Journal of Positive Psychology & Wellbeing http://journalppw.com
2022, Vol. 6, No. 5, 4326–4335

Comparative Analysis of Deep Learning Models for PCB Defects Detection and Classification

Farhanath .K, Owais Farooqui, Dr.Senthil Kumar.R,Asique .K

Abstract

In the modern-day, just about any electronic device uses Printed Circuit Boards (or PCBs). Nevertheless, these are still prey to manufacturing defects, which are extremely hard to detect manually. Other researchers have come up with a slew of ways to improve PCB inspection and problem categorization. As a result, fewer innovative approaches in this sector are being developed as a result of the researchers' failure to disclose their datasets before. The Weibo Huang and Peng Wei study titled 'A PCB Dataset for Defects Detection and Classification'[1] was used to produce a synthetic PCB dataset. The dataset consists of 1386 photos with six types of faults for the detection, classification, and registration of tasks. As part of the evaluation, it looks at how well various convolutional neural network designs diagnose errors using the reference-based technique (proposed in the same research that generated this dataset). When compared to traditional techniques that need pixel-by-pixel processing, a novel method first locates the faults and then classifies them using the selected neural network, which performs better on the dataset than previous methods. This method also has a proposed neural network [1] whose performance is tested against popular models like Inception and VGG using transfer learning.

Keywords

Printed Circuit Board; Comparative Analysis; Automated Optical Inspection; Deep Learning; Convolutional Neural Network; XCEPTION; Inception V3; ResNet50; VGG16; VGG19; EfficientNetB7; PCB Dataset; Reference Based Method

Introduction

A Printed Circuit Board (PCB) serves as the linking foundation for all electronic devices and houses a plethora of electrical components. The ability of these components to function well and communicate with each other depends heavily upon the pathways that connect them, which exist on the PCB board. The conventional way used to detect the defects that may arise in a PCB due to manufacturing errors was to use manual detection, but it leads to the workers being easily fatigued due to the strain put on their eyes as they try to detect a tiny defect, and thus have low efficiency. A popular alternative used now is automated optical inspection (AOI) based on machine vision. These function well for simple PCBs, but AOI sometimes fails to detect and classify the defects within a permissible error range as PCBs grow more complex over time. Research is further hampered by the fact that many researchers in this field do not publish their datasets, which leads to fewer new research methods being proposed.

Standard AOI methods for detecting PCB defects can be divided into three categories [2]: comparison, non-reference verification, and hybrid A standard picture of a PCB that is free of faults is created as a template, and then the PCB that requires inspection is compared to it in order to discover any previously unknown problems. Even while it seems simple at first, there are a slew of underlying issues to contend with, such as inconsistent lighting, massive storage needs, and incorrect labeling. Using a process known as non-reference verification, a circuit is wire track, pad, and hole alignment may be checked without the need of a template board. There is no need for a template PCB using this technique. Nevertheless, it may have difficulties detecting larger defects as it has to scan the image of the PCB in a very detailed manner to detect minor defects.

The hybrid method proposed in the PCB datasets paper [1] combines both methods mentioned above.

