Object Distance Measurement System Using Monocular Camera on Vehicle

Fussy Mentari Dirgantara
School of Electrical Engineering and Informatics
Institut Teknologi Bandung
Bandung, Indonesia
fussymd@students.itb.ac.id

Arief Syaichu Rohman
School of Electrical Engineering and Informatics
Institut Teknologi Bandung
Bandung, Indonesia
arief@lskk.ee.itb.ac.id

Lenni Yulianti
School of Electrical Engineering and Informatics
Institut Teknologi Bandung
Bandung, Indonesia
lenni@lskk.ee.itb.ac.id

***Abstract*— To support autonomous vehicles that are currently often studied by various parties, the authors propose to make a system of predicting the distance of objects using monocular cameras on vehicles. Distance prediction uses four methods and the input parameter was obtained from images processed with MobileNets SSD. Calculations using linear regression are the simplest calculations among the four methods but have an error of 1% with a standard deviation of 1.65 meters. While using the first method, the average error value is 9% with a standard deviation of 0.43 meters. By using the second calculation, the average error resulted in 6% with a standard deviation of 0.35 meters. The experimental method had an average error of 1% with a standard deviation of 0.26 meters, so the experimental method was used.**

Keywords—distance prediction, monocular camera, MobileNets SSD, vehicle

I. INTRODUCTION

The development of robotics which is increasing at this time supports certain machines that already have the ability to infer the conditions of their surroundings to continue to develop. The ability to detect and track objects is an important condition of a robot in order to know the surrounding conditions. The development of image-based sensors for the improvement and progress of robotics has been applied to remote surveillance and environmental reading. The main task of image-based sensors on a vehicle is to detect objects and collect visual information between the camera and the object [1].

Applicative areas that implement algorithms to detect objects include robots and unmanned vehicles. These two studies use information obtained from visions mounted on robots. In this case, unmanned vehicles make use of object detection from stored videos and in real time. The challenge for unmanned vehicles in the next generation is to reduce collisions and improve the safety of vehicles and objects that surround them using lane information [2]. Further research shows that object detection can help the sense of sight of the driver and the system in the unmanned vehicle. Areas of research that are actively developing warning systems include suppliers, industry and education.

The tools used to detect objects can vary, including using a LIDAR, ultrasonic sensor, stereo camera, or monocular camera. However, in some studies the tool that is often used is a monocular camera. According to study by Chen, et al. in 2011, processing systems in monocular cameras use low power, in addition to the low implementation costs [3]. This has become one of the advantages of monocular cameras because it can be mass produced by a company. In previous studies according to Guaman, et al. in 2019, research on automatic object detection has become very important in recent years, where the development of systems for detecting objects is the main step to help drivers who need to calculate distances between objects so as to warn the driver to slow down his vehicle and avoid accidents in the form of collisions [4].

Distance is calculated between the camera and object detected by MobileNets SSD. MobileNets is popular because of its sleek and simple CNN architecture for computer vision practice [5]. An action camera will be mounted on the unmanned electric vehicle in order to get a wide view afore, camera capture frames and process the image to draws bounding box circumscribing for every particular object. An object that had been narrowed down on is sent to the processor. Steps that must be done to know the distance between platform to object is using a camera to detect and track the location of the nearest object in forepart. Next step is assessing the distance between the host vehicle with the objects in front [6].

Based on the above information, this research will implement an object detection system that utilizes a monocular camera and processed with MobileNets SSD. Lightweight pre-trained data from diverse Caffe model gathers data of distance through the platform processor for the autonomous vehicle decision-making process. In this paper, specific inputs is the location of the bounding box of detected object.

II. LITERATURE STUDY

A. MobileNets SSD

MobileNets SSD (Single Shot MultiBox Detector) is a main contemporary approach in object detection by reason due to its accuracy. MobileNets is one of the convolutional neural network (CNN) architectures which can be used to address the need for computing excessive resources. Therefore, SSD is faster than Fast Region-based Convolutional Neural Network based on study by Howard, et al in 2017. In addition, SSD has a lightweight architectural

load, namely networks that use 3x3 integral convolutions so that they can produce up to 9 times less separation compared to other standard convolution [7].

