journal homepage: https://journal.pktm.com.my/

International Journal of Technical Vocational and Engineering Technology

e-ISSN2710-7094, Vol 5, No. 2, 2024

Design of Defect Detection Tools in Through Hole Technology (THT) Solder Joints with Classification Based on Deep Learning

Efrizon[1*], Dara Aprila[1], Laxsmy Devy[1], Tri Artono[1], and Era Madona[1]

[1]Department of Electrical Engineering, Politeknik Negeri Padang, Jl. Kampus, Limau Manis, Padang, 25164, Indonesia.

*Corresponding Author email: efrizon@pnp.ac.id

ARTICLE INFO
Article History:
Received 15 June 2024
Revised 30 September 2024
Accepted 17 October 2024

Keywords:
PCB THT;
Soldering Defect;
Webcam Logitech c920;
Jetson Nano;
YOLO CNN.

ABSTRACT
A tool has been created for soldering defect detection to reduce the processing time for testing the quality of solder joints for THT PCBs in soldering practical activities at the Padang State Polytechnic. This tool is equipped with the help of a Logitech c920 Webcam and a Jetson Nano microprocessor which is used to store and run programs that have been created in Python programming software so that this tool can be used portably. The research method starts with creating a dataset, labeling, training data, and making tools. In this research, the YOLO Convolutional Neural Network method was used to help determine soldering defects with four classifications, namely hole soldering, bridge soldering, void on solder pad, and good soldering. The test results showed that the tool was able to detect these four classifications with mAP@0.5 values, Bridge 97.20, Good 54.50%, Hole 83.90%, and padless 57.20%. Overall, the tool can function well.

1.0 Introduction

With the rapid development of artificial intelligence, several fields have begun to integrate this technology to increase efficiency and accuracy in their processes. The efforts have been made to assess recent advancements in various processes, such as welding and soldering (Buang, A. S. et al., 2024). Padang State Polytechnic, Electronics Study Program faces challenges in evaluating student practice results in the Printed Circuit Board (PCB) Technology course, especially in terms of assessing PCB soldering results which is currently still done manually by instructors or lecturers. However, with a large number of students, this manual assessment method is less efficient. Therefore, it is necessary to develop automatic visual inspection tools using artificial intelligence to detect defects in PCB solder (Wang et al., 2017).

In facing this challenge, there are two main options for developing visual inspection tools, namely deep learning and image processing (Al et al., 2020). The rapid advancement of artificial intelligence technology, especially in the context of deep learning, has opened new opportunities in detecting and interpreting complex patterns from visual data such as PCB solder images (Susa et al., 2020). Deep learning has the advantage of providing a high level of accuracy in detecting defects in PCB solder because it can learn complex patterns (Tsai et al., 2019). However, the disadvantage is that it requires heavy computing and significant resources, and model development and training are time-consuming. Meanwhile, image processing has the advantage of having simpler computations and can be implemented on hardware with lower specifications (Dovbnych & Plechawska–Wójcik, 2021). However, the drawback is that the detection accuracy is usually lower than deep learning, especially for complex cases such as detecting defects in PCB soldering.

Based on these problems, a defect detection tool in solder joints through hole technology (THT) with deep learning-based classification is developed. Several related studies have been carried out previously, using Convolutional Neural Networks (CNN) (Lim et al., 2019) for the inspection of surface-mounted technology (SMT) component soldering results. The research results show that the proposed system is effective for devices with complex shapes, which are numerous Lots. Similar research was also carried out by (Zhang & Shen, 2021) who applied Faster Region-based Convolutional Neural Network (R-CNN). Tests carried out on this algorithm produced an accuracy rate of 94%. The disadvantage of using this algorithm is the high computational requirements. Meanwhile, research conducted by Li et al. (2019) compared several deep learning algorithms, including Faster R-CNN with YOLO.