1. Methods and Methodologies

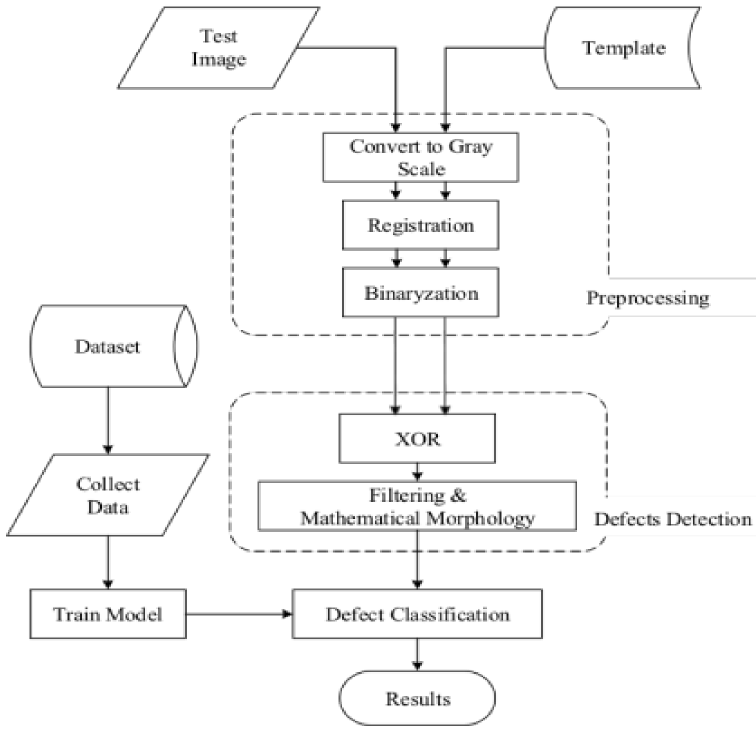


Fig. 1. The flowchart for PCB inspection in the proposed hybrid way

If a deep learning methodology can learn precise representations and handle data in a sequential manner, it has become quite popular in recent years because of this. Several known deep learning models, such as VGG16, InceptionV3, and Xception, are built in this study and their performance is analyzed.

As an alternative to the ones we use now, a system developed by Wen-Yen Wu et al. [2] provided an automated visual inspection system for PCBs. A technique called "elimination-subtraction" was used, which removes parts of the examined picture and then adds back in the template image. In order to find any remaining errors in the PCB picture, an elimination technique is next performed. “In order to classify a defect, we look at three different metrics: the kind of item identified, the difference in object numbers, and the difference in background numbers between an inspected picture and a reference image.

LI Zheng-ming et al. [3] also employed digital image processing technology-based reference techniques to categorize the flaws by obtaining the number of linked areas, Euler numbers, and the area of defects of the template and inspected image, respectively. The results of the trial demonstrated that the approach was capable of detecting suspicious activity in real-time.”

Wiring tracks, soldering pads, and holes were divided into three sections by Vikas Chaudhary et al. [4]: positive and negative faults, respectively. Pixels and the number of linked components may be used to classify each flaw in a specific portion.

"Shashi Kumar et al. [5] offered a non-referential strategy in light of the problems in registration. The examined picture was divided into copper and non-copper sections to be analyzed individually, and a 3D color histogram was used to capture the overall color distribution." Analyses comparing this model's accuracy to that of other non-referential techniques are based on data from the PCB manufacturing sector.

Rudi Heriansyah et al. [6] developed a novel method for classifying faults that relies on neural networks. Thousands of faulty patterns were utilized for training and testing, and several defective patterns reflecting relevant defect kinds were created. Defect classification technique based on neural networks performed well, as shown by the findings.

To put it another way, printed circuit board assembly (PCBA) is a board that has all of its components connected onto it and is capable of executing its intended electrical function before the dataset [1] was created. Our goal is a PCB without any components, not a PCBA that has been destroyed for recycling. As a result, we cannot use these PCBA datasets. The authors of the aforementioned study [1] provide a synthetic dataset consisting of 1386 photos of bare circuit boards. As templates with varied flaws, half of these PCBs have been manually rotated to mimic a circumstance where the PCBs are wrongly fitted. Using human eyes, we verified that each picture comes from one of ten widely accepted standard template boards for web design. Each PCB contains 3 to 5 flaws, and we offer a bounding box for each defect. PCBs that have been rotated are likewise given rotation data. An extensive collection of PCB-related techniques for detecting, classifying and registering PCBs may be compared using this dataset. You may get the dataset at no cost by going to this website.

Using this dataset and the procedure mentioned in Fig. 1, a method to train the neural networks is made clear.

2. Dataset Description

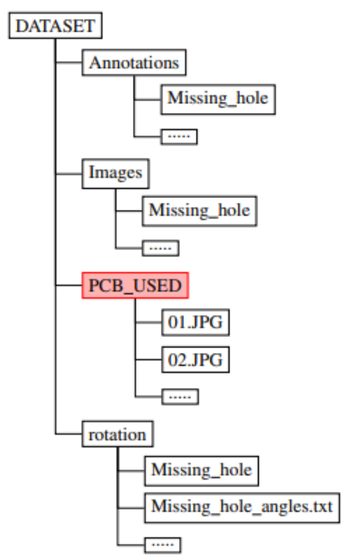


Fig. 2. The dataset structure

.