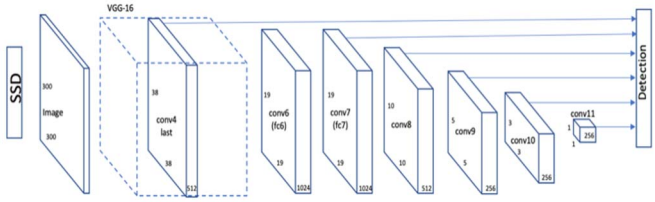

Fig. 1. Model architecture of Single Shot MultiBox Detection (reproduced from reference [5]).

Figure 1 shows that the SSD architecture model originates from a D_F x D_F input image then extracts valuable form features. For every input channel, the depth-wise convolution adjusts a single filter in MobileNets [7]. Depth-wise Separable Convolution is a composite of depth-wise convolution and spatial convolution.

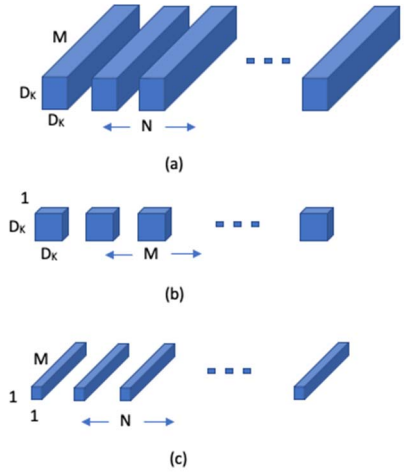

Fig. 2. Disparity of Standard Convolutional Filters with MobileNets (reproduced from reference [7]).

In Figure 2, M denotes the number of the input channel, D_K is Kernel size and N is the number of the output channel. Based on the research picture described above, (a) represent Standard Convolution Filters, (b) indicate Depth-wise Convolutional Filters and (c) portray 1x1 Convolutional Filters called Pointwise Convolutional, both (b) and (c) in the context of Depth-wise Separable Convolution. This method has a streamlined function that will be applied to the object detection system. The swiftness of the computational process is one of its huge advantages, since the system will be plugged on the real-time database. Steering for the autonomous vehicle would run smoothly to avoid object [8].

B. Object Detection

The type of information obtained from detection object via camera can vary, depending on the functionality and goals that are to be achieved. In addition to knowing whether a captured image presents a defined object if it is then it will indicate its position on the image. Object detection is very dependent on classification, but what is also important in determining objects is the selection of the right features. This system has the main goal of choosing the most characteristic-specific features of the object sought, it can be said, very detailed review, so that classifier can produce an accurate response.

Operations of the method that are required for detecting object are:

- Gain sampling points characteristic in the captured frame.
- Implement clustering of the sampling point. To achieve high performance, it needs to iterate this process several times.
- Shape a group of classifiers that consist of trained sampling point of dissimilar data section alongside its membership weights.
- The object is detected and will continually move to a sequence of another frame [9].

Feature extraction is essential for object detection, functioning as an extractor of data. It has been captured by a digital vision sensor to data which can be understood by the computer. The figure below shows the difference between two object detection method results.

Fig. 3. Object Detection using Viola-Jones.

In Figure 3, reflection is seen to have a larger shape than real objects and the Viola-Jones method has a longer data processing time, which is not really suitable for real time systems. This has a number of differences in computational time with other object detection methods.

Fig. 4. Object Detection using MobileNets SSD method.

In Figure 4, objects detected by MobileNets SSD have a more precise form. For indexing performance, there are identification classifications such as MobileNets SSD. In

processing image data when being compared to Viola-Jones, MobileNets SSD resulted in a faster process. The computing time required to perform Viola-Jones was slower at 0.163 seconds per frame, while SSD MobileNets resulted in 0.084 seconds per frame.

