapter 7
Implementation

7.1 Discuss dataset

Image Processing

Defect /PCB Types	PCB 01	PCB 05	PCB 07	PCB 10	PCB 11	Total
Missing Hole	20	10	10	5	10	55
Mouse Bite	20	10	10	5	10	55
Open Circuit	20	10	10	6	10	56
	60	30	30	16	30	166

Table2. Original dataset of bare PCB images

We also performed rotations of all images at 90, 180 and 270 degrees from the original orientation. As a result, we had 4 times the total number of original images, ie, 166 * 4 = 664.

Transfer Learning - VGG-16

- Dataset - The dataset used for this project is a combination of two open source datasets obtained from github. The dataset has 115 images each of missing hole, mouse bite, open circuit, short, spur and spurious copper. Out of these set of images, we will be working with two defects; missing hole and mouse bite. Further we divided these images into training and validation sets. For mouse bite the training set consists of 84 images and the testing set consists of 31 images. Similarly for missing holes and open circuit the training and testing sets consist of 82 and 33 images respectively. The images in the dataset are synthetic pcb images. From the second dataset we obtained 1570, 1538, 1454 training images for mouse bite, missing hole and open circuit respectively. Also, 282, 294, 299 testing images for mouse bite, missing hole and open circuit respectively.The dimensions of each image are 3034,1586,3. 3 because they are RGB images. Evidently the number of images present in the dataset is very miniscule. In order to increase the number of images in the data set we performed image augmentation which helped us achieve a total of 1680 training and 620 testing images for mouse bite along with 1640 training and 660 testing images for missing holes.

- Image Augmentation - Image Augmentation is used to artificially expand the existing dataset by developing a variety of versions of each image present. Image augmentation is performed to increase the number of images which inturn helps train the model more efficiently. Image can be processed in a number of ways such as, horizontal flip, vertical flip, zoom in, zoom out, contrast, rotation at specific or random angles etc. This is achieved using Image Data Generator API from Keras. Hence using this technique we created 20 random augmented images for each of the images present in the original dataset.

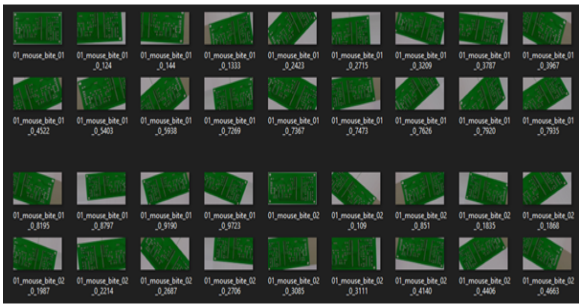


Fig4. Augmented Images

7.2 Methodology

Image Subtraction

Image processing is a method to perform certain operations on an image, which will result in an enhanced image, or extracting some useful information from it. In this type of processing, the input is an image and the output may be an image or characteristics/features associated with that image. Image processing basically includes the following three steps:

- Importing the image via image acquisition tools
- Analysing and manipulating the image
- Output to show the results

The approach used in this system is based on the Image Subtraction technique, in order to detect the PCB defects from the images of bare PCBs. Image subtraction is one of the image processing techniques, also known as pixel subtraction. In this technique, the digital numeric values of pixels from an image are subtracted from the values of pixels of another image. This is done for the purpose of detecting the changes between the two images. The functions used within the system are taken from the OpenCV library of Python. Initially, pre-processing is done on the template and test images, to resize them, so that both are of the same size. This is necessary, since pixel-to-pixel mapping is done during image subtraction. Both the images should also be of the same orientation. Then, they are converted to grayscale images. After that, both images are blurred using a blurring method, and adaptive thresholding is performed on them. Finally, an image subtraction function is applied, which subtracts the test image from the template image, to get the result as the difference image. This image contains the defective part, which is shown as bright(white) pixels on a dark(black) background. We can thus observe the defective part, and easily see where the defects are located. Contour detection is also applied to count the number of defects detected in the test image. This can further be used while calculating accuracy of the system.