	Train	Val	Test
Missing hole	599	203	192
Mouse bite	600	190	194
Open circuit	598	189	177
Short	600	189	177
Spur	600	187	189
Spurious copper	600	203	203
Total	3597	1161	1148

“Table 1. Distribution of defects in training, val and test folders”

3. Training and Classification

It is necessary to determine the defect category after determining the exact location of flaws. Based on a comparison of the test picture to a template, conventional approaches use pixel-by-pixel comparisons to choose enough features to indicate errors. [2], [3], [4], [6].

End-to-end deep learning models allow the faulty picture to be used directly as input for the model, eliminating previously discussed computationally intensive approaches. This allows for faster classification results. Use of convolutional neural networks for defect classification has been discussed above.

3.1 VGG 16 [11] [14]

Karen Simonyan and Andrew Zisserman created the VGG-Visual Geometry Group at Oxford University in 2014. An essay entitled "Very deep convolutional networks for wide scale image recognition" has also been released. 2014 ImageNet Large Scale Visual Recognition Challenge was won by this research paper (ILSVRC). An image collection with over a million photos and 1000 classifications named ImageNet evaluated the block model with 92.7 percent accuracy on the top five. This network may be classified as simple or uniform in architecture. Three convolutional layers, each weighted by its own convolutional weight, are stacked on top of each other. Replaces Alex Net's huge kernel size filter with a smaller one.

3.2 Inception V3 [12]

At its core, Inception is an advanced CNN. Because of this, new models like inception V2, V3, V4, and inception Resnet have been created, each one building on the preceding one in a different way. Including the pooling levels, inception v1 and v2 have a total of 27 layers. An auxiliary loss occurred in the V1 model, which is resolved in V2. A total of 48 layers are included in Inception V3, which includes the pooling layers.

3.3 Xception [13]

The Xception Model was suggested by Francois Chollet. Only depthwise separable convolution layers are used in this convolutional neural network design. Speed increases significantly when used in combination with the Initiation Architecture, which substitutes ordinary Inception modules with depth-aware Separable Convolutions.

3.4 ResNet-50

One of Kaimimg He is Convolution Neural Networks, the Resnet-50, has 50 Deep Layers and was unveiled in 2015. Convolution Block and Identity Block are two of the blocks in this concept. In Resnet-50, the idea of a "skip connection" is employed. Convolution layers make up 48 of the 50 layers; the other two are the average and maximum pool layers. It is possible to import the ImageNet database, which contains over a million examples, and pre-train it. [15] [16]

3.5 VGG 19 [19]

“This is a VGG model version with 19 layers, called VGG19 (16 convolution layers, 3 fully connected layers, 5 MaxPool layers and 1 SoftMax layer). VGG11, VGG16, and additional variations are available. There are 19.6 billion FLOPs in VGG19.”

3.6 EfficientNetB7 [17]

EfficientNet relies heavily on MBConv, an inverted bottleneck conv formerly referred to as MobileNetV2. An in-depth separable convolution was used to reduce calculation time by roughly k2 over conventional layers, using shortcuts between bottlenecks by linking a much smaller number of channels (relative to expansion layers). A 2-dimensional convolution window has a kernel size k, which sets the window's height and breadth.

3.7 Custom CNN [1]

The Custom CNN is built as defined in the paper where this dataset was created. The architecture is defined in Table 2.

Layers	Output Size	Operation
Convolution	32×32	7×7 Conv, stride 2
Pooling	16×16	3×3 Max Pool, stride 2
Block1	16×16	$\begin{bmatrix} 1\times1 & Conv \\ 3\times3 & Conv \end{bmatrix} \times 6$
Transition Layer	16×16	1×1 Conv
	8×8	2×2 Avg Pool
Block2	8×8	$\begin{bmatrix} 1\times1 & Conv \\ 3\times3 & Conv \end{bmatrix} \times 6$
Classification Layer	1×1	7×7 Adaptive Avg Pool
		6D fully-connected, softmax

Table 2: Detailed Parameter Setting of the Custom CNN

4. Results

The experiments measured accuracy, ROC score, recall, precision, F1 score and a confusion matrix. The Adam analyzer with an LR of 0.0001 was used to construct all move learning models for 50 age groups. Stochastic Gradient Descent was used to create a bespoke model for 50 different ages, with a learning rate of 0.01% and a decay rate of 0.10%. On the same test set, the measurements were taken.

The review is created by dividing the number of true positives by the number of genuine negatives.