C. *Frame Based Distance Measurement*

Calculating the distance between the host platform and the object in front was chosen for this method. Horizontal and vertical distances will later form a distance score. The first method is the method we adapted from Jamzad et al. This method calculates the distance from the position of objects in the image matrix. This method does not depend on the size of the detected object and therefore has fewer errors [11].

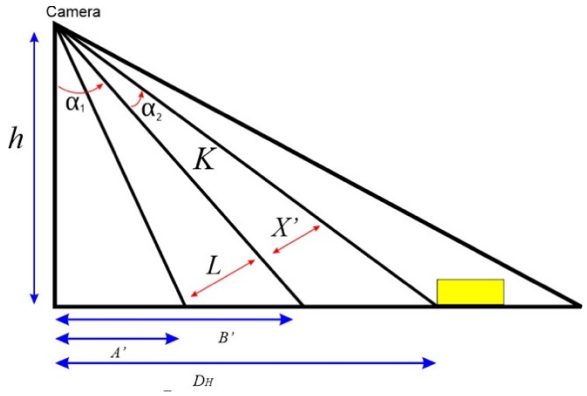

Fig. 5. Object distance calculation method (reproduced from reference [11]).

In Figure 5, A ', B', and h are constant variables that can be calculated offline. A 'is the distance from the actual position of the camera to the bottom of the frame that was taken. B 'represents the distance from the actual position of the camera to the centerline of the frame taken. And h is the real height of the camera lens. Other variables can be calculated using the equation as below [11]:

$$\alpha_1 = tan^{-1}\left(\frac{B'}{h}\right) \quad (1)$$

$$L = (B' - A')\, sin\left(\frac{\pi}{2} - \alpha_1\right) \quad (2)$$

$$K = \sqrt{h^2 + A'^2 - L^2} \quad (3)$$

$$X' = L\left(1 - \frac{2X_0}{I_p}\right) \quad (4)$$

$$\alpha_2 = tan^{-1}\left(\frac{X'}{K}\right) \quad (5)$$

$$D_H = h\, tan(\alpha_1 + \alpha_2) \quad (6)$$

Where α_1 stands for angle of camera to the distance of *B'* which as calculated on Equation (1). *L* state for diagonal distance between *A'* and *B'*. *K* is the hypotenuse value between *h* and *B'*. D_H is the adjacent value from camera to the object. X_0 represent value of pixels got from image bottom position to the lowest point of y in detected object using MobileNets, and I_p is the number of pixels obtained from image vertical height in captured figure which is 720 px.

In Figure 6, the *h* value is obtained from the height of the camera on the actual platform. Whereas *k* is calculated from the apparent distance between the camera to the lowest part of the image captured by the sensor.

Second method used for distance measurement is determined by combining the ratio between moving object pixels and the angle ratio, with the following equation [8]:

$$\frac{\gamma_r}{2\alpha} = \frac{X_0}{Np_{max}} \quad (8)$$

$$D_h = h\, tan(\theta + \gamma_r) \quad (9)$$

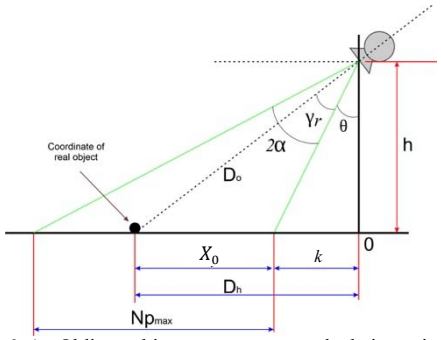

Fig. 6. An Oblique object measurement calculation using second method (reproduced from reference [8]).

