Design of printed circuit board detection with image information technology

Tibyani¹, N Hayati², I Sutrisno³, A Khumaidi¹, M A Rahman⁴, F Gumelar⁵, K Dhipanusa⁶, Filemon⁷ and E Supriyanto⁸
¹Faculty Of Computer Science, Brawijaya University, Indonesia
²,³,⁴Politeknik Perkapalan Negeri Surabaya, Indonesia
⁵,⁶,⁷Politeknik Pelayaran Sorong, Indonesia
⁸Politeknik Negeri Semarang, Indonesia

E-mail: tibyani@ub.ac.id

Abstract. In this research, an automatic PCB's defects checking tool will be made with image processing technology that adopts Haar Cascade Classifier method as a decision maker for whether or not a broken circuit on the PCB, with this technology the PCB will be detected using a square feature to show the broken path. will be displayed on the interface on the monitor. From the results of the test, data using the Haar Cascade Classifier method has the average percentage of successful detection of broken paths on the PCB at a camera distance of 26 cm and 16 cm is 52.49% with the maximum hassle obtained when the camera distance is placed at a distance of 26 cm and an angle position of 105 degrees which has a success rate of 97.5%. This method is good enough for PCB defect detection tools.

1. Introduction
Today there are still many printed circuit boards produced or in this book it will be abbreviated as PCB conventionally. PCB is a board used to connect electronic components with the layer of the conductor, usually copper. The process of making PCB is conventionally carried out in several basic stages, namely the design stage of the copper path shape and the PCB etching stage. The etching stage is the process of removing unwanted copper from the PCB board. One way to do this etching process is to use ferric chloride (FeCl₃) which is dissolved in hot water. During the etching stage, sometimes the etching process doesn't happen well. This can cause defects in the copper lines on the PCB, defects in the copper lines on this PCB can be in the form of broken copper lines or attached copper lines due to undesirable dissolving of the copper layer resulting in short conditions.

When used in the production of electronic equipment, PCBs are very influential in the manufacture of these electronic devices, for example when there are only a few PCB lines that are disconnected or damaged, the electronic device cannot be operated properly. So it is very important in the PCB Quality Check process to check whether the PCB is damaged or not. Usually, in PCB inspection, only direct checking is used in conventional ways.

In large scale PCB production especially for single layer PCB and conventional PCB printing methods. Checking the broken paths on the PCB using conventional methods cannot be done optimally. One of them is due to fatigue, decreased accuracy and lack of resources.[1] From the results of a survey conducted at a company that produces PCBs, it is known that PCB quality control is still done...
manually and sometimes human errors occur where the error in the percentage value reaches 10%. Therefore, we need a system that can perform PCB path checks quickly and precisely.

Therefore, in this study, the author tries to make and analyze a PCB defect checker with the help of a camera to replace human vision and to make it easier to apply this PCB checker, a technology called MiniPC or Raspberry Pi is used. With these two technologies, Image Processing can be used to detect objects with the OpenCV and Tensorflow libraries. According to Hanugra Aulia Sidharta, OpenCV (Open Source Computer Vision Library) is an open source library developed by Intel that focuses on simplifying programming related to digital images (Sidharta, 2017). From the use of the OpenCV library, images can be obtained in real time.

With the above technology, the authors will conduct research to make a PCB defect detector with the help of Image Processing with the Haar Cascade method to help determine broken paths on the PCB.

2. Methodology

2.1. System Design

From Figure 1 the systems work first initial in all the system make sure all the hardware connection can work properly. secondly make sure camera censor can work well for catching image and already installed in raspberry Pi after that the image that already taken by camera will be processed by image processing using OpenCV and all the library that already installed such tensor flow, PIL, and tesseract. PCB’s image will be processed by Haar Cascade Classifier Method using the datasets that already train before. If there are defects on the PCB’s Circuit as dataset then the defect’s will be shown in the Interface.

![Figure 1. System design.](attachment:image.png)
2.2. PCB’s Defect Detector Design

![Image](image_url1)

**Figure 2.** PCB’s defect detector design.

Figure 2 is the hardware design will be used on this research, it has 35 cm tall and there are 3 lines that have 10 cm distance each, these lines use to put the camera censor. And there is a table with 30 cm width and 40 cm long, this table use to put the PCB that will be detected. And there is small surface beside the big table it used to put Raspberry Pi.