Recall (R) = TP/ (TP + FN)

In computing, precision is the percentage of retrieved examples that are relevant to the task at hand. Positive results are subtracted from all negative results to arrive at this value.

Precision (P) = TP/ (TP + FP)

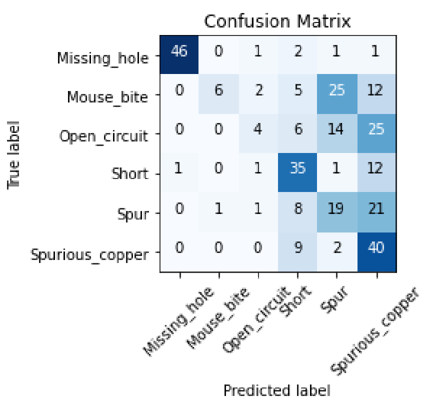
The harmonic average of recall and accuracy is what is used to get an F1 rating. Only when accuracy and recall are both 1, does the F1 score become 1.

F1 score = 2 * P * R/ (P + R)

Where R and P are Recall and Precision respectively

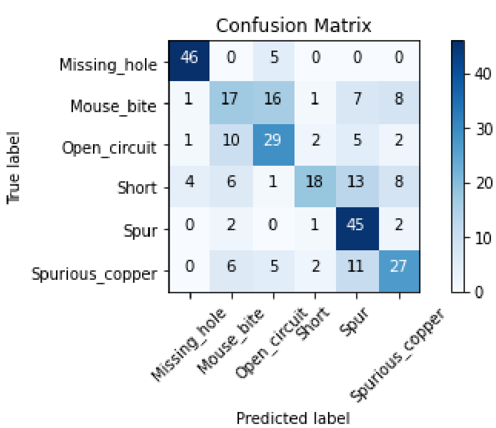
Following are the confusion matrices along with recall, precision and F1 scores obtained for each network. 0,1,2,3,4,5 are the classes Missing hole, Mouse bite, Open circuit, Short, Spur, Spurious copper in that sequence.

4.1 VGG 16



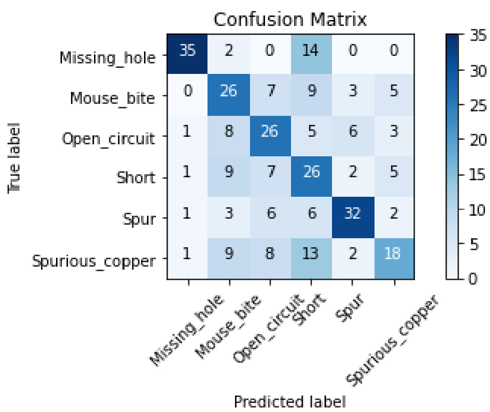
	precision	recall	f1-score
0	0.98	0.90	0.94
1	0.86	0.12	0.21
2	0.44	0.08	0.14
3	0.54	0.70	0.61
4	0.31	0.38	0.34
5	0.36	0.78	0.49

4.2 Inception V3



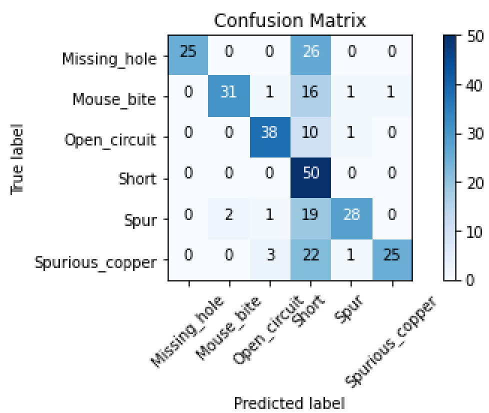
	precision	recall	f1-score
0	0.88	0.90	0.89
1	0.41	0.34	0.37
2	0.52	0.59	0.55
3	0.75	0.36	0.49
4	0.56	0.90	0.69
5	0.57	0.53	0.55

4.3 Xception



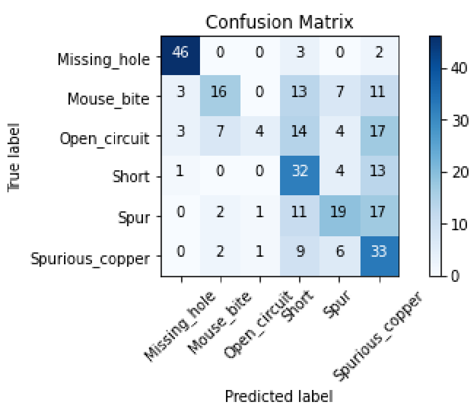
	precision	recall	f1-score
0	0.90	0.69	0.78
1	0.46	0.52	0.49
2	0.48	0.53	0.50
3	0.36	0.52	0.42
4	0.71	0.64	0.67
5	0.55	0.35	0.43