$$D_o = \sqrt{{D_h}^2 + h^2} \quad (10)$$

From Figure 6, we can see that γ_r is the angle of the object calculated vertical FOV (field of view). 2α is the vertical angle of the frame known from the camera. X_0 shows the total number of pixels from the bottom frame catch to the detected object. N_{pmax} represents the number of pixels calculated in the Y-direction camera FOV. D_h is the distance calculated from the camera to the object and D_o is the diagonal distance of the camera to the detected object

III. Experimental Setup

This section discusses the whole system design approach in this paper. As shown below in Figure 7, the action camera on the vehicle dashboard is connected using a USB cable to a processor (laptop):

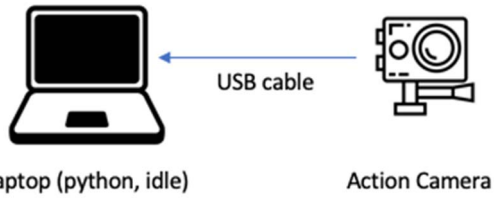

Fig. 7. Main object detection system to measure distance on automated vehicle prototype.

In this study we used the SJ Cam X1000 action camera (Figure 8) to capture moments as images or videos, which can later be stored as offline data or processed in real time.

Fig. 8. SJCAM X1000 connected to the processor.

Image processing runs in a portable computer (laptop) powered by Intel Core i5 1,4 GHz with 4GB RAM. Table I is an action camera specification seen from the vertical angle of view, image size and frame rate of the camera.

TABLE I. CAMERA SPECIFICATION

Vertical Angle of View	74.589°
Image Size	1080 x 720
Frame Rate	30 fps

Information from the visual sensor is sent to IDLE version 3.6.6. Because the captured image has a 1080x720 px frame, then we changed the IDLE frame size to 300x300 beforehand so MobileNets can process it lightly. MobileNets SSD itself is used to check whether objects are detected or not until the next literacy. When the object has been identified, a bounding box will appear around the object. This will also send the number of pixels from the bottom frame to the object box with the lowest limit and the number of pixels received from the height of the image.

Fig. 9. Camera mounted in the vehicle.

Figure 9 shows how the camera is mounted in vehicle at an altitude of 1.41 meter. An action camera is mounted in the middle to get a wider field of view. Also, experiments are carried out during the day and objects are placed right in front of the trajectory that is passed by the vehicle.

In the initial stages the camera will capture images and send data to the processor, it detects whether the object is detected or not. If the object is not detected, the program will continue to look for other objects in the next frame. When an object is detected, the computer will reach X_0 data as an input parameter for distance calculation. Input parameters will be processed and we will get distance estimation. After the estimated distance is reached, the computer will read the image sensor to detect the object and calculate the distance as in Figure 10.

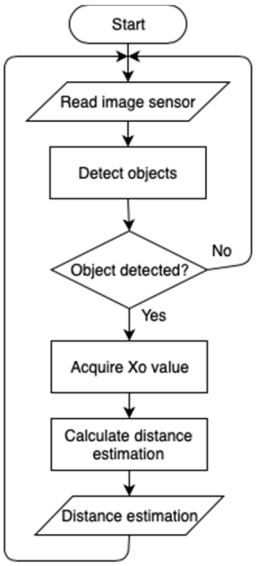

Fig. 10. Flow Chart of the distance measurement method.

IV. EXPERIMENT RESULT

In this section, we will explain the results of certain tests to detect objects and calculate distances. Initial tests describe the procedure of object detection systems for obtaining frames and drawing bounding boxes. The bounding box is formed by the detection of objects from image profiles in the sequences of acquired digital images. The distances are measured in spaces between 1 m between 5 m and 20 m. The size of the image taken by OpenCV is 1080 x 720 pixels and then converted to 300x300 so that the image can be used lightly by MobileNets SSD.

