Testing the performance of a single pole detection algorithm using the confusion matrix model

M Yusrö1,*, E Suryana1, K Ramli2, D Sudiana2 and K M Hou3

1 Universitas Negeri Jakarta, Jakarta, Indonesia
2 Universitas Indonesia, Depok, Indonesia
3 Université Clermont Auvergne, Clermont-Ferrand, France

*myusro@unj.ac.id

Abstract. This study aims to examine the single pole detection algorithm using a confusion matrix model which is a specific table that makes it easy to visualize the performance of an algorithm. The algorithm tested is the YuRHoS pole detection algorithm, a new algorithm developed by researchers to detect pole objects. Methods used are by calculating three aspects of algorithm performance in machine learning, namely sensitivity, specificity, and accuracy. The value of the three aspects of performance depends on four variables, namely true positive, true negative, false positive and false negative. The calculation process is done by matching the pixel detection region with the ground-truth region. The test results for 4 (four) different single pole images found that the YuRHoS pole detection algorithm is better than other algorithms on two measurement aspects, namely specificity and accuracy. Excellence aspects of specificity obtained because of its ability in detecting the object instead of a pole. Excellence aspects of accuracy indicated because more accurate in detecting a pole. As for sensitivity aspects, both the detection algorithms are having the same reliability in correctly predicting a pole.

1. Introduction
Related studies on detection of object had been made by several researchers, but none of the researches especially discussed on pole detection. In 2006, there was a research to detect straight lines with various orientation using Principal Component Analysis (PCA) [1]. In 2014 a research to detect pole objects such as traffic light, traffic sign posts and street lamp post based on Mobile Laser Scanning Cloud had been made. The Mobile Laser Scanning Cloud Mobile Laser Scanning Cloud is derived from Light Detection and Ranging (LiDAR) [2]. The purpose of this study was to develop a new algorithm to improve accuracy in detecting poles. The Algorithm developed in this research is called YuRHoS pole detection algorithm, which is taken from the researcher’s name (M. Yusro, K. Ramli, K.M. Hou and D. Sudiana) [3,4].

The development of the YuRHoS pole detection algorithm uses the optimization of vertical Canny edge line detection and Hough Transform. The Canny method is used to detect edge from a picture input [5,6]. In this method there is a process called Non-Maximum Suppression (NMS) which compares the pixel with the neighboring pixel based on pixel gradient (θ/theta). Optimizing the vertical line edge detection was made in the NMS stage, by having a compared pixel is the pixel which is in the θ = 0°
group, but the pixel which has a group 0 other than 0° will be ignored. The product of this process is vertical edge line to an image.

To test the accuracy of pole detection from the YuRHoS algorithm, the YuRHoS algorithm will be compared with a similar pole detection algorithm, hereinafter referred to as the Non-YuRHoS algorithm. In this experiment, the basic assumption is that the definition of a pole is 2 (two) vertical lines parallel to almost the same length. For the vertical line identification process, Hough transformation or HoughLines function is used in the OpenCV Library. To make this happen, the processing function of cvHoughLines2 is used in OpenCV [7]. In this study, the comparison of the results of the second detection of the pole detection lagoon was tested through simulation using Eclipse software [8].

2. Methods

To calculate the algorithm performance in machine learning, there are three aspects of measurement that can be used (the confusion matrix model) [9,10]; sensitivity, specificity and accuracy. Sensitivity (true positive level) which measures the positive proportion correctly identified. Specificity (true negative level) which measures the negative proportion correctly identified. Accuracy refers to the closeness of value measured with standard values or known values. Whereas, the formulation of the three aspects is [11]:

\[
\text{Sensitivity (sen)} = \frac{TP}{TP + FN} \\
\text{Specificity (spc)} = \frac{TN}{TN + FP} \\
\text{Accuracy (acc)} = \frac{TP + TN}{TP + FN + FP + TN}
\]

The evaluation of the simulation of both algorithms in pole-detection is obtained by using sensitivity, specificity, and accuracy score [12]. In this research, sensitivity is the ability of algorithm in predicting the correct result; specificity is the ability of algorithm in detecting the correct result by diverting false result; and accuracy is the amount of correct results compared to existing population. The procedure of algorithm simulation in the detection of poles of Non-YuRHoS and YuRHoS is explained in Table 1 below.