4.4 ResNet-50



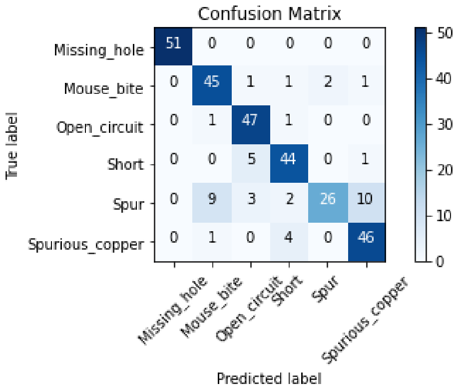
	precision	recall	f1-score
0	1.00	0.49	0.66
1	0.94	0.62	0.75
2	0.88	0.78	0.83
3	0.35	1.00	0.52
4	0.90	0.56	0.69
5	0.96	0.49	0.65

4.5 VGG19



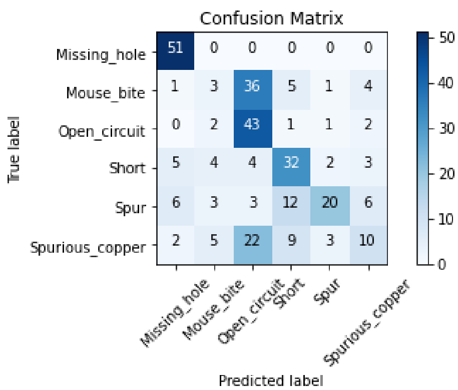
	precision	recall	f1-score
0	0.87	0.90	0.88
1	0.59	0.32	0.42
2	0.67	0.08	0.15
3	0.39	0.64	0.48
4	0.47	0.38	0.42
5	0.35	0.65	0.46

4.6 EfficientNetB7



	precision	recall	f1-score
0	1.00	1.00	1.00
1	0.80	0.90	0.85
2	0.84	0.96	0.90
3	0.85	0.88	0.86
4	0.93	0.52	0.67
5	0.79	0.90	0.84

4.7 Custom CNN



	precision	recall	f1-score
0	0.78	1.00	0.88
1	0.18	0.06	0.09
2	0.40	0.88	0.55
3	0.54	0.64	0.59
4	0.74	0.40	0.52
5	0.40	0.20	0.26

Model	**Accuracy**	**ROC AUC Score**
VGG 16	0.50	0.847
Inception V3	0.60	0.878
Xception	0.54	0.853
ResNet-50	0.65	0.970
VGG 19	0.50	0.842
EfficientNetB7	0.86	0.978
Custom CNN	0.76	0.949

5. Conclusion

PCB Defect Detection methods still have scope for massive improvements. Better detection will enable us to build electronics that last longer and work better. Deep learning has enabled us to do this in ways that free us from the restrictions of human limits.

In this paper, a complete evaluation of defect detection abilities is performed. The evaluation is carried out using different algorithms, and multiple CNN architectures are explored. The public database for training and evaluation is presented. The availability of a large public database is one of the most challenging scenarios to improve a robust and effective PCB flaw detection system. We believe that, with the help of academics working on datasets from various public databases, greater levels of deep learning may be developed and are anticipated to identify errors adequately.

On evaluating the 7 CNN architectures, we concluded that EfficientNetB7 had outperformed all others by a generous margin, followed by the Custom CNN. We can see the apparent benefits of transfer learning. Not only does it provide us with a model with higher accuracy, but it also is less time-consuming.

It is the goal of this project to give a simple deep learning model for the identification of PCB faults to engineers, data scientists, and the research community as a whole. Despite the fact that additional photos may be gathered in the future, our dataset is sufficient for our needs at this point in time. It is possible that future study may concentrate on a larger dataset, a more robust algorithm, a faster detection procedure, and the development of successful non-reference comparison approaches that eliminate the use of PCB 'templates,' all while enhancing accuracy.