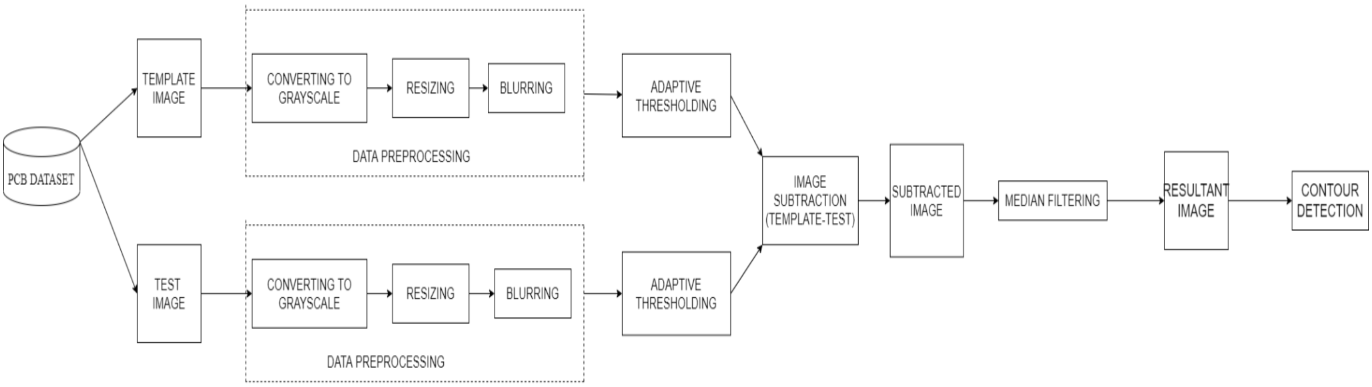


Fig5. Image Processing Module

Transfer Learning

Transfer learning allows solutions to multiple problems and applications. It is primarily a technique in which any neural network model is initially trained on a similar dataset to that of the actual problem. On completion of stage one, certain layers from the then trained model are retrained to generate a new model addressing the required problem statement. A model is initially trained using a larger dataset. The trained layers are frozen and used as it is while addressing a problem statement with a comparatively smaller dataset. In such cases only certain layers, usually the fully connected and output layers are modified as per the problem statement. Hence using this technique models can be trained using smaller datasets as well without compromising the accuracy. Some popular models include ResNet-50, VGG-16, VGG-19, Inception V3, XCeption. Various word embedding models include Word2Vec, GloVe, FastText. The pre-trained model which we have used for pcb fault detection is VGG-16. Using pre-trained models is recommended when one has a large dataset in hand. The original VGG-16 architecture has 2 fully connected layers followed by a softmax for output. The 16 in VGG-16 refers to it having 16 layers that have weights. The network is a pretty large network and it has about 138 million parameters. All the hidden layers use ReLU as its activation function. ReLU is more computationally efficient because it results in faster learning and it also decreases the likelihood of vanishing gradient problem. When using VGG-16 there is no requirement of a reference image. For training we used Transfer Learning in Keras on VGG-16. First and foremost we imported keras and all the methods and functions required to build and train the model. Inorder to import all the images we used ImageDataGenerator from keras. We had to resize the images to 224,224,3 as the model was originally trained using this size. We then import VGG-16 from keras with pre-trained weights which were originally trained on imagenet.

Once we have downloaded the model, the model now needs to be modified as per our problem statement. This needs to be done because pcb detection does not belong to any of the 1000 classes which was used originally. Next, there were in all 1000 classes, as per our problem statement we have only 3 classes which are missing holes, open circuit and mouse bite. The activation function used was ReLu in the original model. In case of more classes we can also use the softmax activation function. The output layer consists of 1000 neurons, we will have to modify it to 3 output neurons. Along with this we add a flatten layer as well.Whatever output we receive we condense that into one dimension using the flatten layer. The model was trained twice. Once using the original dataset and then using the dataset consisting of the augmented images. Different combinations of epochs were experimented with to attain maximum accuracy.

