Depth estimation for a robotic arm based on monocular images

Fikrul Akbar Alamsyah, Chi-Cheng Cheng
1 Mechanical and Electromechanical Engineering Department, National Sun Yat-Sen University
2 Mechanical Engineering Department, Brawijaya University
MT Haryono 167, Lowokwaru, Malang, Indonesia
Email: fikrul@ub.ac.id

Abstract Regaining depth information of objects from two-dimensional images is one of the fundamental issues and essential thing in the field of machine vision. There is a depth estimation method using non-inclined with forward movements that was already presented with feature extraction of image processing based on one direction only. This study proposes a depth estimation approach based on forward and backward movement with non-inclined and inclined orientations. Binary threshold will be used in this study to replace feature extraction. The study can approximate the most effective depth estimation using forward and backward movement in frame of non-inclined and inclined orientations. The images captured by an un-calibrated ordinary monocular camera mounted at the end effector of a robot arm. In this approach, the first image is captured and then the camera parameters remain unchanged. The second image is acquired after moving a distance d along the optical axis. Then image segmentation and binary threshold are implemented on the two images separately, and the numbers of black pixels in the images are counted. The experiments were carried out by simulations using webot software and python programming code. The results showed that the second position of the camera 0.1 meter from its first position demonstrated the best estimation performance, both for non-inclined and inclined movements, while the 0.01 meter distance resulted in the greatest error.

Keywords: machine vision, depth estimation, monocular images

1. Introduction
In the context of the smart manufacturing paradigm, a robot is expected to be able doing inspection, measurement and fault detection. These are commonly control procedures and crucial in almost every industrial assembly process [1]. In order to achieve that, the robot should be able to regain depth information of objects and this is one of the fundamental issues in machine vision and robotics. Regaining depth information of objects can be obtained from binocular images with careful camera calibration.

Commonly, there are several methods have been offered to achieve depth estimation, for example by extracting the visual features from the images such as shading, texture, known object sizes, color, defocus, linear perspective, etc. [2-5]. At present, the stereo vision system is the most applicable technique to acquire depth information. The principle of this method is to use two cameras that focus on the same area from a different angle of perspectives. The resulting images
are used in this technique to provide information for determining the position in 3D space. Both the camera’s intrinsic parameters and the parameters of the spatial correspondence between two cameras need to be provided, which means that camera calibration is needed.

Today many industrial tasks are accomplished by robot arms, the mechanical design is based on applications, which are designate to. Commonly robots' arm is equipped with cameras sensors to deal with complex jobs and to make sure that the performance is as intended. There are diverse types of cameras that can be applied, from regular RGB cameras, LIDAR scanners, IR cameras, depth cameras, etc. When cameras are on the robot’s arm, machine vision algorithms handle the image processing. The camera sensor helps the robot arm to watch the target and interpret the environment.

Develop a real robot is still high-priced and lengthy while using commercial robots is relatively limited. Therefore, implementation of robot simulation can be a choice for robotic research and education purposes [6]. Simulation approach and evaluation methods in virtual environments can provide an affordable and reliable framework for experimenting with different sensing and acting mechanisms in order to verify the performance functionality of the robot in dynamic scenarios [7]. One of virtual environment softwares is webot, which is a professional mobile robot simulation software platform and proposes a rapid prototyping environment, that lets the user to create 3D virtual worlds with physical properties such as mass, joints, and friction coefficients, etc. [8].

Lixin He et.al, 2017, developed a novel approach based on SIFT (the Scale Invariant Feature Transform) method to measure depth information of objects from two monocular images. This approach is able to approximate the depths of objects in two images, which are captured by an un-calibrated monocular camera. The method is the first image is captured and the camera parameters remain unchanged. The second image is acquired after moving the camera with distance along the optical axis. Next, image segmentation and SIFT feature extraction are applied on the two images individually, and then the objects in the images can be matched. Finally, the object’s depth is solved by the lengths of a set of straight-line segments. To ensure that the most proper pair of straight-line segments are chosen, and reduce computation, convex hull theory and knowledge of triangle similarity are employed. The experimental results show the approach is effective and practical [9].

In this paper, an automatic measurement method is proposed to compute the depth of target objects using monocular camera mounted at end effector of robot arm. Since there is only a monocular camera, no calibration is required. First step, the image 1 is captured, and then the image 2 after robot arm moving backward or forward with a certain distance along the optical axis. The camera’s intrinsic parameters do not change during the process. Second step, we get the objects by segmenting the images 1 and 2. Then the binary threshold is employed in both images 1 and 2. The sizes of the same object in the images 1 and 2 can be obtained by calculating the number of black pixels and then the absolute depth of an object therefore can be computed.

2. Method
The simulation was done using the 2019 webot robot simulator software, which is open-source and free to use. And it also accommodates several programming languages to control robots, such as Matlab, c / c ++, and python. In this research, to control the robot is achieved by the python programming language and OpenCV is applied to process image data. In general, the work steps for this research can be depicted as shown in Figure 1.