The tests carried out showed that YOLO had the highest performance with a mean Average Precision (mAP) of 94.19%. YOLO uses a deep learning architecture, but object detection is capable of using CPU computing. In this research, we implemented defect classification in the soldering process based on three types of soldering defects as shown in Figure 1 following the soldering practice assessment standards at the Padang State Polytechnic based on deep learning using the YOLO algorithm. The research objective is to implement artificial intelligence to increase accuracy in THT PCB solder defect detection. By creating this tool, it is hoped that it can reduce the processing time for testing the quality of solder joints in soldering practicums at the Padang State Polytechnic and contribute to the field of computer vision for solder defect detection, which is applied in the field of soldering competency training in vocational schools.

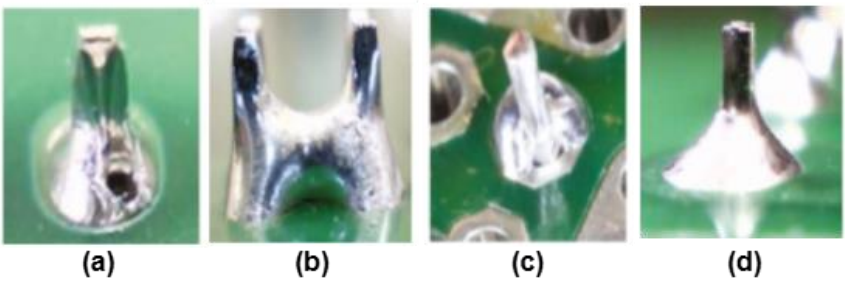


Figure 1: Type of defects considered in the practicum assessment (a) hole solder, (b) Bridge soldering, (c) Void on solder pad (d) Good soldering

2.0 Research Methodology

Several electronic devices are needed for research. These are Jetson Nano, web camera, and TFT LCD. Stages for achieving each target research on research methods are made in the form flow diagram (flowchart). Research method flow diagram, as shown in Figure 2.

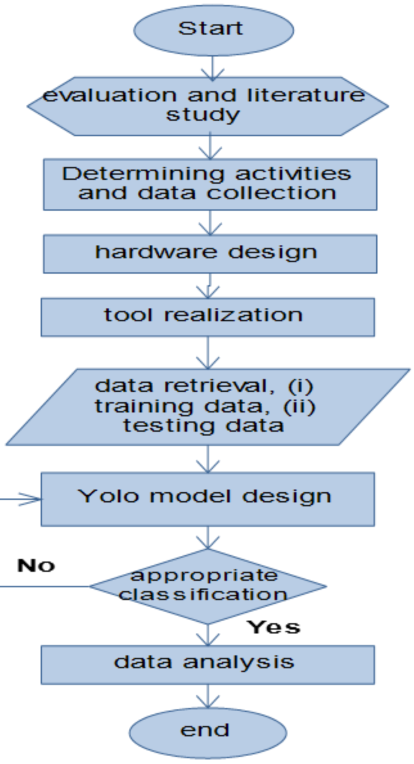


Figure 2: Research method flow diagram

2.1 General Architecture

According to Ahmad Agung et al., (2023), this THT PCB solder defect detection tool uses a webcam that is activated using the Python and OpenCV programs which is connected to the Jetson Nano (Varna & Abromavičius, 2022). The LED connected serially to the USB is used for lighting. The THT PCB is placed on the webcam for the image to be captured, then the PCB path is processed and detected by the YOLO algorithm to determine the possibility of solder defects on the PCB path. The detection results are then displayed on the TFT screen in the form of a user interface (UI) display as shown in Figure 3.