2.3. Interfacing Software Design

At this stage what the author does is to use the Thonny software which is used to create views and input coding. Thonny software is used for coding assignments and running programs, as well as for system interfacing. In addition, the Thonny software imports several libraries that are used.

![Image](image_url2)

**Figure 3.** Interfacing Design
On this interfacing Design There are two windows, small windows and Big windows. The small window used to show the real time image from the PCB. The bigger window is shown PCB detection result as a picture, and capture button use to take the image of realtime PCB. Search button use to find the dataset that can be load to test the software work properly or not. And then hasil deteksi or detection result label use to show how many Detected defect circuit from the PCB.

3. Result and Discussion

3.1. Camera Testing Result
Camera is the main censor in this research especially for taking PCB’s real time image. Author already testing the camera that already installed and programmed in to the Raspberry Pi.

![Figure 4. Camera testing.](image)

3.2. Haar- Like Feature
This test was done by selecting the haar-like feature in to square, in this case the selecting feature are PCB’s broken circuits. The first step in this test was changing the RGB image from PCB into Grayscale image so the system can deciding broken circuit feature.

![Figure 5. Conversion from RGB image to grayscale](image)
After converting image to Grayscale image, the next step is to choose Haar-Like feature. Haar-Like Feature processing image into squares, with in one square there are some pixels. Than every square will be count and produce result a threshold, threshold is the difference that show dark and light part. Those value that will be processed on image processing.

![Figure 6. Haar-like feature](image_url)

### 3.3. PCB’s A Defect Detection

In this test, the test will be carried out with a distance of 26 cm, 16 cm, and 6 cm, the distance which is the distance between the camera and the table where the PCB is placed, with each distance set in the camera angle of 60°, 90° and 105°. And before that author want to test if the system can detected a 0.5 mm broken circuit width or not, the result will be shown on table 1.

**Table 1. Broken circuit with 0.5 and 1.5 mm width**

| No | Lebar Jalan | Pengacuan konvensional | Pengacuan menggunakan model Haar Cascade Classifier |
|----|-------------|------------------------|---------------------------------------------------|
| 1  | 1,5 mm      |                        |                                                    |
| 2  | 0,5 mm      |                        |                                                    |

From Table 1 one we can conclude that this tools can work as expected, after this PCB A will be tested in camera distance 26, 16, and 6 Cm with camera declivity angle 160°, 90° and 105°. The result show in table 2.
Table 2. PCB A Detection Result

| Distance | 105° | 90°  | 60°  | Average |
|----------|------|------|------|---------|
| 26 cm    | 97.5%| 72.5%| 52.5%| 74.16%  |
| 16 cm    | 65%  | 25%  | 25%  | 38.33%  |
| 6 cm     | Not detected | Not detected | Not detected | Not detected |

52.49%

3.4. PCB’s B Defect Detection

In this test, the test will be carried out with a distance of 26 cm, 16 cm and 6 cm, with each distance set in the angle of 60°, 90° and 105°. In PCB B this time the system can be considered running 100% when there is no PCB detection with broken paths.

Table 3. PCB B Detection Result

| Distance | 105°  | 90°  | 60°  | Average |
|----------|-------|------|------|---------|
| 26 cm    | 90%   | 90%  | 80%  | 86.67%  |
| 16 cm    | 97.5% | 92.5%| 95%  | 95%     |
| 6 cm     | Not detected | Not detected | Not detected | Not detected |

90.83%

4. Conclusion

The success rate of the system is influenced by the distance of the camera and also the position of the camera angle when taking images. It can be seen from each percentage that the average detection success rate is 97.5% for a distance of 26 cm, 38.83% for a distance of 16 cm, and cannot be detected at a distance of 6 cm. this is because the closer the camera position to the PCB, the narrower the field that can be taken and detected.

5. References

[1] Sulistyo dkk 2012 Rancang Bangun Prototipe Aplikasi Pengenalan Wajah untuk Sistem Bnsensi Alternatif dengan Metode Integral Feature dan Eigenface
[2] Pradhan A and Prayudi, Y 2008 Penggunaan Template Matching untuk Identifikasi Kecacatan pada PCB
[3] Sidharta H A 2017 Introduction to Open CV (Ilmu Komunikasi: Binus Malang, Oktober)

[4] Ahmad H 2010 Haar Cascade Classifier.
https://www.researchgate.net/figure/HaarCascadeClassifier_fig1_330553730