![Figure 1. General stages of working steps to accomplish depth estimation.](image-url)
In this study, the basic principle of image formation supported by the similarity of triangles is applied to acquire the value of the depth. The basic principle of imaging can be modeled as

\[ \frac{1}{f} = \frac{1}{u} + \frac{1}{v} \]  

(1)

where \( f \) is the focal length, \( v \) is the distance between the image plane and lens and \( u \) is object distance (depth). The method used in this study to obtain an image can be described in Figure 2. Figure 2a shows the first position of image capture. Figures 2b and 2c are the second image capture positions for forward or backward movement of the camera that mounted in robot arm, while \( d \) is the small movement of the robot arm to get the second image.

![Diagram](image)

**Figure 2.** a) The first image capture scheme, b) Forward movement to get the second image, c) Backward movement to get the second image.

Based on figure 2 and related to the basic principle of imaging and also triangular similarity, the following formula can be obtained:

\[ u = \frac{h_2}{h_1-h_2} d \]  

(2)

The formula uses \( h_1 \) and \( h_2 \) as the length or height of the target object, which will be replaced by the square root of the number inside pixels of the target object image after binary thresholding is carried out.

In this study, the IRB 4600-40 ABB robot arm model is employed and the working area of this model based on data sheet IRB 4600 features and specification [10] is illustrated as shown in Figure 3. This robot model has 6 axes and it is usually mounted on the floor, rack, upside down, and tilted. Main applications include arc welding, assembly, material handling, machine maintenance, material
removal, and cleaning/spraying, etc. The camera is mounted at the end effector of the robotic arm and the camera is directed to the target object. In order to control the position of the robot arm of end effector, inverse kinematics is implemented with the Python programming language to reduce the complexity of the calculation.

**Figure 3.** Working area specifications are, A = 2872 mm, B = 1735 mm, C = 680 mm, D = 1393 mm, E = 2552 mm, F = 2202 mm.

**Figure 4.** Cubed and star shape as target objects with: a) non-inclined, b) inclined orientation

Figure 4 display two different attitudes of the camera relative to the target object, namely non-inclined (Figure 4a) and inclined (Figure 4b). Non-inclined refers to the camera axis towards the target object, while inclined refers to the axis of the target object which forms a 45-degree angle relative to the camera. The camera position is one meter away from the target object and this distance is inside of working area of the robot arm. In simulation experiments, the camera resolution is 1280 x 1024 pixels. To get the depth of the target object relative to the robot arm, the working steps are illustrated as shown in Figure 5. These working steps are applied for each data retrieval.

Step one implements the inverse kinematics control to make sure the camera reaches coordinates that are in line with the target object’s axis. The process in the second step is carried out using the python programming language including depth calculations.
3. Results and discussion
First step of the process after receiving input image is image segmentation, which separates the target object from background. After segmentation, a region of interest (ROI) will be obtained. Region of Interest typically is determined based on pixel intensity values or user-determined areas by masking or drawing.

![Image Processing Steps](image)

**Figure 5.** The working steps for simulation experiments consisting of input, process, and output.

**Table 1.** Relation of depth measurement to the shape of object and movement of camera

| Object | Length (m) | Non-Inclined Forward | Non-Inclined Backward | Inclined Forward | Inclined Backward |
|--------|------------|----------------------|-----------------------|------------------|------------------|
| Cube   | 0.01       | 0.983                | 1.021                 | 1.050            | 0.855            |
|        | 0.04       | 1.010                | 0.911                 | 1.062            | 1.036            |
|        | 0.07       | 1.082                | 0.960                 | 1.043            | 1.023            |
|        | 0.1        | 1.039                | 1.004                 | 1.006            | 1.040            |
| Star   | 0.01       | 0.885                | 1.270                 | 0.939            | 1.346            |
|        | 0.04       | 0.990                | 1.014                 | 0.940            | 1.035            |
|        | 0.07       | 1.010                | 1.024                 | 0.988            | 1.022            |
|        | 0.1        | 0.997                | 1.022                 | 0.992            | 1.050            |

In this study, we use a square of 500 x 500 pixels as image segmentation with the center point in the middle of the target object (Figure 6b). The size of the image segmentation can be changed by the user according to the size of the target object. After the image segmentation, the next step is to change the image to grayscale, where the advantage of grayscale is only one channel with a value of 0-255 where 0 for black and 255 for white (Figure 6c). Next, a binary threshold is applied to convert a grayscale image into an image with only black and white, by using a threshold with a
certain value (Figure 6d). After the target object is successfully classified into either black or white, then a calculation is carried out for the number of pixels inside the target object. Finally the numbers of pixels from the target object for the position of the first robot arm (before moving) and the position of the second robot arm (after moving) are entered in (2) to get the depth (distance) value.

