IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 12, No. 3, September 2023, pp. 1070~1082
ISSN: 2252-8938, DOI: 10.11591/ijai.v12.i3.pp1070-1082

Vehicle detection system based on shape, color, and time-motion

Afritha Amelia[1], Muhammad Zarlis[2], Suherman[3], Syahril Efendi[1]
[1]Department of Computer Science and Information Technology, Universitas Sumatera Utara, Medan, Indonesia
[2]Information Systems Management Program, Binus University, Jakarta, Indonesia
[3]Department of Electrical Engineering, Universitas Sumatera Utara, Medan, Indonesia

Article Info

Article history:

Received Feb 8, 2022
Revised Nov 2, 2022
Accepted Dec 21, 2022

Keywords:

Disparity
Image depth
Path error
Phase detection auto focus
Vehicle detection and tracking

ABSTRACT

Vehicle detection application can assist in-vehicle surveillance functions and have implications for various fields. A vehicle can be identified through the license number attached to its license plate, the color and its shape. Vehicle detection can make use of multimedia sensors so that the design and detection performances can be optimal. Sensor performances are influenced by factors such as the number of multimedia sensors, sensor placement, sensor positioning, and schemes in case of system failure. This study makes use of multimedia sensors with cameras equipped by a phase detection auto focus (PDAF) technology which is like a pair of eyes to see an object. This study analyses 134 vehicles with number detection and various colors to see the effect on the detection and recognition processes. The cars were passed through the camera 10 times at a speed of 10-15 km/hour with various camera distances and positions. Various values and depths of the images were generated. The farther the distance the higher the disparity values. For maximum distance of 50 m, disparity is 6.20×106 and image depth is 16.88×109. Vehicle color influences detection with orange has the best accuracy, but the gray has the largest path error value.

Corresponding Author:

Suherman
Department of Electrical Engineering, Universitas Sumatera Utara
Medan, Indonesia
Email: suherman@usu.ac.id

1. INTRODUCTION

Target detection and tracking is one of the important applications on the multimedia wireless sensor network (MWSN) network which can be used for missing vehicle tracking, determining vehicle location, calculating the amount of fuel [1], [2], tracking assets and object [3], war sites monitoring [4], illegal immigrants detection at the border [5], gas leak detection [6], forest fire detection [7], and wildlife tracking [8]. Target recognition and target search algorithms on MWSN network require sensor nodes that can detect targets according to the direction of their movement and manage the information centrally. Target detection and tracking algorithms are expected to have energy efficient concern and to be able to track multiple objects. Therefore, the MWSN nodes in object detection, specially vehicle detection should be placed and positioned at a certain direction at a certain time on the side of the vehicle way and able to work on heavy traffic. This is important for tracking targets and predicting the target's presence. So that the target search capability will be faster and efficient so that the information management on traffic controller can efficiently manage sensors to carry out tasks simultaneously which leads to a more optimal tracking process.

A vehicle can be identified by its color and the license number for security, odd-even traffic control, and electronic payment systems for toll and parking payments [9], [10]. Wavelet decomposition and convolutional neural network (CNN) algorithms recently are used for the detection and recognition of vehicle plate numbers and colors and produce an accuracy up to 99.43% for vehicle location determination and 98.9% for vehicle plate number recognition [9]. Image enhancement algorithm combined with CNN have been introduced in vehicle license number detection in Indonesia by accuracy of 99%. The growing popular you only look once (YOLO) algorithm is easily implemented for object detection algorithm with comparable 97.82% accuracy for plate detection accuracy and 96% for character recognition accuracy [11]. In other case, non-standard license plate detection has been implemented in Pakistan. Private number plates should be recognized and monitored for several purposes including security as well as a well-developed traffic system [12]. Other algorithms such as single shot detection (SSD) algorithm can classify and detect real-time objects precisely. This algorithm can train itself by using 1,524 images of motorcycle license plates and then, the characters of plate numbers are extracted by using the optical character recognition (OCR) module [13]. While the image processing algorithm used aims to detect and track vehicles based on the color and shape of the vehicle features with statistical models [14], the target tracking via color information can also be processed by using image processing algorithm [15].

The system for detecting and recognizing vehicle plate numbers and vehicle colors uses multimedia sensor equipped by a camera. These sensors have different capabilities and different prices to achieve the desired role and performance [16]. Selection of the type/type of sensor device and placement of multimedia sensors have an important role on the highway, which is used as a medium for taking pictures of detecting plate numbers and vehicle colors. The detection work process can be done by knowing whether the color of the vehicle can affect the detection and tracking of the vehicle plate number. For this reason, it is necessary to consider the optimal number of multimedia sensors to be used and the positioning of the multimedia sensors if the solution occurs in the event of a system failure [17], [18]. The methodology and achievement of a performance of this multimedia sensor installation are for visual information collection, identification, and tracking of vehicle plate numbers, and the effect of vehicle color on the recognition and tracking of vehicle plate numbers [12].

This research makes use MWSN sensor equipped by the camera with phase detection auto focus (PDAF) technology, namely autofocus technology that can adjust images based on one object phase. This technology is similar to a pair of right and left eyes seeing an object [19]. The sensor network contains 4 sensor with 3/32 GB RAM and memory specifications, 5.0 inch IPS screen, 1080×1920 pixels, Android OS Marshmallow 6.0.1, Snapdragon 625 chipset (14 nm), 13 MP Dual LED flash camera, and battery: Li-Po 4100 mAh. Sensors are mounted at a height of 1 m, 2 m, 3 m, and 4 m using poles.

The object of taking pictures to detect vehicle plates is in the form of 2 cars with white plate numbers BK 1801 ZD and black BK 1272 OY. The two cars will pass the camera sensor at a speed of 10 to 15 km/h 10 times. To collect data on the effect of the vehicle color, 132 cars objects with various colors were used. The cars passed camera sensors at a speed of 10 to 15 km/h from 12 p.m. to 1 p.m. when traffic was heavy. The impact of this research is the placement of multimedia sensors on the vehicle plate number detection system and the effect of vehicle color on the detection process. Optimal sensor placement will have the effect of optimal image capture in the form of vehicle plate numbers and vehicle colors so that the data reading time will be optimal as well.