[5] Jati 2015 Deteksi obek menggunakan Haar Cascade Classifier implemented.
https://jati.stta.ac.id/2015/09/deteksi-obek-menggunakan-haar-cascade.html

[6] lcdwiki.com 2019 3.5 inch Raspi display http://www.lcdwiki.com/3.5inch_RPi_Display

[7] Fadilla N 2017 Alat keselamatan pengendara dari kondisi telah dilengkapi dengan multi level safety berbasis pengolahan citra dengan metode Haar Cascade Classifier (Tugas Akhir PPNS)

[8] Raspberrypi.org 2018 raspberrypi. [Available at: https://www.raspberrypi.org] Kementrian ESDM: Data Tumpahan Minyak 2017. [Online] Available: http://statistik.migas.esdm.go.id/index.php?r=dataTumpahanMinyak/index. [Accessed: 26-Jan-2020]

[9] Jami’in M A, Hu J, Marhaban M H, Sutrisno I and Mariun N B 2016 Quasi ARX neural network based adaptive predictive control for nonlinear systems. IEEE Trans Elec Electron Eng 11 pp 83-90

[10] Jami’in M A, Sutrisno I and Hu J 2012 Lyapunov learning algorithm for Quasi-ARX neural network to identification of nonlinear dynamical system. IEEE International Conference on Systems, Man, and Cybernetics (SMC) (Seoul, 2012) pp 3147-3152 doi: 10.1109/ICSMC.2012.6378275

[11] Jami’in M A, Sutrisno I and Hu J 2013 Deep searching for parameter estimation of the linear time invariant (LTI) system by using Quasi-ARX neural network. The 2013 International Joint Conference on Neural Networks (IJCNN) (Dallas, TX) pp 1-5 doi: 10.1109/IJCNN.2013.6707091

[12] Jami’in M A, Sutrisno I and Hu J 2015 Maximum power tracking control for a wind energy conversion system based on a quasi ARX neural network model. IEEE Trans Elec Electron Eng 10 pp 368-375

[13] Jami’in M A, Sutrisno I, Hu J, Mariun N B and Marhaban M H 2014 An adaptive predictive control based on a quasi-ARX neural network model. 13th International Conference on Control Automation Robotics & Vision (ICARCV) (Singapore, 2014) pp 253-258 doi: 10.1109/ICARCV.2014.7064314

[14] Sutrisno I, Jami’in M A and Hu J 2014 Modified fuzzy adaptive controller applied to nonlinear systems modeled under quasi-ARX neural network. Artif Life Robotics 19 pp 22–26 (2014). https://doi.org/10.1007/s10015-013-0137-6

[15] Sutrisno I, Jami’in M A and Hu J 2014 An improved elman neural network controller based on quasi ARX neural network for nonlinear systems. IEEE Trans Elec Electron Eng 9 pp 494-501

[16] Sutrisno I, Che C and Hu J 2014 An improved adaptive switching control based on quasi-ARX neural network for nonlinear systems. Artif Life Robotics 19 pp 347–353 (2014)

[17] Sutrisno I, Jami’in M A and Hu J 2012 Neural predictive controller of nonlinear systems based on quasi-ARX neural network. 18th International Conference on Automation and Computing (ICAC) (Loughborough, 2012) pp 1-6

[18] Rinanto N and Khumaidi Aplikasi Android- Raspberyy Pi pada Kapal Tanpa Awak untuk Pencarian Korban Kecelakaan Laut

[19] Widodo R B et al Grasping and attached mode in human computer interaction in the study of mouse substitution. Journal of Physics: Conference Series 1196 1 012034

[20] Sutrisno I, Jami’in M A, Hu J, Marhaban M H 2015 Self-organizing quasi-linear ARX RBFN modeling for identification and control of nonlinear systems. 54th annual conference of the Society of Instrument and Control Engineers of Japan (SICE). IEEE

[21] Jami’in M A, Sutrisno I and Hu J 2014 Nonlinear adaptive control for wind energy conversion systems based on quasi-ARX neural networks model. Proceedings of the International Multi
Conference of Engineers and Computer Scientists March 12-14, 2014 (Hong Kong, 2014)

[22] Sutrisno I 2009 *Pemrograman Komputer Dengan Software Matlab*. (Surabaya: ITS Press)

[23] Rinanto N and Khumaidi *Aplikasi Android-Raspberry Pi pada Kapal Tanpa Awak untuk Pen- carian Korban Kecelakaan Laut*