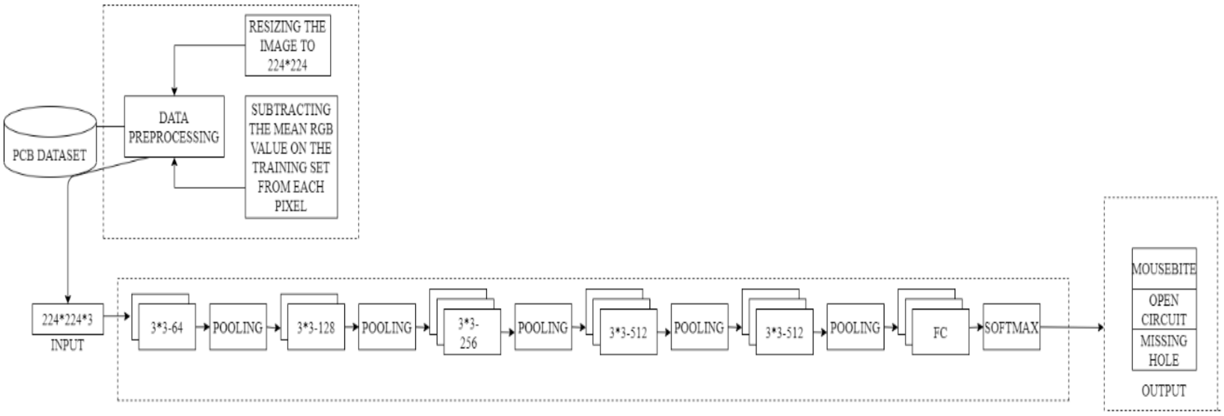


Fig6. Transfer Learning Module

7.3 Algorithm

Image Processing

As Shown in Fig 2, the steps followed in image subtraction process are:

- Grayscaling - Grayscaling is the process of converting an image from other color spaces to shades of gray. By finding the average of the R, G and B values a gray scale image is obtained.

Gray scale Value = (R + G + B)/3

In a grayscale image a single byte for each pixel stores a value from 0 to 255, and thus covers all possible shades of gray.

- Thresholding - It is used to turn a grayscale image into a binary image based on a specific threshold value.Pixels below that value are converted to black, and pixels above that value are converted to white. There are two main categories of thresholding: Simple thresholding and Adaptive thresholding.Various types of simple thresholding are binary thresholding, binary thresholding inverted, truncated thresholding, etc.
- Adaptive thresholding - It is an advanced type of thresholding. The basic difference in simple and adaptive thresholding is that in simple thresholding - the threshold value is global, i.e., same for all the pixels in the image while in adaptive thresholding - the threshold value is calculated for smaller regions, i.e., different threshold values for different regions. The function used is:

cv2.adaptiveThreshold()

- Image subtraction - The pixel values in two images are subtracted with the help of

cv2.subtract()

Both the images should be of equal size, depth and orientation.The difference between two images img1 and img2 can be expressed as :

ResultantImg = cv2.subtract(img1,img2)

- Detection - The three types of defects detected are: MissingHole, MouseBite and OpenCircuit. The subtracted image is passed through a median filter to remove the noise. Following is the syntax of this method:

  medianBlur(src, dst, ksize)

  The image thus obtained can detect the different defects.
- Contour Detection - In OpenCV, finding contours is like finding white object from black background. The contours are a useful tool for shape analysis and object detection and recognition. We used this to detect the number of defects in an image. The method used for this purpose is:

  cv2.findContours()

Transfer Learning

VGG16 Methodology

1. Get path of the directory containing dataset
2. Using os, numpy and image libraries convert the image dataset into a numpy array of images resized to (224,224)
3. Roll axis and transform array into the shape (no. of images, 224,244,3)
4. Label the data and use one hot encoding
5. Shuffle and Split the data into training and testing
6. Use Pretrained VGG16 model from keras without including top
7. Using the output from the convoluted block 5 make custom fully connected layers and output layer of the number of required classes
8. Set the layers till convolution block 4 as not trainable and the rest as trainable
9. Compile model using adam optimizer and categorical cross entropy
10. Fit model to the train test split dataset