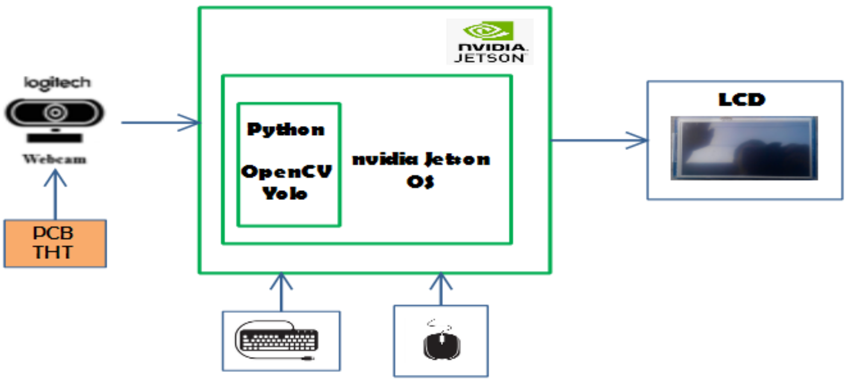


Figure 3: System Block Diagram

2.2 Dataset

Creating a dataset begins with determining the classification or class that will be detected. The classifications made are excess, bridge, disturbed, and good soldering. Image data collection of 200 images with various positions and PCB groups. The dataset collected is in the form of images with the entire image format using (.jpg) format as shown in Figure 4.

Figure 4: Dataset

2.3 Labelling Dataset

This labeling process begins by creating a bounding box along with the class name for each object. The collected dataset can be labeled as; categories hole with 30 labels, padless with 90 labels, bridge with 112 labels, and good 165 labels. This labelling uses the YOLO format where image data will be converted into array data which will later be used to create the YOLO algorithm. The labeling image is shown in Figure 5.

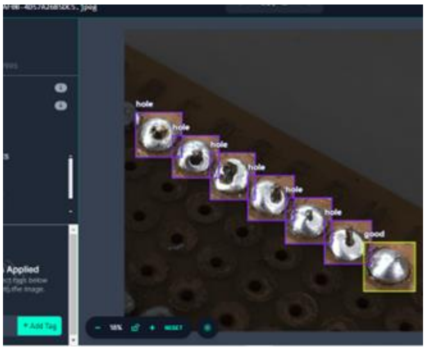
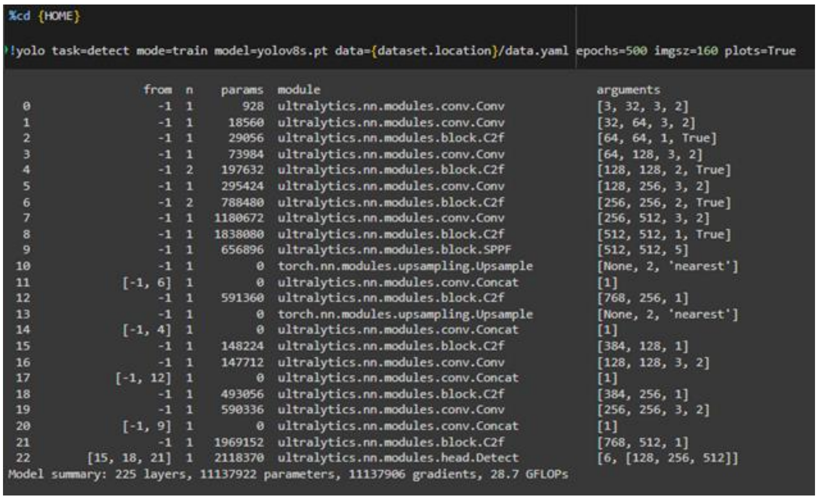