2. THE ALGORITHMS

2.1. Deep learning

Deep learning is a sub-section of the scientific field of machine learning [20]–[22]. This field teaches a computers be able to perform as humans do. In deep learning, a computer can learn to classify images, sounds, text, or videos [23], [24]. In processing, a computer is trained to use a sufficiently large number of labeled data sets, and then it can convert pixel values of an image into an internal representation or feature map. Feature maps can be used to detect or classify patterns in input.

The deep learning method approaches data in two sessions, namely training and testing [25]–[27]. In a training sessions, we learn feature extraction from each data then distinguish one label from another. In the testing session, tested data can be analyzed from the results of a training session. One example of an algorithm in deep learning is a CNN [28] and YOLO [29].

2.2. Convolutional neural network

A CNN is one of the deep learning algorithms. CNN is designed to process image data. CNN is aimed to classify data labels using by supervised learning method. CNN is limited to processing only grid structures, such as digital images and videos. CNN has several parts, such as convolution layer, activation function rectified linear unit (ReLU), pooling layer, fully connected layer, and classification, and detection output [30]–[32].

The convolution layer consists of neurons are arranged by a kernel filter according to the length and height of pixels [33]–[35]. Convolution layer is the most important layer in CNN. The convolution is an operation of linear algebra which multiplies the filter matrix. The pooling layer is done after convolution layer processing. The pooling process is reducing the number of parameters by down-sampling operation [36], [37]. The kind of pooling layer algorithm is max pooling. In max pooling, the matrix value taken is a maximum value. The pooling layer can reduce the size of a volume, the number of parameters, and calculations. After the max-pooling process, then the layer will be entered into a fully connected layer. This layer connects all neurons. At this stage, an object can be classified and detected whether that object is appropriate or not.

2.3. YOLOV5

YOLOv5 is the fifth-generation object detection model that was released in April 2020 [38]. In general, the architecture of this model is not much different from the previous YOLO generations. YOLOv5 is written in the python programming language. YOLOv5 is not C language in previous versions. It makes installation and integration into internet og things (IoT) devices more easier. YOLOv5 also uses CNN algorithm. In addition, the PyTorch community is also larger than the darknet community. It means the PyTorch will get more contribution in the future.

3. RESEARCH METHOD

This study uses the research and development (R&D) method in [39] as a program for the detection, and recognition of vehicle plates number using original images taken from camera sensors. The flowchart is used to make it easier to understand the steps in the research as shown in Figure 1. Figure 1 shows the research preparation in the form of procurement of vehicles, camera sensors, camera support poles, and other supporting staff. At the data collection stage, the activities are in the form of placing camera sensors on the roadside, driving a vehicle/car at a speed of 10-15 km/h, and sensors detecting vehicle number plates. The results of image detection as input for the software used and the output of the software in the form of image capture results in the condition of the vehicle plate being visible and not visible.

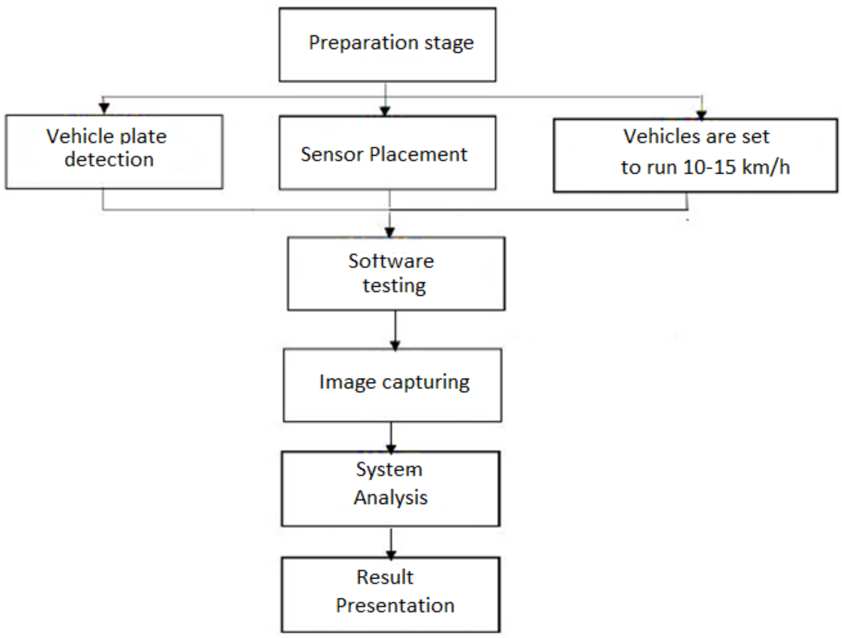


Figure 1. Research flowchart

This study uses materials that support object detection and recognition methods using block diagrams which can be seen in Figure 2. Research was conducted in both field experiment (1 and 2) and computer processing (3 and 4). Figure 2 can be explained.

i) The camera position is set at a height of 1 m, 2 m, 3 m, and 4 m. To collect car plate detection data with plate numbers "BK1801ZD" and "BK1272OY" by way of the car passing through the camera at a speed of 10-15 km/h. Both cars passed the sensor 10 times so that 10 frames were taken at each altitude. A total of 40 frames were captured on the “BK 1801 ZD” and 40 frames for the “BK 1272 OY”.

ii) The camera position can be seen in Figure 3. For car color detection data retrieval, carried out during the day around 12.00-13.00 WIB where traffic is very heavy and vehicle speed is around 10-15 km/h. The total sample taken was 132 cars with various colors, namely gray 11 cars, white 41 cars, silver 19 cars, black 53 cars, red 4 cars, brown 1 car, yellow 1 car, blue 1 car, and orange 1 car.
iii) The output of these activities is a video recording of the vehicles that are the target of detection and recognition.