Table 1. The simulation procedure on poles.

| YuRHoS and Non-YuRHoS |
|-----------------------|
| 1. Expert judgment is used to classify the poles and non-poles in the image examined. |
| 2. The use of image template for TP score (True Positive), TN score (True Negative), FP score (False Positive) and FN score (False Negative) from the result in the detection of poles. |
| 3. The use of confusion matrix for scores of sensitivity, specificity and accuracy from the result in the detection of poles. |
| 4. To compare the result of simulation in the detection of YuRHoS and Non-YuRHoS. |

The simulation of the algorithm result of the detection of YuRHoS and Non YuRHoS starts with classifying objects in the image as pole object (positive object) and non-pole object (negative object). The determination of positive or negative objects (pole or non-pole) is based on expert judgment via image processing to classify objects from the image examined. In theory, expert judgment can be used in four functions [13,14], namely: a) to complete, validate, interpret and integrate existing data, to assess the impact of changes; b) to predict future incidents and consequences from decisions made; c) to determine the condition of knowledge in a field; d) to provide elements required for decision making between several acceptable alternatives.
3. Results and discussion

3.1. Labelling procedure by experts
In this simulation, the images examined are a one pole image and the image of poles. The algorithm execution resulted in markers for every pole detected/found, whereas, markers are able to identify poles correctly or incorrectly. To evaluate the result of the simulation automatically (using a program), therefore, each image is labelled (ground-truth) between pole objects (positive object) and non-pole objects (negative object). The labelling mechanism is obtained by using procedures as follow:

- For every image, the expert will determine which objects are positive and which are negative.
- Label every positive object (pole images). The positive objects are read one by one and then labelled with black colour, meanwhile, other images are in white colour. Every positive label is saved in a file named <name_file>_t<i>.png where i stands for truth (index starts from 1), in example, g1_t1.png (meaning g1=image examined number 1, t1=positive object number 1 or, true-1). After one positive object is labelled into a new image, other positive object is labelled and so forth until every positive object is read.
- Label every negative object (non-pole images). The characteristic of negative objects is having vertical edges. Negative objects are then read one by one, whereas, every negative object is labelled in black colour and other objects are in white colour. Every negative label is saved in a file named <name_file>_f<i>.png, in example, g1_f1.png (meaning, g1=image examined number 1, f1=negative object number 1 or, false-1). After one negative object is labelled into a new image, other negative objects are labelled and so forth until every negative object is read.

Below are examples of images examined and labelled (ground-truth) with positive and negative object (Table 2). Labelling is performed for every image, based on expert judgment.

| Table 2. Labelling process (ground-truth). |
|------------------------------------------|
| **Single Pole Image** | **Labeling (Gorund-Truth)** |
| ![Single Pole Image](image) | ![Labeling (Gorund-Truth)](image) |
| (1a) Positive (pole) | (1b) Negative-1 (left tree) |
| ![Negative-2 (billboard)](image) | ![Negative-3 (building)](image) |

3.2. The result of algorithm simulation on non-YuRHoS and YuRHoS
Figure 1 is showing four one-pole images which were tested using two different algorithms (Non-YuRHoS and YuRHoS).
Figure 1. Images for one-pole simulation.

The result on the simulation of one-pole images using algorithm in the detection of Non-YuRHoS and YuRHoS poles is presented in Table 4-7 below:

Table 3. The result of image simulation of single poles (Image-1).

| Image Examined | Result for Non YuRHoS | Result for YuRHoS |
|----------------|------------------------|-------------------|
|                | TP = 1, FN = 0, TN = 0, FP = 1 | TP = 1, FN = 0, TN = 1, FP = 0 |
| Expert judgment: one pole object and one non-pole object (street sign) | - TP = 1; one pole detected as pole. | - TP = 1; one pole detected as pole. |
| | - FN = 0; no pole detected as pole. | - FN = 0; no pole not detected as pole. |
| | - TN = 0; no non-pole object detected as non-pole. | - TN = 1; one non-pole object not detected as non-pole. |
| | - FP = 1; one non-pole object detected as pole (street sign). | - FP = 0; no non-pole object detected as pole. |

\[
\text{Sen} = \frac{TP}{TP+FN} = 1, \quad \text{Spc} = \frac{TN}{TN+FP} = 0, \quad \text{Acc} = \frac{TP+TN}{TP+FP+TN+FN} = 0.5
\]