Figure 6: Labelling datasets

2.4 Training Data

This data training was carried out after obtaining annotations from the detected object classes, then Google Colab was used and to be more specific, Yolo training was used on a network with OpenCV. This iterative convolution method in YOLO applies regions to locate detected objects. The facilities required for this model training process are GPU facilities. The framework used is Darknet with a pre-trained weight, namely darknet53.conv.74. The results of this model training are files in the format (.cfg) and (.weights), namely files containing configuration sequences and weights related to Sequence Alignment and Modeling. The training data image is shown in Figure 7.



```
%cd {HOME}

!yolo task=detect mode=train model=yolov8s.pt data={dataset.location}/data.yaml epochs=500 imgsz=160 plots=True

                   from  n    params  module                                       arguments
  0                  -1  1       928  ultralytics.nn.modules.conv.Conv             [3, 32, 3, 2]
  1                  -1  1     18560  ultralytics.nn.modules.conv.Conv             [32, 64, 3, 2]
  2                  -1  1     29056  ultralytics.nn.modules.block.C2f             [64, 64, 1, True]
  3                  -1  1     73984  ultralytics.nn.modules.conv.Conv             [64, 128, 3, 2]
  4                  -1  2    197632  ultralytics.nn.modules.block.C2f             [128, 128, 2, True]
  5                  -1  1    295424  ultralytics.nn.modules.conv.Conv             [128, 256, 3, 2]
  6                  -1  2    788480  ultralytics.nn.modules.block.C2f             [256, 256, 2, True]
  7                  -1  1   1180672  ultralytics.nn.modules.conv.Conv             [256, 512, 3, 2]
  8                  -1  1   1838080  ultralytics.nn.modules.block.C2f             [512, 512, 1, True]
  9                  -1  1    656896  ultralytics.nn.modules.block.SPPF            [512, 512, 5]
 10                  -1  1         0  torch.nn.modules.upsampling.Upsample         [None, 2, 'nearest']
 11             [-1, 6]  1         0  ultralytics.nn.modules.conv.Concat           [1]
 12                  -1  1    591360  ultralytics.nn.modules.block.C2f             [768, 256, 1]
 13                  -1  1         0  torch.nn.modules.upsampling.Upsample         [None, 2, 'nearest']
 14             [-1, 4]  1         0  ultralytics.nn.modules.conv.Concat           [1]
 15                  -1  1    148224  ultralytics.nn.modules.block.C2f             [384, 128, 1]
 16                  -1  1    147712  ultralytics.nn.modules.conv.Conv             [128, 128, 3, 2]
 17            [-1, 12]  1         0  ultralytics.nn.modules.conv.Concat           [1]
 18                  -1  1    493056  ultralytics.nn.modules.block.C2f             [384, 256, 1]
 19                  -1  1    590336  ultralytics.nn.modules.conv.Conv             [256, 256, 3, 2]
 20             [-1, 9]  1         0  ultralytics.nn.modules.conv.Concat           [1]
 21                  -1  1   1969152  ultralytics.nn.modules.block.C2f             [768, 512, 1]
 22        [15, 18, 21]  1   2118370  ultralytics.nn.modules.head.Detect           [6, [128, 256, 512]]
Model summary: 225 layers, 11137922 parameters, 11137906 gradients, 28.7 GFLOPs
```

Figure 7: Training Data

2.5 Model Evaluation

There are several methods for evaluating system performance, namely recall, precision, F1 score, Intersection over Union, mean Average Precision, and Accuracy (Quach et al., 2023). Provisions for prediction classes based on the location in the confusion matrix data after training are shown in Table 1.

Table 1: Confusion Matrix

		Actual Class	
		Positive	**Negative**
Prediction Class	Positive	True Positive (TP)	False Negative (FP)
	Negative	False Negative (FN)	True Positive (TN)

3.0 Result and Discussion

3.1 Evaluation of THT PCB Soldering Defect Model

THT PCB Soldering Defect Model Evaluation Results are shown in Table 2.

Table 2. Performance Evaluation of THT PCB Soldering Defect Model

Class	Ground Truth	Detection Result		Precision	Data Testing	mAP The average value of Average Precision
		False Positive	True Positive			
Bridge	13	2	11	0.972	25	0.663
Good	20	1	19	0.545		
Hole	69	7	62	0.839		
Pad_less	155	16	139	0.572		

Object detection results/ground truth from 25 images that have been tested from four classes, namely Bridge with 13, Good with 20, hole with 69, and padless with 155. Table 2 also shows for the Bridge class the precision values are 97.20, Good 54, 50%, Hole 83.90%, and padless 57.20%, and the average accuracy of mAP with confidence 0.5 was 66.30%.