Chapter 8
Performance Evaluation

Performance evaluation of both modules is as given below:

Image Subtraction Module	Transfer Learning Module
For evaluating the performance of image subtraction technique, we used a simple percentage calculation method to find the accuracy of detection of the defects. The formula used is as follows: **acc_percent = (total_detected/total_defects)*100** where total_detected is the total number of defects detected in an image, and total_defects is the total number of defects actually present in that image.	Accuracy: This metric creates two local variables, total and count that are used to compute the frequency with which y_pred matches y_true. This frequency is ultimately returned as binary accuracy: an idempotent operation that simply divides total by count. Loss: Categorical cross entropy, it is a Softmax activation plus a Cross-Entropy loss. If we use this loss, we will train a CNN to output a probability over the C classes for each image. It is used for multi-class classification.

Chapter 9
Result and Analysis

- ## Image Subtraction

We implemented the handcrafted algorithm of image subtraction on all the images in our dataset where we subtracted the test image from the reference image of five different PCBs. We then analysed the result by finding the accuracy of the detection using a simple percentage calculation method. The average detection accuracies of missing hole, mouse bite and open circuit were found to be 86.05, 72.64 and 80.43 respectively.

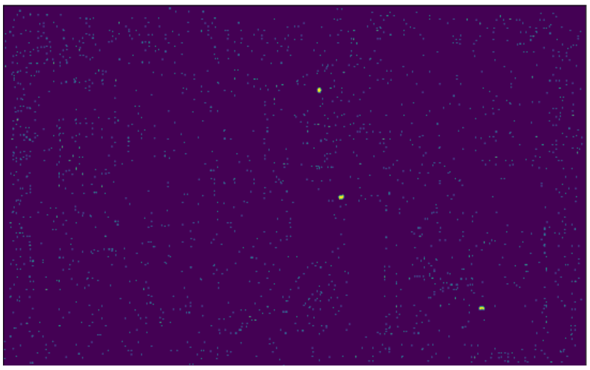


Fig7. After Image Subtraction (Missing hole)

Fig8. After Median Filtering (Missing hole)

Defect	PCB 01	PCB 05	PCB 07	PCB 10	PCB 11	Average accuracy
Missing Hole	93.3	41.3	100	100	95.6	86.05
Mouse Bite	70.8	60.4	72	84	76	72.64
Open Circuit	71.7	84.4	70	93.33	82.7	80.43

Table3. Result table for defect detection(image subtraction)

- **Transfer Learning- VGG-16**

  VGG16 Transfer Learning model was implemented on the augmented dataset to achieve these results. The results returned by the model are illustrated in a tabular format depicting the number of Epochs that the model was trained for, the number of steps completed in each epoch, the loss calculated by binary cross entropy and finally the accuracy that was obtained.

DATASET USED	STEPS_PER_EPOCH	EPOCH	LOSS	ACCURACY
ORIGINAL	5	10	4.4146	0.5373
AUGMENTED	5	10	4.3124	0.5562
MODIFYING THE LAYERS	23	12	0.6653	0.7232

Table4. Result table for defect detection(VGG-16)

Applications

- This project can be used to detect defects such as missing holes, mouse bite and open circuit on bare PCBs.
- Educational institutes can use the system to maintain the PCBs used in their laboratories
- PCB manufacturing industry can use this system for Quality Assurance
- Electronic gadgets which extensively use PCBs can also benefit from this system

Conclusion

In this project, we have tried to develop a system that would be helpful in the field of PCB manufacturing. Our prototype models will detect and classify bare PCB defects, which will help in minimizing the waste of defective PCBs which are otherwise discarded altogether. The comparative study will help in identifying the better and more accurate approach among the two implemented, according to the circumstances and available resources. The dataset used is an open-source dataset, which can later be benchmarked and people can use this prototype along with the dataset, for further research and development.

Future prospects of the project

In the future work, such a dataset can be used which includes different types(designs) of PCBs, i.e. a variety of PCB images. Also, a dataset with a large number of images can be used. For an unbiased result, a more balanced dataset can be used, which contains equal amounts of defected and non-defected images.