Conclusions derived from the Simulation:

- YuRHoS Algorithm has a better specificity (score 1) compared to Non-YuRHoS Algorithm (score 0). YuRHoS Algorithm is capable of detecting the correct result (pole) and able to detect the incorrect result (non-pole).
- YuRHoS Algorithm is better in accuracy (score 1) in comparison to Non-YuRHoS Algorithm (score 0.5). YuRHoS Algorithm is capable of detecting the correct result (pole) more out than of every object detected.
Table 4. The result of image simulation of single poles (Image-2).

| Image Examined | Result for Non YuRHoS | Result for YuRHoS |
|----------------|------------------------|-------------------|
|                | TP = 1, FN = 0, TN = 3, FP = 1 | TP = 1, FN = 0, TN = 4, FP = 0 |
| Expert judgment: one pole object and four non-pole objects (three street signs and one tree) | - TP = 1; one pole detected as pole. - FN = 0; no pole not detected as pole. - TN = 3; three non-pole objects detected as non-pole. - FP = 1; one non-pole object detected as pole (top street sign). | - TP = 1; one pole detected as pole. - FN = 0; no pole not detected as pole. - TN = 4; four non-pole objects detected as pole. - FP = 0; no non-pole object detected as pole. |
| Sen = \( \frac{TP}{TP+FN} = 1 \) | Sen = \( \frac{TP}{TP+FN} = 1 \) |
| Spc = \( \frac{TN}{TN+FP} = 0.75 \) | Spc = \( \frac{TN}{TN+FP} = 1 \) |
| Acc = \( \frac{TP+TN}{TP+FN+TN+FP} = 0.8 \) | Acc = \( \frac{TP+TN}{TP+FN+TN+FP} = 1 \) |

Conclusions derived from the simulation:
- YuRHoS Algorithm has a better specificity (score 1) compared to Non-YuRHoS Algorithm (score 0.75). YuRHoS Algorithm is capable of detecting the correct result (pole) and the incorrect ones (non-pole).
- YuRHoS Algorithm has a better accuracy (score 1) in comparison to Non-YuRHoS Algorithm (score 0.8). YuRHoS Algorithm is capable of detecting the correct result (pole) more out of all the objects detected.

Table 5. The result of image simulation of single poles (Image-3).

| Image Examined | Result for Non YuRHoS | Result for YuRHoS |
|----------------|------------------------|-------------------|
|                | TP = 1, FN = 0, TN = 2, FP = 0 | TP = 1, FN = 0, TN = 2, FP = 0 |
| Expert judgment: one pole object and two non-pole objects (two street signs) | - TP = 1; one pole detected as pole. - FN = 0; no pole not detected as pole. - TN = 2; two non-pole objects detected as non-pole. - FP = 0; no non-pole object detected as pole. | - TP = 1; one pole detected as pole. - FN = 0; no pole not detected as pole. - TN = 2; two non-pole objects detected as non-pole. - FP = 0; no non-pole object detected as pole. |
Table 5. Cont.

\[
\begin{align*}
\text{Sen} &= \frac{TP}{TP + FN} = 1 \\
\text{Spc} &= \frac{TN}{TN + FP} = 1 \\
\text{Acc} &= \frac{TP + TN}{TP + TN + FP + FN} = 1
\end{align*}
\]

Conclusion derived from the Simulation:
- Both Algorithms (YurHoS and Non-YuRHoS) have the same capabilities in detecting poles.

Table 6. The result of image simulation of single poles (Image-4).

| Image Examined | Result for Non YuRHoS | Result for YuRHoS |
|----------------|------------------------|-------------------|
| ![Image](image-url) | ![Image](image-url) | ![Image](image-url) |

- TP = 1; FN = 0, TN = 0, FP = 1
- TP = 1, FN = 0, TN = 1, FP = 0

Expert Judgment: one pole object and one non-pole object (traffic light)
- TP = 1; one pole detected as pole.
- FN = 0; no pole not detected as pole.
- TN = 0; no non-pole object detected as non-pole.
- FP = 1; one non-pole object detected as pole (traffic light).