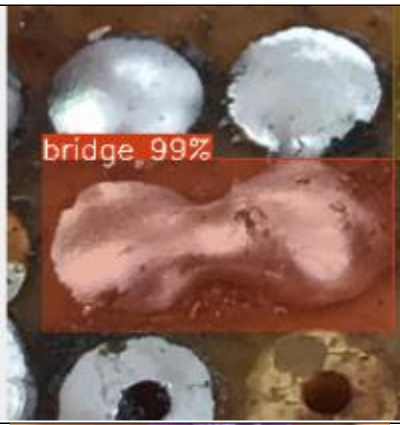
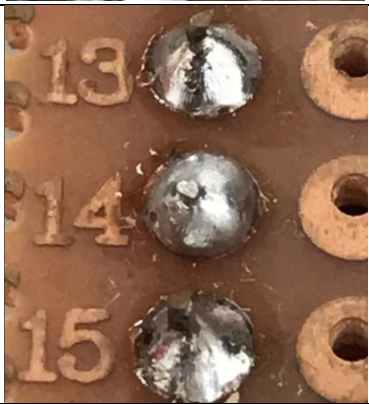
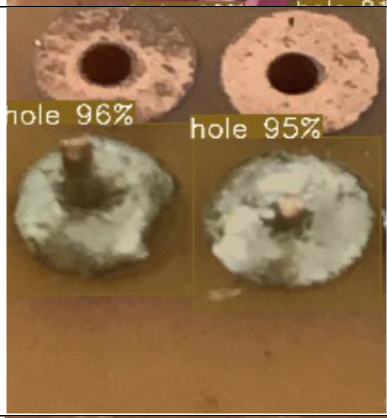
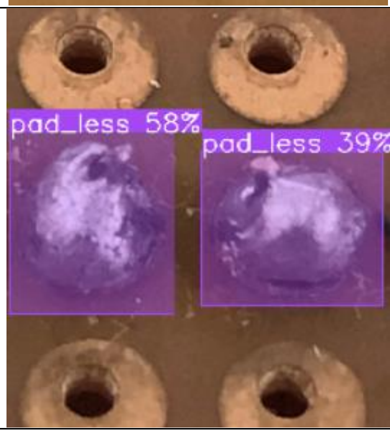
3.2 System testing

The data taken by the Webcam placed on the Webcam Stand is shown in Figure 8. The data that has been taken is extracted and processed on the Jetson Nano and displayed on the monitor in the form of a UI.

Figure 8: Tool System Testing

Then sampling of 25 random data was carried out. Testing was carried out at the Padang State Polytechnic and sampling data was prepared; the test results are shown in Table 3.

Table 3. System test results

No	Test data	Label (Confidence)	Result
1		Bridge accuracy rate 99%	
2		Good accuracy rate of 82% to 85%	
3		Hole accuracy rate 95% to 96%	
4		Padless accuracy rate 39% to 58%	

From the test results, an evaluation is carried out using equations (Quach et al., 2023). Table 4 shows the evaluation results of the tests carried out. Based on Table 4, from the 25 data tested, the precision value was 77.30%, recall 68.00%, with mAP 66.30%.

Table 4. Test Data Evaluation Results

Class	Precision	Recall	mAP
All	0.773	0.68	0.663
Bridge	0.972	1	0.995
Excess	0.938	0.125	0.273
Good	0.545	0.429	0.429
Hole	0.839	0.963	0.96
Pad_less	0.572	0.882	0.655

4.0 Conclusion

The THT PCB soldering defect detection tool was built with Deep Neural Network (DNN) and the YOLOv3 algorithm using Jupyter Notebook on Google Colab. PCB damage can be read when prediction confidence > 4.9 and cannot be read when confidence < 5.0. The dataset consists of a total of 200 PCB images with 30 labels for holes, 90 labels for padless, 112 labels for bridges, and 165 labels for good. In some comparison results there are still boxes that are considered defective, this is due to the light intensity and PCB position changing when taking pictures. Evaluation testing with 25 images that were similar to the labeling image resulted in mAP@0.5 Bridge 97.20, Good 54.50%, Hole 83.90%, and padless 57.20%. Suggestions for further development are to collect more datasets so that the detection process is more accurate. Using additional equipment, such as a conveyor to later become a sorter for PCB soldering defects can also create additional types of PCB damage.

Acknowledgements

Thank you to all academic community Politeknik Negeri Padang, especially the Research and Community Service Center Politeknik Negeri Padang **for** funding this research.