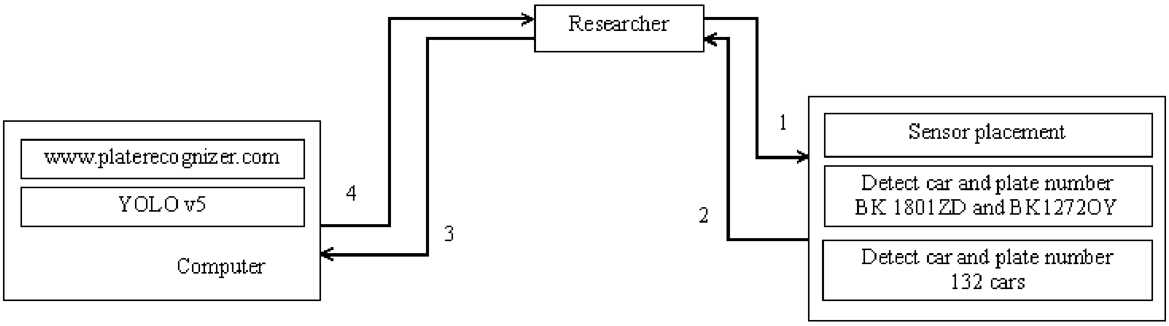


Figure 2. Block diagram of research

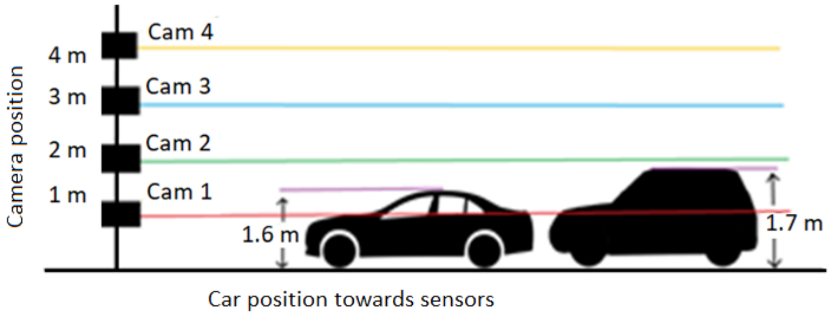


Figure 3. Camera placement toward the vehicle

3.1. Hardware

This study uses MWSN sensor network with 4 sensors with specifications:

- Lens baseline = 28.3 mm
- Focal length = 3.7 mm
- Horizontal field of view = 65.0º
- Vertical field of view = 50.7º
- Sensor medium = 4.7×3.5 mm
- Physical sensor size = 1/3 inch
- Pitch size = 16.2 mm^2
- Maximum picture resolution = 4160×3120
- Resolutions = 13.0 MP

3.2. Software

Vehicle plate detection using the online application www.platerecognizer.com as shown in Figure 4. The detection of vehicle plates is used by the hardware in the form of 2 vehicle objects to detect the black car Honda Mobilio plate no. BK1272OY and the white car Toyota Avanza plate no. BK1801ZD. The system will detect the license plate number and color of the two cars. Dimensions required for Honda Mobilio dimensions are 4,386-4,398 mm long×1,683 mm wide×1,603 mm high [40]. The dimensions of the Toyota Avanza are 4,395 mm long×1,730 mm wide×1,665-1700 mm high [41]. Detection of the vehicle color, used a sample of 132 cars with various colors, namely gray 11 cars, white 41 cars, silver 19 cars, black 53 cars, red 4 cars, brown 1 car, yellow 1 car, blue 1 car, and orange 1 car.

The software used for object detection is using YOLOv5. The placement of the shooting position of object detection is by Figure 3. YOLOv5 is consist of,
– www.github.com/ultralytics/yolov5 for download YOLOv5 sourcecode.
– Python 3.10.1
– CUDA 11.3
– CUDNN 8.4
– Pytorch framework
– Visual studio 2022
– Anaconda powershell prompt

All items must be installed in computer and using the anaconda prompt types "python detect.py –source data/images/frame_1.png", then enter as seen in Figure 5. If the file before detection is taken from the data subfolder then the detection results can be seen in the runs subfolder as shown in Figures 6 and 7.