The system can be modified to incorporate detection of more PCB defects. The system can also be improvised in such a way that real-time defect detection is possible.

An integrated system can be developed in which the defects present on the bare PCB can be detected and localized using image subtraction and the defects can further be classified using a pre-trained model.

One more prospect can be to experiment using different pre-trained models, and other handcrafted algorithms.

References

[1] Pratiksha R. Masalkar, Prabha S. Kasliwal, “Study of the Image Processing algorithms for defect detection of PCBs”, International Journal of Engineering Technology Science and Research, June 2017

[2] Harshitha R, Apoorva G C, Ashwini M C, Kusuma T S, “Components Free Electronic Board Defect Detection and Classification Using Image Processing Technique”, International Journal of Engineering Research & Technology ,2018

[3] Daniel Katz Bonello, Yuzo Iano, Umberto Bonello Neto, “A New Based Image Subtraction Algorithm for Bare PCB Defect Detection”, International Journal Multimedia and Image Processing, September 2018

[4] G. Prathima, A. Yasaswini Naga Lakshmi, et. al., “Defect Detection in PCB Using Image Processing”, International Journal of Advanced Science and Technology, 2020

[5] Ms. Vinita P, Mr. shivkant kaushik, “PCB Fault Detection by Image Subtraction Method”, International Journal of Scientific Engineering and Applied Science (IJSEAS), January 2016

[6] Karen Simonyan & Andrew Zisserman, “Very deep convolutional networks for large-scale image recognition”, ICLR, 2015

[7] Fa'iq Raihan, Win Ce, “PCB Defect Detection USING OPENCV with Image Subtraction Method”, IEEE, 2017

[8] Ihar Volkau, Abdul Mujeeb, Dai Wenting, Erdt Marius, Sourin Alexei, “Detection Defect in Printed Circuit Boards using Unsupervised Feature Extraction Upon Transfer Learning”, IEEE, 2019

[9] Leandro H. de S. Silva, George O. de A. Azevedo, Bruno J. T. Fernandes, Byron L D Benzerra, “Automatic Optical Inspection for Defective PCB Detection Using Transfer Learning.”, IEEE, 2019

[10] Ajay Pal Singh Chauhan, Sharat Chandra Bhardwaj, “Detection of Bare PCB Defects by Image Subtraction Method using Machine Vision”, Proceedings of the World Congress on Engineering 2011 Vol II

[11] Jithendra P R Nayak, Parameshachari B D, et. al., “Identification of PCB Faults using Image Processing”, IEEE, 2017

[12] Prachi P. Londe, S. A. Chavan, “Automatic PCB Defects Detection and Classification using Matlab”, International Journal of Current Engineering and Technology 2014

Publication Details

Our results-based paper(manuscript) on our project was submitted to and accepted by the following international journals:

- The International Organization of Scientific Research (IOSR) Journal
- International Journal for Research in Applied Science and Engineering Technology (IJRASET)

Published the paper titled "Comparative Study of Image Processing and Transfer Learning Techniques for an Automated PCB Fault Detection System" at IJRASET Volume 9, Issue VI, June 2021.