In this study, after the image from the action camera is captured, the picture will be processed in order to produce X_0. The distance value of *A'* is 2.841 meter and *B'* is 29 meter, data got from manual calculation in the initial experiment. Data given from comparing D_H from the first method with input parameter (X_0) in formula (4), as shown in Figure 11, the blue-colored line represents value from the first method, and the orange-colored line represents real distance value:

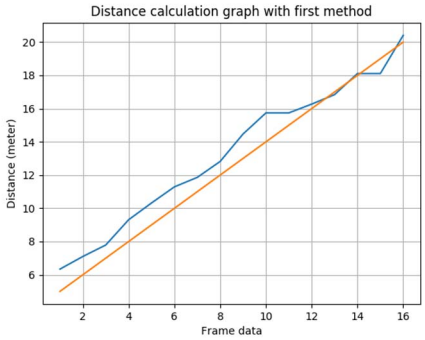

Fig. 11. Graph of prediction system with first method.

Other data needs to be applied for distance calculation in other formulas needed for X_0 can be given. X_0 is the total amount of pixels from lowest frame capture to the detected object. Graph describes for X_0 towards real distance from formula (8) as shown in Figure 12 has a different value from Figure 11. The resulting figures are different but have a similar graphical shape because the input parameter values are the same. :

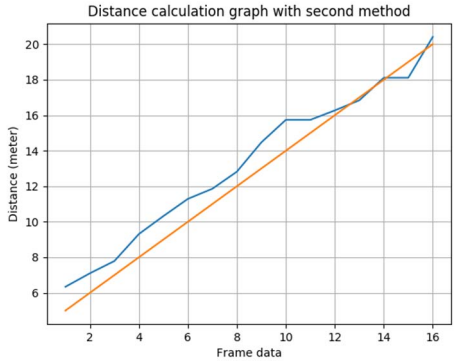

Fig. 12. Distance prediction with second method.

By using a curve fitting tool in Matlab as an experimental research, we looked for the relationship between the value of X_0 and the real distance value at the time of measurement. The value of X_0 is entered into a curve fitting tool as data X and the real distance value as data Y. The general model used is Gaussian, with the number of terms is 2. Then we got a new equation to be included in the distance value so that the graph is obtained in Figure 13 as follows:

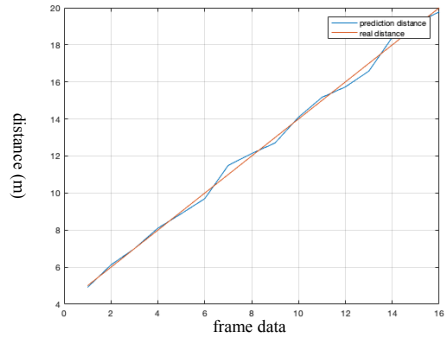

Fig. 13. Distance prediction with experimental computation.

We can have the estimated distance value from using input parameter X_0. At this stage, we will estimate the detected value that can be determined using linear regression. The linear regression equation model is written in the following formula:

$$Y = a + bX$$

Where Y is the dependent variable from 5m to 20m, X represents the independent variable obtained from the input parameters, a is a constant, and b is the regression coefficient.

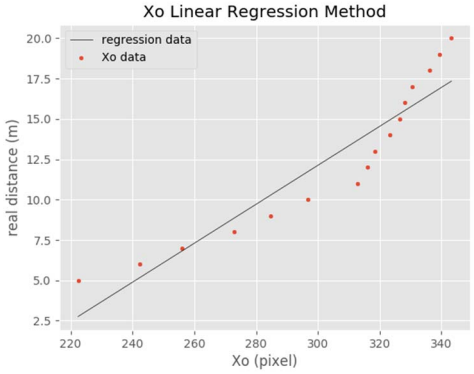

Fig. 14. Distance prediction with linear regression.

In Figure 14, it can be seen that the distribution values of the data obtained are located around a straight line in the range of 5m to 20m. From the data taken, it is known that each calculation has different results.

Table II below is the standard error of the four methods. It can be seen that the smallest standard deviation and average error is experimental data. It means that it can be concluded that experimental data is the best data.