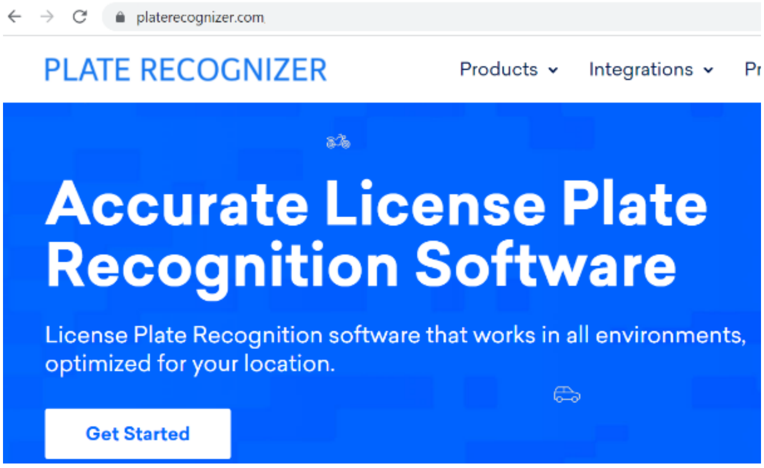


Figure 4. Appearance of platerecognizer.com

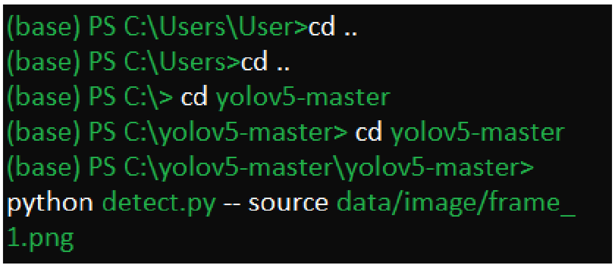


Figure 5. Appearance of anaconda prompt

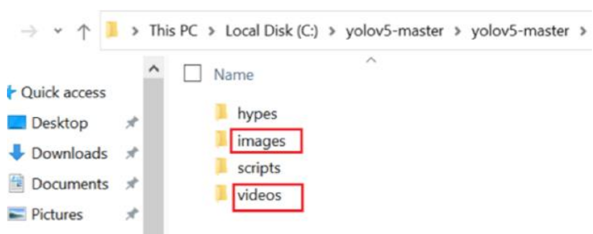


Figure 6. Input subfolder

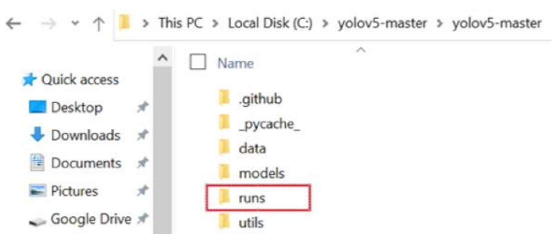
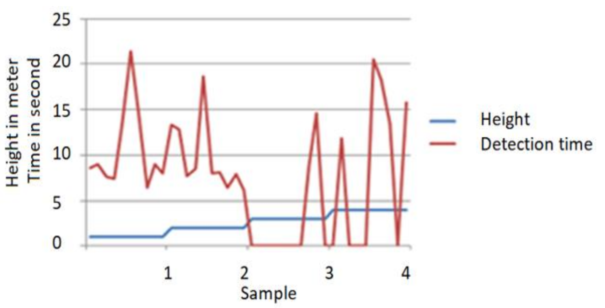


Figure 7. Output subfolder

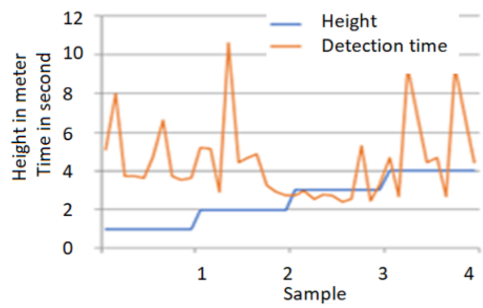
4. RESULTS AND DISCUSSION

4.1. Results

We used the online application www.platerecognizer.com to detect plate numbers BK1272OY and BK1801ZD. Detection time for BK1272OY is shown in Figure 8(a) and detection time for BK1801ZD is shown in Figure 8(b). The generated accuracy for both figures is shown in Table 1. Table 1(a) shows that BK1801ZD was detected in sensor positions of 1, 3, and 4. Table 1(b) shows that BK1272OY was detected in all positions.



(a)



(b)

Figure 8. Detection time for (a) BK1801ZD and (b) BK1272OY

Table 1. Accuration result for (a) BK1801ZD and (b) BK1272OY

Plate number detected	Height (m) 1	2	3	4	Total of frames
	Number of frames	Number of frames	Number of frames	Number of frames	
(a) BK1801ZD					
B1801ZD	5	1	-	-	6
BK1801ZD	2	4	-	1	7
B1301Z0	1	-	-	-	1
BC1801ZD	1	-	-	-	1
BK1801Z0	1	-	-	-	1
B1801Z0	-	1	-	-	1
BH1901	-	1	-	-	1
BZ1801ZD	-	1	-	-	1
BH1801ZD	-	1	-	-	1
41891	-	1	-	-	1
BN1801ZD	-	-	1	-	1
B1301ZB	-	-	1	-	1
B1901Z0	-	-	-	1	1
01801ZD	-	-	-	1	1
BM1801ZD	-	-	-	1	1
W100120	-	-	-	1	1
Not detected	-	-	7	5	12
(b) BK1272OY					
BK12720Y	5	9	1	1	16
EK12720Y	3	1	-	8	12
BX12720Y	1	-	-	-	1
EK1272BY	1	-	-	-	1
BL1272BT	-	-	1	-	1
EK1272BW	-	-	1	-	1
E1272BY	-	-	1	-	1
ET12720Y	-	-	1	-	1
B412720Y	-	-	1	-	1
WB1272	-	-	1	-	1
EK12120Y	-	-	1	-	1
6412720Y	-	-	1	-	1
L1272BY	-	-	1	-	1
EK12728Y	-	-	-	1	1

Next, vehicle detections were performed based on the car's color using 132 cars sample that crossed the highway during the experiment. There were various colors such as gray 11 cars, white 41 cars, silver 19 cars, black 53 cars, red 4 cars, brown 1 car, yellow 1 car, blue 1 car, and orange 1 car. Detection result is shown in Table 2 and the time detection graph is shown in Figure 9.

In this study, the YOLOv5 is implemented for object detection as well. Samples were compiled in Figure 10 and Figure 11. The recapitulation of measurements for car detection is presented in Table 3.

Table 2. Detection results in vehicle plate number based on car color

No	Cars color	Cars amount	Plate number detected	Plate number not detected	Percentage
1	Gray	11	7	4	63.64 %
2	White	41	37	4	90.24 %
3	Silver	19	16	3	84.21 %
4	Black	53	48	5	90.57 %
5	Red	4	4	0	100.00 %
6	Brown	1	1	0	100.00 %
7	Yellow	1	1	0	100.00 %
8	Blue	1	1	0	100.00 %
9	Orange	1	1	0	100.00 %