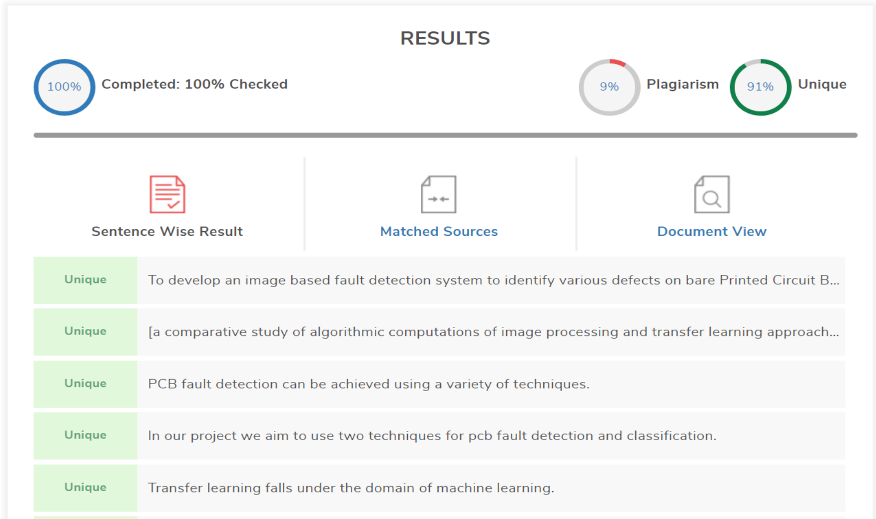
Appendices

A. Base Paper(s)

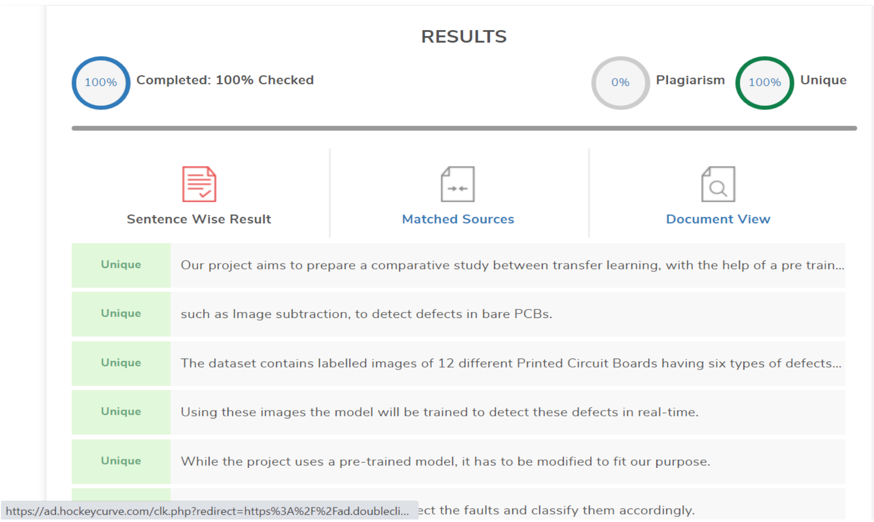
1. Study of the Image Processing algorithms for defect detection of PCBs
2. Very deep convolutional networks for large-scale image recognition