Author Contributions

Efrizon: Methodology, Validation, Formal Analysis, Investigation, Writing – Original Draft, Writing – Review & Editing, Visualization; **Dara Aprila**: Conceptual, Resources, Supervision, Funding Acquisition; **Laxsmy Devy**: Methodology, Resources, Supervision; **Tri Artono**: Methodology, Investigation, Supervision, Project Administration; **Era Madona**: Resources.

Conflicts of Interest

The manuscript has not been published elsewhere and is not being considered by other journals. All authors have approved the review, agree with its Submission, and declare no conflict of interest in the manuscript.

5.0 References

Ahmad Agung, F., Herizon, H., Madona, E., Rohfadli, M., & Ja'far, J. (2023). Implementation of LiDAR Sensor for Mobile Robot Delivery Based on Robot Operating System. JECCOM: International Journal of Electronics Engineering and Applied Science, 1(2), 67–79. https://doi.org/10.30630/jeccom.1.2.67-79.2023

Al, A. B. D., Abu, R. M., & Mousavi, A. (2020). A Review and Analysis of Automatic Optical Inspection and Quality Monitoring Methods in Electronics Industry. 8.

Buang, A. S., Abu Bakar, M. S., & Rohani, M. Z. (2024). A Review Of Trend Advanced Welding Process And Welding Technology In Industries . International Journal Of Technical Vocational And Engineering Technology, 5(1), 133-145.

Dovbnych, M., & Plechawska–Wójcik, M. (2021). A comparison of conventional and deep learning methods of image classification. Journal of Computer Sciences Institute, 21(September), 303–308. https://doi.org/10.35784/jcsi.2727

Li, J., Gu, J., Huang, Z., & Wen, J. (2019). Application research of improved YOLO V3 algorithm in PCB electronic component detection. Applied Sciences (Switzerland), 9(18). https://doi.org/10.3390/app9183750

Lim, D. U., Kim, Y. G., & Park, T. H. (2019). SMD Classification for Automated Optical Inspection Machine Using Convolution Neural Network. Proceedings - 3rd IEEE International Conference on Robotic Computing, IRC 2019, 395–398. https://doi.org/10.1109/IRC.2019.00072

Quach, L. Da, Quoc, K. N., Quynh, A. N., & Ngoc, H. T. (2023). Evaluating the Effectiveness of YOLO Models in Different Sized Object Detection and Feature-Based Classification of Small Objects. Journal of Advances in Information Technology, 14(5), 907–917. https://doi.org/10.12720/jait.14.5.907-917

Susa, J. A. B., Mariquina, E., Tria, M. L., Adolfo, C. M., & Castro, J. C. (2020). Cap-Eye-citor: A Machine Vision Inference Approach of Capacitor Detection for PCB Automatic Optical Inspection. 7th IEEE International Conference on Engineering Technologies and Applied Sciences, ICETAS 2020, 5–9. https://doi.org/10.1109/ICETAS51660.2020.9484182

Tsai, Y. H., Lyu, N. Y., Jung, S. Y., Chang, K. H., Chang, J. Y., & Sun, C. T. (2019). Deep learning based aoi system with equivalent convolutional layers transformed from fully connected layers. IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM, 2019-July, 103–107. https://doi.org/10.1109/AIM.2019.8868602

Varna, D., & Abromavičius, V. (2022). A System for a Real-Time Electronic Component Detection and Classification on a Conveyor Belt. Applied Sciences (Switzerland), 12(11). https://doi.org/10.3390/app12115608

Wang, W. C., Chen, S. L., Chen, L. B., & Chang, W. J. (2017). A Machine Vision Based Automatic Optical Inspection System for Measuring Drilling Quality of Printed Circuit Boards. IEEE Access, 5(June), 10817–10833. https://doi.org/10.1109/ACCESS.2016.2631658

Zhang, K., & Shen, H. (2021). Solder joint defect detection in the connectors using improved faster-rcnn algorithm. Applied Sciences (Switzerland), 11(2), 1–15. https://doi.org/10.3390/app11020576