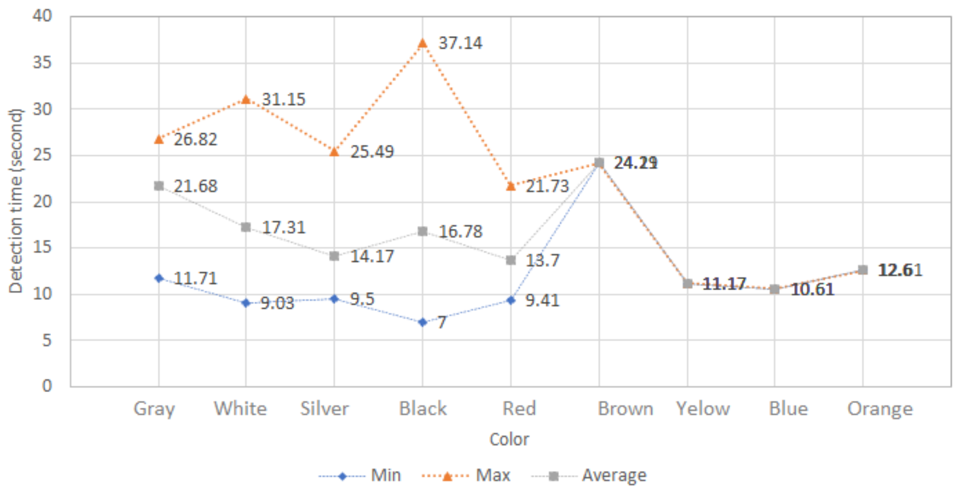


Figure 9. Detection time chart based on cars color

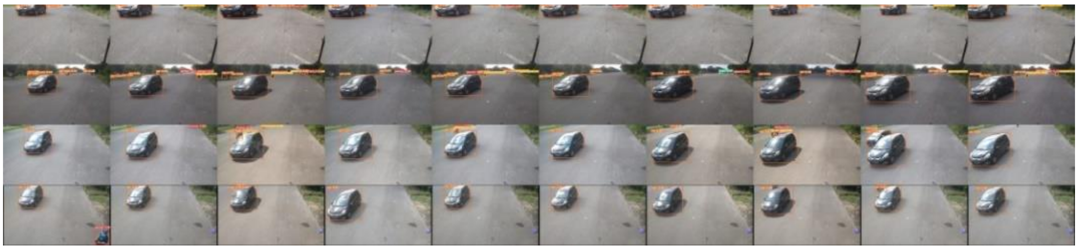


Figure 10. Compilation results of car detection plate number BK1272OY

Figure 11. Compilation results of car detection plate number BK1801ZD

Table 3. Recapitulation of measurement for car detection

No	Frame	BK 1272 OY				BK 1801 ZD			
		Accuracy				Accuracy			
		1 m	2 m	3 m	4 m	1 m	2 m	3 m	4 m
1	Frame_1	0.92	0.94	0.92	0.93	0.90	0.92	0.88	0.91
2	Frame_2	0.93	0.94	0.89	0.92	0.90	0.92	0.89	0.91
3	Frame_3	0.93	0.93	0.90	0.92	0.90	0.92	0.90	0.91
4	Frame_4	0.93	0.94	0.91	0.92	0.90	0.91	0.87	0.92
5	Frame_5	0.93	0.94	0.88	0.93	0.90	0.91	0.88	0.92
6	Frame_6	0.93	0.94	0.91	0.92	0.91	0.91	0.88	0.93
7	Frame_7	0.93	0.94	0.92	0.90	0.88	0.92	0.89	0.90
8	Frame_8	0.92	0.94	0.91	0.89	0.86	0.92	0.89	0.90
9	Frame_9	0.93	0.94	0.89	0.92	0.89	0.93	0.89	0.91
10	Frame_10	0.93	0.93	0.84	0.90	0.88	0.90	0.88	0.92
	Average	0.93 (93%)	0.94 (94%)	0.90 (90%)	0.92 (92%)	0.89 (89%)	0.92 (92%)	0.89 (89%)	0.91 (91%)

Figure 12 shows image compilations from the experiment. The measurement recapitulations for color detection are depicted in Table 4. Orange exerts the highest accuracy and blue experienced the worst.

Figure 12. YOLOv5 detection result for color detection

Table 4. Measurement recapitulation for color detection

No	Vehicle color	Accuracy
1	Gray	0.90
2	White	0.89
3	SIlver	0.91
4	Black	0.91
5	Red	0.91
6	Brown	0.87
7	Yellow	0.89
8	Blue	0.80
9	Orange	0.94

4.2. Analysis

4.2.1. Convex area

The area of monitoring must be convex as shown in Figure 13. The PDAF technology can be adjusted based on light reading from two different angles until they become one phase. The process is similar to a pair of eyes where right side and left side will look to one side.

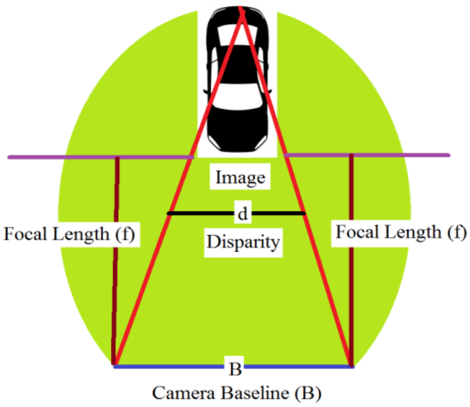


Figure 13. Camera view of PDAF [42]

4.2.2. Field of view

The object must be captured symmetrically into the field of view (FoV) of the camera. In this study, there was the difference in height of the camera, namely heights of 1 m, 2 m, 3 m, and 4 m. The illustration of the FoV form can be seen in Figure 14. Figure 14(a) shows 1 m camera height, Figures 14(b) to 14(d) show the 2, 3 and 4 m height camera where FoV are located higher than the object.

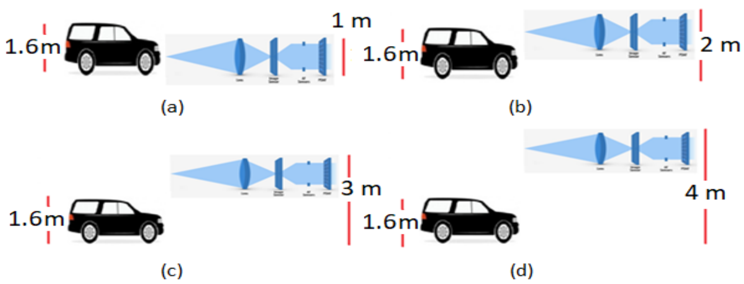


Figure 14. Camera placement at (a) a height of 1 m, (b) a height of 2 m, (c) a height of 3 m, and (d) a height of 4m

Table 4 shows that highest accuracy at a height of 2 m. Detection results for plate no. BK1272OY is 0.94 (94%). Detection results for plate no. BK1801ZD is 0.92 (92%). Based on Figure 14, The height of 2 m can cover the whole car dimension. There is a different disparity value according to object movement toward the camera. Disparity value, (d) can be calculated using (1) [43].

$$d = \frac{s\ x\ resolution}{f\ x\ B} \quad (1)$$

Disparity shows to the difference of object location seen by the left and right eyes. For the camera, disparity, (d) is calculated based on focal length, (f) which is 3.7 mm, camera baseline, which is 28.3 mm and resolution which is 13 MP. The calculation result by using (1) is shown in Table 5.