The distance between the camera and the target object is 1 meter. This distance is used as a reference for the position of the first image taking while the second position is based on distance variations as shown in Table 1. Based on Table 1, it clearly indicates that getting the exact depth / distance value of 1 meter is quite difficult; however, our research shows that the method used is able to approach the actual value. The values shown in Table 1 also implies that the movement of the end effector of the robot arm or the camera with a value of 0.01 meter produces unstable depth estimation. A value of 0.01 meter can produce a low accuracy of 1.346 meter or 65.381% accuracy as shown in the target object star with inclined backward movement. In addition, there is also a target object with a non-inclined backward movement with a value of 1,270 meters or 73.002% accuracy. This be due to the insignificant difference in the movement of the camera for the second position and can affect the resulting image and then image processing and pixel counting afterwards. However, a distance of 0.01 can also produce a distance calculation that is quite high in accuracy, namely 0.983 meters or 98.281% accuracy, which is generated from the targeted cube object with non-inclined forward movement. So it can be concluded that the movement of 0.01 causes depth estimation that is not stable enough. On the other hand, the 0.1 meter movement shows quite good results, such as the non-inclined forward with the star target object, which produces a value of 0.997 meters or 99.7% accuracy. While the lowest value of 0.1 meter movement is obtained at the target star object with backward inclined movement, the value is 1.05 meters or 95.003% accuracy. Overall, the statistical distribution of accuracy for each movement can be summarized as depicted in Figure 7.

It can be seen in Figure 7 that the camera movement of 0.01 meter produces a relatively less accurate than 0.1 meter movement and both of them have significant difference. Different things occur in the 0.04 meter and 0.07 meter movements, which produce relatively similar accuracy, 96.05% and 96.79%. The average value of the accuracy of the movement of 0.01 meter for the cube target object is 94.192%, while for the star object target is 80.199%. As shown in figure 7, the error bar visually shows a fairly low standard deviation. Especially for the 0.01 meter movement where the accuracy data is relatively different compared to the other three types of movements. For more details, please refer to Table 2.
Table 2. Depth measurement accuracy with respect to the shape of object and movement of camera

| Object | Movement Length (m) | Average value of accuracy |
|--------|---------------------|---------------------------|
| Cube   | 0.01                | 94.192                    |
|        | 0.04                | 95.069                    |
|        | 0.07                | 95.289                    |
|        | 0.1                 | 97.779                    |
| Star   | 0.01                | 80.199                    |
|        | 0.04                | 97.023                    |
|        | 0.07                | 98.286                    |
|        | 0.1                 | 97.925                    |

Based on data analysis, it appears that the distance of 0.01 meters presents a fairly large error value, both at inclined and non-inclined orientations. This may be caused by small difference between the first and second positions which cannot be honestly displayed under limited image resolution. While 0.1m indicates outstanding error values ranging between 0.2% and 5%.

4. Conclusion
A hybrid method combining the binary threshold and the basic principle of the image formation for calculating depth was developed in this study. Based on our analysis, this method is quite simple and applicable. To be used on a robot arm is also quite efficient because it only needs a camera and does not occupy much space. This can provide a cost-effective advantage compared to depth measurement using other devices, such as stereo vision which requires two cameras. The working step of this approach is not too complicated and it can be said that it is quite efficient because it only involves a few processes. The level of accuracy of this method appears to be able to reach above 90%, so this approach has potential to be extended to industrial applications.
6. References

[1] Frustaci F 2020 An embedded machine vision system for an in-line quality check of assembly process. *International Conference on Industry 4.0 and Smart Manufacturing* pp 211-218

[2] Ali Y, Malik A, May Z 2011 Depth estimation using monocular cues from single image *National Postgraduate Conference (NPC)* pp 1-4.

[3] Chan P P K, Jing B Z, Ng W W Y, Yeung D S 2011 Depth estimation from a single image using defocus cues *International Conference on Machine Learning and Cybernetics (ICMLC)*, pp. 1732-1738.

[4] Haris S M, Zakaria M K, Nuawi M Z 2011 Depth estimation from monocular vision using image edge complexity *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 868-873.

[5] Jung J I and Ho Y S 2010 Depth map estimation from single-view image using object classification based on Bayesian learning *3DTVConference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON)*, 2010 pp. 1-4.

[6] Rizal Y 2019 Computer Simulation of Human-Robot Collaboration in the Context of Industry Revolution 4.0 IntechOpen Open Access Publisher.

[7] Shamshiri R R, Hameed I A, Pitonakove L, Weltzien C, Balasundram S K, Yule I J, Grift T E, Chowdhary G, 2018 Simulation software and virtual environment for acceleration of agricultural robotics: Features highlights and performance comparison, *International Journal of Agricultural and Biological Engineering* pp 15 - 31

[8] Webot (2020). Introduction to Webots. [online] Available at: https://cyberbotics.com/doc/guide/introduction-to-webots [Accessed 25 May 2020].

[9] He L, Yang Y, Kong B, Can W 2017 An Automatic Measurement Method for Absolute Depth of Objects in Two Monocular Images Based on SIFT Feature Applied Science MDPI.

[10] ABB (2020). Data sheet IRB 4600 features and specification. [online] Available at: https://search.abb.com/library/Download.aspx?DocumentID=ROB0109EN_G&LanguageCode=en&DocumentPartId=&Action=Launch