\[
\begin{align*}
\text{Sen} &= \frac{TP}{TP + FN} = 1 \\
\text{Spc} &= \frac{TN}{TN + FP} = 0 \\
\text{Acc} &= \frac{TP + TN}{TP + TN + FP + FN} = 0.5
\end{align*}
\]

\[
\begin{align*}
\text{Sen} &= \frac{TP}{TP + FN} = 1 \\
\text{Spc} &= \frac{TN}{TN + FP} = 1 \\
\text{Acc} &= \frac{TP + TN}{TP + TN + FP + FN} = 1
\end{align*}
\]

Conclusions derived from the Simulation:
- YuRHoS Algorithm has a better specificity (score 1) compared to Non-YuRHoS Algorithm (score 0). YuRHoS Algorithm is capable of detecting the correct result (pole) and incorrect ones (non-pole).
- YuRHoS Algorithm is better in accuracy (score 1) in comparison to Non-YuRHoS Algorithm (score 0.5). YuRHoS Algorithm is more capable of detecting the correct result (pole) out of all the objects detected.

4. Conclusion
From the general conclusions obtained from four pole images simulated, it is concluded that pole-detection using YuRHoS Algorithm is better compared to Non YuRHoS Algorithm in two aspects of measurement, specificity and accuracy. The superiority in specificity indicates that YuRHoS Algorithm is better in detecting pole objects and is capable of differentiating non-pole objects (non-poles).

The superiority of accuracy indicates that YuRHoS Algorithm is more accurate in detecting pole objects compared to Non YuRHoS Algorithm. Whereas, for sensitivity aspect, both algorithms have equal superiorities in predicting pole objects correctly.

Acknowledgments
This work was supported by the Indonesian and French Governments in Joint Program for Research and Higher Education. Authors wished to acknowledge Prof. Edwige Pissaloux and Mr. Imam Firmansyah.
for the fruitful discussions and fabulous advices during completing this research. The authors would like to thank also for some anonymous AASEC 2019 reviewers for their constructive comments and insightful suggestions that improved the quality of this paper.

References

[1] Lee Y-S, Koo H-S, Jeong C-S 2006 A straight line detection using principal component analysis Pattern Recognit Lett. 27 14 1744–1754

[2] Cabo C, Ordoñez C, Garcia-Cortés S and Martinez J 2014 An algorithm for automatic detection of pole-like street furniture objects from Mobile Laser Scanner point clouds ISPRS J Photogramm Remote Sens 87 47–56

[3] Yusro M, Hou K M, Pissaloux E, Ramli K, Sudiana D and Zhang L Z 2014 Concept and Design of SEES (Smart Environment Explorer Stick) for Visually Impaired Person Mobility Assistance Adv Intell Syst Comput. 300 245–259

[4] Yusro M, Ramli K, Hou K M and Sudiana D 2018 Development and performance analysis of Canny and Hough transform based pole detection algorithm as mobility aid for visually impaired person on SEE-S (Smart Environment Explorer-Stick) J Eng Appl Sci. 13 7

[5] Rabbani M A and Chellappan C 2007 A Fast and New Approach to Gradient Edge Detection Int J Soft Comput 2 325–330

[6] Canny J 1986 A Computational Approach to Edge Detection IEEE Pattern Anal Mach Intell Trans. PAMI 8 6

[7] Week M D Algorithms for Image Processing and Computer Vision [ Kindle Edition ] From the Back Cover What Other Items Do Customers Bu After Viewing This Item 1–5

[8] Liao C 2014 Development of a Navigation System Using Smartphone and Bluetooth Technologies to Help the Visually Impaired to Navigate Work Zones Safely

[9] Powers D M W 2011 Evaluation: From Precision, Recall and F-Measure To Roc, Informedness, Markedness & Correlation J Mach Learn Technol 2 1 37–63

[10] Alsoos M and Joukhadar A 2014 Posture Independent Model for Hand Detection and Tracking Adv Intell Syst Comput. 300 301–312

[11] Przystup P, Bujnowski A, Rumiński J and Wtorek J 2013 A multisensor detector of a sleep apnea for using at home 2013 6th Int Conf Hum Syst Interact HIS 513–517

[12] Fawcett T 2006 An Introduction to ROC Analysis Pattern Recognit 27 861–874

[13] Chabrier S, Rosenberger C, Emile B and Laurent H 2008 Optimization Based Image Segmentation by Genetic Algorithms Image (Rochester, NY) 1–23

[14] Lannoy A and Procaccia H 2014 Expertise, safety, reliability, and decision making: practical industrial experience Environ Syst Decis 34 2 259–276