Furthermore, the depth of image value (Z) can be calculated by using (2) [43]. The f value reflects focal point length, B is based line and d is disparity. The calculation result by using (2) is shown in Table 6.

$$Z = \frac{B\ x\ f}{d} \quad (2)$$

Table 5. Disparity towards distance

No	Distance (m)	Disparity
1	5	0.62×10^6
2	10	1.24×10^6
3	15	$1,86\times10^6$
4	20	2.48×10^6
5	25	3.10×10^6
6	30	3.72×10^6
7	35	4.34×10^6
8	40	4.96×10^6
9	45	5.58×10^6
10	50	6.20×10^6

Table 6. Disparity toward depth image

	Disparity	Depth image
1	0.62×10^6	168.88×10^9
2	1.24×10^6	84.44×10^9
3	$1,86\times10^6$	56.29×10^9
4	2.48×10^6	42.22×10^9
5	3.10×10^6	33.77×10^9
6	3.72×10^6	28.14×10^9
7	4.34×10^6	24.12×10^9
8	4.96×10^6	21.11×10^9
9	5.58×10^6	18.76×10^9
10	6.20×10^6	16.88×10^9

4.2.3. Color analysis

Object detection needs certain features as reference. Color is one of the feature as a reference. There are several factors such as noise in an image, natural articulation of objects, blocked by another object, complicated object shape, and drastic changes in lighting. Table 3 shows the various types of car colors that were sampled in this study with methods that clarify the image, namely:

– RGB filters (red, green, blue)

RGB Filters are the basic filter of all images captured by the camera. The RGB filter will separate the red, green, and blue components and their combinations so that only the object image appears and the background image is filtered. This filter is used for images with red, yellow, blue, and white car samples.

– CMY filters (cyan, magenta, yellow)

CMY Filter is a derivative filter of the basic RGB filter. The CMY filter will separate the components of the cyan, magenta, and yellow, and their combinations so that only the object image appears and the background image is filtered. This filter is used for images with black, brown, and orange car samples.

– Colorizations

Colorizations help black, gray, and white images to become colorful [44], [45]. This is done so that the image becomes better and clearer. These colorizations are used for images that have a gray car sample. Table 3 shows the gray car sample has an accuracy percentage of 63.64% and this value is the lowest compared to the others. This is because each colorization's performance is calculated based on the peak signal-to-noise ratio (PSNR) value. The PSNR formula is:

$$PSNR = 10\, log_{10} \left(\frac{255^2}{MSE}\right) \quad (3)$$

Mean square error (MSE) is the similarity parameter between the colored image and the original image. The more similar the MSE value is closer to 0. In the object tracking process, the MSE value is affected by the path error with (4).

$$MSE = \sqrt{\frac{\sum(x_{mean})^2 + (y_{mean})^2}{n}} \quad (4)$$

Description:
x_{mean} = object average detection time response x
y_{mean} = object average detection time response y
n = number of trials (10 times)
Based on the graph of the response time of detection in Figure 10, the car is gray (x) with x_{mean} = 21.68 sec and the car is black (y) with y_{mean} = 16.78 sec. By using (4), the path error value for the gray car (x) = 4.84 and the path error value for the black car (y) = 3.75.

5. CONCLUSION

Vehicle plate detection by MWSN sensor has been performed by using license number and colors. Object detection by using YOLOv5 produces the best accuray at sensor height of 2 m. Object detection is influenced by the position of the sensor and the type of the technology which is represented by the disparity values. These values were influenced by the distance of the vehicle to the camera. Based on the calculation results, the farther the vehicle is from the camera, the higher the disparity value and the lower the depth image value. The disparity value of the maximum distance of 50 m is 6.20×10^6. At this condition, the depth image is 16.88×10^9. The detection accuracy of both vehicles is not affected by the detection process time. Fast and long detection time yield similar accuracy. However, image transmission within network influences detection process. The larger the image size, the longer the transmission time. Meanwhile, object color influences the detection, recognition, and tracking performance. Orange has the best accuracy, but the gray has the largest path error value.

ACKNOWLEDGEMENTS

This work was supported by the Doctoral Research Schema of Politeknik Negeri Medan, 2020.