TABLE II. AVERAGE ERROR AND STANDARD DEVIATION

Method	Average Error	Standard Deviation
First method	9%	0.43 meter
Second method	6%	0.35 meter
Experimental	1%	0.26 meter
Linear regression	1%	1.65 meter

V. CONCLUSION

We demonstrated the implementation of computer vision adaptation using the Viola-Jones and MobileNets SSD for object detection. The first approach using Viola-Jones produced slow calculations for object detection method, it is not acceptable to send direct data in real-time. Other experiments with MobileNets SSD resulted in smoother interpretation and faster numbering, assisting with better display sensors for automatic vehicles. Distance measurements using the first method and the second method produced accurate distances in several ranges and inaccurate calculations over several lengths. While using experimental method through a curve fitting tool, distance calculation had better overall prediction. Linear regression had the lowest calculation than other methods, but had bigger average error among them all. The measurement results showed that the accuracy value is sufficient due to some reading errors from the object detection method. If the object detection was more precise, it could produce a better validity value than the distance measurement. To improve the detection results, we must pay attention to lighting factors, image resolution, and processor for computing. Because these three factors affect the object detection results.

REFERENCES

[1] E. Maggio and A. Cavallaro, Video Tracking: Theory and Practice, West Sussex: John Wiley & Sons, 2011.

[2] S.-Y. Kim, J.-K. Kang and S.-Y. Oh, "An Intelligent and Integrated Driver Assistance System for Increased Safety and Convenience Based on All-around Sensing," in *J Intell Robot Syst*, Pohang, 2008.

[3] S.-H. Chen and R.-S. Chen, "Vision-Based Distance Estimation for Multiple Vehicles Using Single Optical Camera Feature," in *Second International Conference on Innovations in Bio-inspired Computing and Applications*, Kaohsiung, 2011.

[4] L. R. B. Guamán and J. E. Naranjo, "Object detection in rural roads through Single Shot Multibox Detector Mobilenet network," in *Proceedings of 2019 the 9th International Workshop on Computer Science and Engineering*, Hong Kong, 2019.

[5] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu and A. C. Berg, "SSD: Single Shot MultiBox Detector," in *Springer International*, Chapel Hill, 2016.

[6] C. Jiangwei, J. Lisheng, G. Lie, Libibing and W. Rongben, "Study on Method of Detecting Preceding Vehicle Based on Monocular Camera," in *IEEE Intelligent Vehicles Symposium*, Parma, 2004.

[7] G. A. Howard, M. Zhu, B. Chen and D. Kalenichenko, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," in *Google Inc*, 2017.

[8] Y. M. Chiang, N. Z. Hsu and K. L. Lin, "Driver Assistance System Based on Monocular Vision," in *Lecture Notes in Computer Science*, 2008.

[9] B. Cyganek, Object Detection and Recognition in Digital Images: Theory and Practice, Krakow: John Wiley & Sons, 2013.

[10] P. Viola and M. Jones, "Robust Real-Time Face Detection," in *Proceedings Eighth IEEE International Conference on Computer Vision*, Cambridge, 2001.

[11] M. Jamzad and A. Foroughnassiraei, "Middle Sized Soccer Robots: ARVAND," in *Proceeding of RoboCup-99: Robot Soccer world Cup III*, Iran, 2000.

[12] A. Mukhtar, L. Xia and T. B. Tang, "Vehicle Detection Techniques for Collision Avoidance Systems: A Review," in *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS*, Tronoh, 2015.

[13] Z. Sun, G. Bebis and R. Miller, "Monocular Precrash Vehicle Detection: Features and Classifiers," in *IEEE TRANSACTIONS ON IMAGE PROCESSING*, Reno, 2006.

[14] N. Sasaki, S. Tomaru and S. Nakamura, "Development of Inter-Vehicle Distance Measurement System using Camera-Equipped Portable Device," in *2017 17th International Conference on Control, Automation and Systems*, Jeju, 2017.