References

[1] W. Huang and P. Wei, “A PCB dataset for defects detection and classification,” 2019, arXiv:1901.08204. [Online]. Available: http://arxiv.org/abs/1901.08204

[2] Wen Yen Wu, Mao Jiun J Wang, and Chih Ming Liu. Automated inspection of printed circuit boards through machine vision. Computers in Industry, 28(2):103–111, 1996.

[3] L. I. Zheng-Ming, L. I. Hong, and Jun Sun. Detection of PCB based on digital image processing. Instrument Technique & Sensor, 61(8):87–89, 2012.

[4] Vikas Chaudhary, Ishan R. Dave, and Kishor P. Upla. Automatic visual inspection of printed circuit board for defect detection and classification. Proceedings of the 2017 International Conference on Wireless Communications, Signal Processing and Networking, WiSPNET 2017, 2018-Janua:732–737, 2018.

[5] Shashi Kumar, Yuji Iwahori, and M K Bhuyan. Proceedings of International Conference on Computer Vision and Image Processing. 459, 2017.

[6] Rudi Heriansyah, Syed Abdul Rahman Al-attas, and Muhammad Munim Ahmad Zabidi. Neural Network Paradigm for Classification of Defects on PCB. Jurnal Teknologi, 39(1):87–103, 2003.

[7] Christopher Pramerdorfer and Martin Kampel. A dataset for computervision-based PCB analysis. Proceedings of the 14th IAPR International Conference on Machine Vision Applications, MVA 2015, pages 378–381, 2015

[8] K. Seyhan, T.N. Nguyen, S. Akleylek, K. Cengiz, S.H. Islam

Bi-GISIS KE: Modified key exchange protocol with reusable keys for IoT security

Journal of Information Security and Applications, 58 (2021), Article 102788

[9] B.D. Parameshachari, R.P. Kiran, P. Rashmi, M.C. Supriya, Rajashekarappa, H.T. Panduranga

Controlled partial image encryption based on LSIC and chaotic map

ICCSP (2019, January), pp. 60-63

[10] K. Yu, L. Lin, M. Alazab, L. Tan, B. Gu

Deep Learning-Based Traffic Safety Solution for a Mixture of Autonomous and Manual Vehicles in a 5G-Enabled Intelligent Transportation System

IEEE Transactions on Intelligent Transportation Systems (2021),

[11] Srikanth Tammina (2019); Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images; International Journal of Scientific and Research Publications (IJSRP) 9(10) (ISSN: 2250-3153), DOI: http://dx.doi.org/10.29322/IJSRP.9.10.2019.p9420

[12] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2818-2826, doi: 10.1109/CVPR.2016.308.

[13] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1800-1807, doi: 10.1109/CVPR.2017.195.

[14] Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. CoRR, abs/1409.1556.

[15] V.T. Pham, T.N. Nguyen, B.H. Liu, T. Lin

Minimizing latency for multiple-type data aggregation in wireless sensor networks

2021 IEEE Wireless Communications and Networking Conference (WCNC) (2021, March), pp. 1-6

IEEE

[16] G.B. Rajendran, U.M. Kumarasamy, C. Zarro, P.B. Divakarachari, S.L. Ullo

Land-use and land-cover classification using a human group-based particle swarm optimization algorithm with an LSTM Classifier on hybrid pre-processing remote-sensing images

Remote Sensing, 12 (24) (2020), p. 4135

[17] M. Tan, Q.V. Le, Efficientnet: rethinking model scaling for convolutional neural networks, arXiv:1905.11946(2019).

[18] HRIPCB: a challenging dataset for PCB defects detection https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/joe.2019.1183

[19] Face Recognition using VGG16. https://www.linkedin.com/pulse/face-recognition-using-vgg16-gunish-

[20]. Senthil kumar, R., Geetha, B.G. Signature Verification and Bloom Hashing Technique for Efficient Cloud Data Storage. *Wireless Pers Commun* 103, 3079–3097 (2018). https://doi.org/10.1007/s11277-018-5995-8

[21]. R. Senthilkumar, B. G. Geetha, "Asymmetric Key Blum-Goldwasser Cryptography for Cloud Services Communication Security," Journal of Internet Technology, vol. 21, no. 4 , pp. 929-939, Jul. 2020.

https://jit.ndhu.edu.tw/article/view/2322