REFERENCES

[1] A. Rajeevan, N. K. Payagala, and S. Lanka, "Vehicle monitoring controlling and tracking system by using android application," *International Journal of Technical Research and Applications*, vol. 4, no. 1, pp. 114–119, 2017, doi: 10.13140/RG.2.2.10243.40480.
[2] C. Aydin, C. Tarhan, and V. Tecim, "IT based vehicle tracking system for effective management in public organizations," *Procedia Economics and Finance*, vol. 33, pp. 506–517, 2015, doi: 10.1016/s2212-5671(15)01733-5.
[3] K. Marios, C. Konstantinos, N. Sotiris, and R. José, "Passive target tracking: application with mobile devices using an indoors WSN future internet testbed," in *2011 International Conference on Distributed Computing in Sensor Systems and Workshops, DCOSS'11*, 2011, pp. 1–6, doi: 10.1109/DCOSS.2011.5982182.
[4] Y. X. Li, L. Bin Lu, and D. Y. Liu, "Research on battlefield target tracking in wireless sensor networks," in *2010 2nd International Workshop on Database Technology and Applications, DBTA2010 - Proceedings*, 2010, pp. 1–4, doi: 10.1109/DBTA.2010.5658970.
[5] A. Greenblatt, K. Panetta, and S. Agaian, "Border crossing detection and tracking through localized image processing," in *2008 IEEE International Conference on Technologies for Homeland Security, HST'08*, 2008, pp. 333–338, doi: 10.1109/THS.2008.4534473.
[6] P. Spachos, L. Song, and D. Hatzinakos, "Gas leak detection and localization system through wireless sensor networks," in *2014 IEEE 11th Consumer Communications and Networking Conference, CCNC 2014*, 2014, pp. 1130–1131, doi: 10.1109/ccnc.2014.6940506.
[7] Z. Wei, X. Wang, W. An, and J. Che, "Target-tracking based early fire smoke detection in video," in *Proceedings of the 5th International Conference on Image and Graphics, ICIG 2009*, 2009, pp. 172–176, doi: 10.1109/ICIG.2009.173.
[8] F. Lalooses, H. Susanto, and C. H. Chang, "Approach for tracking wildlife using wireless sensor," in *Proceedings of the 7th International Conference on New Technolo- gies of Distributed Systems*, 2007, pp. 1–7.
[9] I. Slimani, A. Zaarane, W. Al Okaishi, I. Atouf, and A. Hamdoun, "An automated license plate detection and recognition system based on wavelet decomposition and CNN," *Array*, vol. 8, p. 100040, 2020, doi: 10.1016/j.array.2020.100040.
[10] S. Samanta and S. Roy, "Future vehicle tracking system based on wmsn," *International journal of engineering sciences & research technology*, vol. 5, no. 7, pp. 1282–1290, 2016.
[11] I. W. Notonogoro, Jondri, and A. Arifianto, "Indonesian license plate recognition using convolutional neural network," in *2018 6th International Conference on Information and Communication Technology, ICoICT 2018*, 2018, pp. 366–369,

doi: 10.1109/ICoICT.2018.8528761.
[12] I. Shafi *et al.*, "License plate identification and recognition in a non-standard environment using neural pattern matching," *Complex and Intelligent Systems*, vol. 8, no. 5, pp. 3627–3639, 2022, doi: 10.1007/s40747-021-00419-5.
[13] M. Darji, J. Dave, N. Asif, C. Godawat, V. Chudasama, and K. Upla, "Licence plate identification and recognition for non-helmeted motorcyclists using light-weight convolution neural network," in *2020 International Conference for Emerging Technology, INCET 2020*, 2020, pp. 1–6, doi: 10.1109/INCET49848.2020.9154075.
[14] I. Kim, M. M. Khan, T. W. Awan, and Y. Soh, "Multi-target tracking using color information," *International Journal of Computer and Communication Engineering*, vol. 3, no. 1, pp. 11–15, 2014, doi: 10.7763/ijcce.2014.v3.283.
[15] K. She, G. Bebis, H. Gu, and R. Miller, "Vehicle tracking using on-line fusion of color and shape features," in *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 2004, pp. 731–736, doi: 10.1109/itsc.2004.1398993.
[16] G. S. V. S. Siva Ram, K. R. Ramakrishnan, P. K. Atrey, V. K. Singh, and M. S. Kankanhalli, "A design methodology for selection and placement of sensors in multimedia surveillance systems," in *Proceedings of the ACM International Multimedia Conference and Exhibition*, 2006, pp. 121–130, doi: 10.1145/1178782.1178801.
[17] P. Śliwiński and P. Wachel, "A simple model for on-sensor phase-detection autofocusing algorithm," *Journal of Computer and Communications*, vol. 01, no. 06, pp. 11–17, 2013, doi: 10.4236/jcc.2013.16003.
[18] Y. Jang, H. Kim, K. Kim, S. Kim, S. Lee, and J. Yim, "A new PDAF correction method of CMOS image sensor with nonacell and super PD to improve image quality in binning mode," *IS and T International Symposium on Electronic Imaging Science and Technology*, vol. 2021, no. 9, p. 220, 2021, doi: 10.2352/ISSN.2470-1173.2021.9.IQSP-220.
[19] C. C. Chan and H. H. Chen, "Improving the reliability of phase detection autofocus," *Electronic Imaging*, vol. 2018, no. 5, p. 241, 2018, doi: 10.2352/ISSN.2470-1173.2018.05.PMII-241.
[20] D. Mengu, Y. Luo, Y. Rivenson, X. Lin, M. Veli, and A. Ozcan, "Response to comment on 'All-optical machine learning using diffractive deep neural networks,'" *Science*, vol. 361, no. 6406, pp. 1004–1008, Oct. 2018.
[21] D. A. Bashar, "Survey on evolving deep learning neural network architectures," *Journal of Artificial Intelligence and Capsule Networks*, vol. 2019, no. 2, pp. 73–82, 2019, doi: 10.36548/jaicn.2019.2.003.
[22] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.
[23] P. Ongsulee, "Artificial intelligence, machine learning and deep learning," in *2017 15th International Conference on ICT and Knowledge Engineering (ICT&KE)*, 2017, pp. 1–6.
[24] Z. Wu, T. Yao, Y. Fu, and Y.-G. Jiang, "Deep learning for video classification and captioning," *Frontiers of Multimedia Research*, pp. 3–29, 2017, doi: 10.1145/3122865.3122867.
[25] M. M. Ghazi and H. K. Ekenel, "A comprehensive analysis of deep learning based representation for face recognition," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pp. 102–109, doi: 10.1109/CVPRW.2016.20.
[26] Z. Yin and J. Zhang, "Cross-session classification of mental workload levels using EEG and an adaptive deep learning model," *Biomedical Signal Processing and Control*, vol. 33, pp. 30–47, 2017, doi: 10.1016/j.bspc.2016.11.013.
[27] D. C. Cireşan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Deep neural networks segment neuronal membranes in electron microscopy images," *Advances in Neural Information Processing Systems*, vol. 4, pp. 2843–2851, 2012.
[28] M. R. Karim, M. Sewak, and P. Pujari, "Practical convolutional neural networks : Implement advanced deep learning models using Python," *Packt Publishing Ltd*, vol. 1°, p. 211, 2018.
[29] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016, vol. 2016-Decem, pp. 779–788, doi: 10.1109/CVPR.2016.91.
[30] M. Jogin, Mohana, M. S. Madhulika, G. D. Divya, R. K. Meghana, and S. Apoorva, "Feature extraction using convolution neural networks (CNN) and deep learning," in *2018 3rd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, RTEICT 2018 - Proceedings*, 2018, pp. 2319–2323, doi: 10.1109/RTEICT42901.2018.9012507.
[31] R. Ahila Priyadharshini, S. Arivazhagan, M. Arun, and A. Mirnalini, "Maize leaf disease classification using deep convolutional neural networks," *Neural Computing and Applications*, vol. 31, no. 12, pp. 8887–8895, Dec. 2019, doi: 10.1007/s00521-019-04228-3.
[32] S. Indolia, A. K. Goswami, S. P. Mishra, and P. Asopa, "Conceptual understanding of convolutional neural network- a deep learning approach," *Procedia Computer Science*, vol. 132, pp. 679–688, 2018, doi: 10.1016/j.procs.2018.05.069.
[33] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *Proceedings of 2017 International Conference on Engineering and Technology, ICET 2017*, 2018, vol. 2018-Janua, pp. 1–6, doi: 10.1109/ICEngTechnol.2017.8308186.
[34] C. J. Ortiz-Echeverri, S. Salazar-Colores, J. Rodríguez-Reséndiz, and R. A. Gómez-Loenzo, "A new approach for motor imagery classification based on sorted blind source separation, continuous wavelet transform, and convolutional neural network," *Sensors (Switzerland)*, vol. 19, no. 20, p. 4541, 2019, doi: 10.3390/s19204541.
[35] H. Aliady and D. T. Utari, "GPU based image classification using convolutional neural network chicken dishes classification," *International Journal of Advances in Soft Computing and its Applications*, vol. 10, no. 2, pp. 1–13, 2018.
[36] M. Sun, Z. Song, X. Jiang, J. Pan, and Y. Pang, "Learning pooling for convolutional neural network," *Neurocomputing*, vol. 224, pp. 96–104, 2017, doi: 10.1016/j.neucom.2016.10.049.
[37] H. Gholamalinezhad and H. Khosravi, "Pooling methods in deep neural networks, a review," *arXiv preprint arXiv:2009.07485*, p. 16, 2020, doi: 10.48550/arXiv.2009.07485.
[38] D. Thuan, "Evolution of yolo algorithm and yolov5: The state-of-the-art object detection algorithm," Oulu University of Applied Sciences, 2021.
[39] D. L. Maloney and W. F., "R&D and development," *The World Bank*, p. 38, 2003.
[40] R. R. Anggraeni and D., "Decision support system selection used car using the fuzzy logic method of tahani model," in *International Conference on Social, Sciences and Information Technology*, 2020, vol. 1, no. 1, pp. 235–242, doi: 10.33330/icossit.v1i1.729.
[41] S. Sutono, S. L. B. R. Ginting, M. F. Wicaksono, and K. R. Tembo, "Multi sensors application for automatic portal access in residential complexes," in *IOP Conference Series: Materials Science and Engineering*, 2019, vol. 662, no. 5, doi: 10.1088/1757-899X/662/5/052006.
[42] C. J. Ho and H. H. Chen, "On the distinction between phase images and two-view light field for PDAF of mobile imaging," *IS and T International Symposium on Electronic Imaging Science and Technology*, vol. 2020, no. 6, 2020, doi: 10.2352/ISSN.2470-1173.2020.14.COIMG-390.
[43] P. J. Aswin, J. S. Chandana, S. Reghunath, and M. Menon, "Stereo-vision based system for object detection and recognition," in *Proceedings of the International Conference on Trends in Electronics and Informatics, ICOEI 2019*, 2019, vol. 2019-April, pp. 1284–1288, doi: 10.1109/icoei.2019.8862588.
[44] N. K. EL Abbadi and E. Saleem, "Automatic gray images colorization based on lab color space," *Indonesian Journal of Electrical*