B. Plagiarism Report from any open source/proprietary source

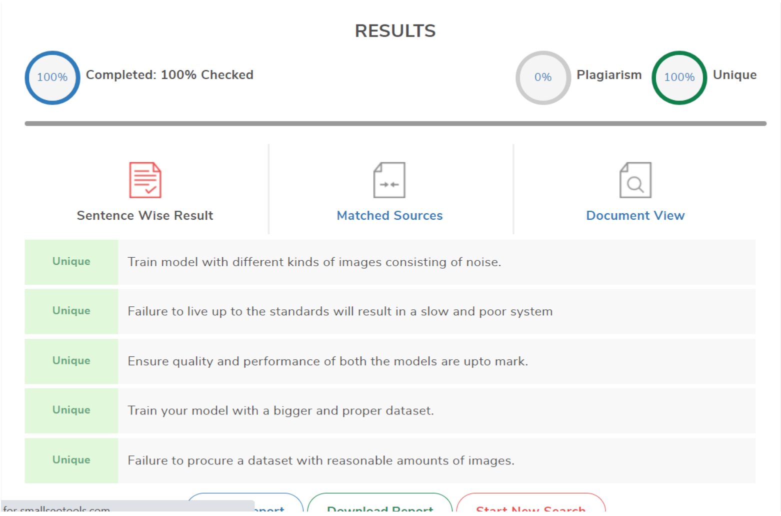
CHAPTER 1



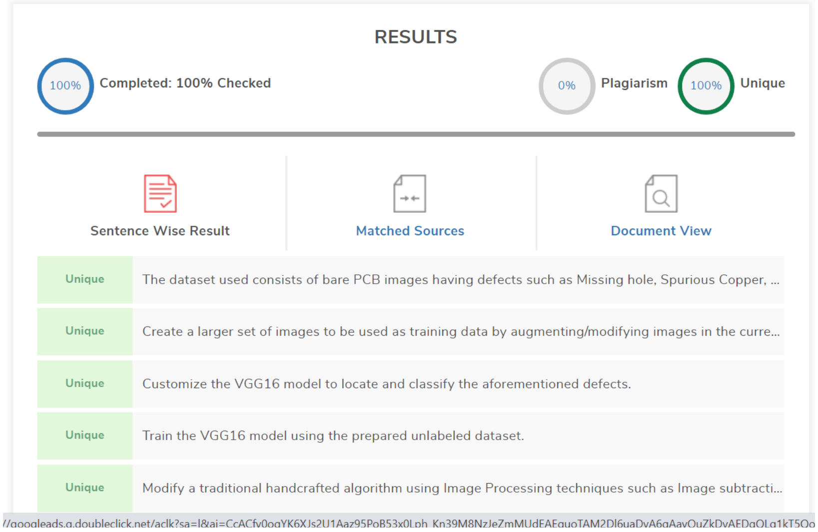
CHAPTER 3



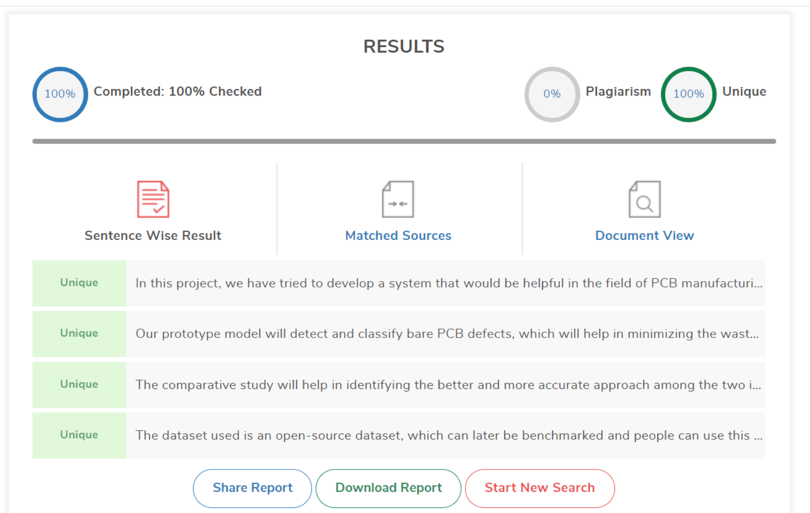
CHAPTER 4



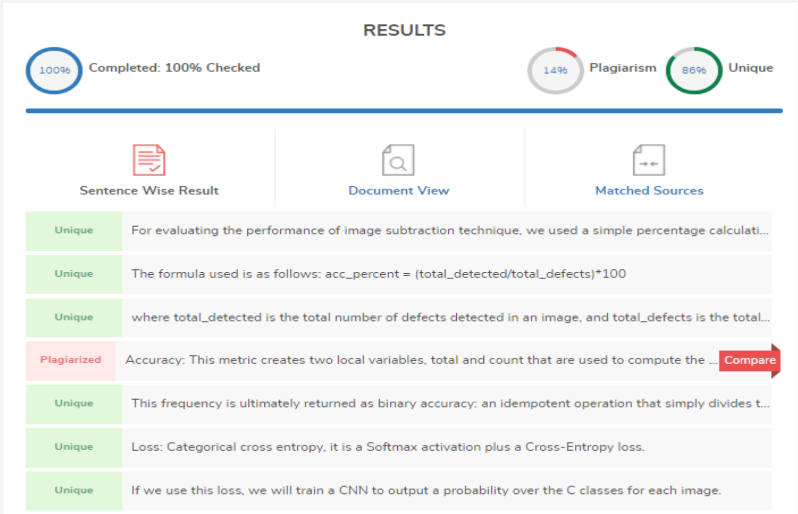
CHAPTER 5



CHAPTER 7



CHAPTER 8



CHAPTER 9