Engineering and Computer Science IJEECS, vol. 18, no. 3, pp. 1501–1509, 2020, doi: 10.11591/ijeecs.v18.i3.pp1501-1509.

[45] I. Fawwaz, M. Zarlis, Suherman, and R. F. Rahmat, “The edge detection enhancement on satellite image using bilateral filter,” in *IOP Conference Series: Materials Science and Engineering*, 2018, vol. 308, no. 1, doi: 10.1088/1757-899X/308/1/012052.

BIOGRAPHIES OF AUTHORS

Afritha Amelia obtained her master’s degree in Multimedia Telecommunication from Institut Teknologi Sepuluh November Surabaya in 2009. Energyly, she is a Ph.D student in Computer Science Universitas Sumatera Utara, Indonesia. Her research is mainly focused on IoT application, wireless telecommunication dan artificial intelligence. She can be contacted at email: amelia.afritha@gmail.com.

Muhammad Zarlis Doctor (Ph.D) in Computer Science from University Sains Malaysia (USM), Malaysia in 2002. Masters in Computer Science from Universitas Indonesia (UI) in 1990, and Sandwich Program with University of Maryland (UoM), USA. He is a Professor at Binus University and his focus researches are intelligent system and big data, computational method and modeling, computational physics (simulation and modelling), algorithm and programming, numerical method and optimization. He can be contacted at email: m.zarlis@usu.ac.id.

Suherman obtained his Ph.D in Electrical and Electronics Engineering from De Montfort University, Leicester, UK, in 2013, Master degree in Networked and Distributed Systems (Computing) from RMIT, Australia in 2009 and Bachelor degree in Electrical Engineering, USU Indonesia in 2000. His research interest includes communication network and protocol, applied communication, and distributed system. Currently, he leads the Electrical Engineering Department and in his spare time, he is studying data science in financial through MSc program in Financial Engineering. He can be contacted at email: suherman@usu.ac.id.

Syahril Efendi Doctor (Ph.D) in Computer Science from Universitas Sumatera Utara, Indonesia in 2013. His focus researches are computational mathematics, neural network, computational method and modeling, numerical method and optimization. He can be contacted at email: syahril1@usu.ac.id.