한국항해항만학회지 제48권 제5호
J. Navig. Port Res. Vol. 48, No. 5 : 400-408, October 2024 (ISSN:1598-5725(Print)/ISSN:2093-8470(Online))
DOI : http://dx.doi.org/10.5394/KINPR.2024.48.5.400

Distance Estimation System by using Mono Camera for Warehouse Mobile Robot

Tan Nguyen Duy · Roi Ho Van* · † Hwan-Seong Kim · Yun-Su Ha***

**PhD. Course, Department of Ocean Convergence Logistics Innovation, Korea Maritime and Ocean University, Busan, 49112, KOREA*
***Professor, Division of Artificial Intelligent Engineering, Korea Maritime & Ocean University, Busan, 49112, KOREA*
† Professor, Dept. of Logistics, Korea Maritime and Ocean University, Busan, 49112, KOREA

Abstract: *Achieving high-accuracy distance estimation is critical for mobile robots navigating complex environments, particularly in warehouse settings. This paper introduces an innovative system for distance estimation in warehouse mobile robots, employing a cost-effective approach - a single (mono) camera. The system utilizes chessboard-based calibration to determine the camera's intrinsic parameters, which are then used to accurately estimate distances to objects based on their apparent size in the image. It can calculate the distance from the camera to known objects in real time through perspective geometry. The article also presents experimental results that validate the system's ability to provide precise distance estimations under controlled conditions with minimal error. Advantages of the system include seamless integration with existing robotic platforms, cost-effectiveness, and simplicity. However, the success of the technique depends on the accuracy of the calibration process and the presence of objects with defined dimensions. The potential applications of this system in mobile robotics include obstacle avoidance, object tracking, and indoor navigation.*

Key words : *distance estimation, camera calibration, mobile robots, mono camera, mobile robot*

1. Introduction

Nowadays, mobile robots have played an important role in automatic files such as warehouse and port operations. They have proved their advantages by automatic operation without human control. However, they must overcome certain tackles to improve their accuracy and performance. Especially during the automatic operation in a warehouse where the mobile robot performs their finding path to moving the cargo between the shelves around the warehouse, they need to define the direction, which way it should go, and the distance, which parameters to detect their position. In this case, a distance estimation system could provide accuracy and fast estimation to reduce the lack of time and processing time when choosing the path of mobile robots. This performance enhancement improves mobile robots' limitations, especially when they operate without human control.

In robotics, accurate perception of the environment is crucial for enabling autonomous navigation and interaction with surroundings. Mobile robots must be equipped to sense distances to obstacles or objects for safe and efficient operation. Various sensors such as LIDAR, ultrasonic sensors, and the stereo vision system are commonly used and come with cost, complexity, and computational demand trade-offs. While LIDAR provides high accuracy, it comes with significant cost and power requirements, making it less suitable for low-cost applications (Karthika et al. ,2020). Stereo vision, which utilizes two cameras to estimate depth through triangulation, offers a balance between accuracy and affordability but requires complex calibration and additional computational resources (Liu et al., 2012). Monocular camera-based laser rangefinders offer an economical alternative to expensive laser scanning sensors while providing reliable distance data for mobile robots (Zhang et al., 2013).

Vision-based distance estimation using a single camera has become a promising alternative because of its simplicity and low cost. Mono cameras, however, lack the inherent depth information provided by stereo setups, making distance estimation more challenging. Various methods have been proposed to solve this limitation. Some approaches could be mentioned as depth-from-motion, where camera movement is used to infer depth (Griffin et al., 2021), and size-based estimation, where the apparent size of known objects is used to calculate distance value. One of the applications that could be mentioned by using mono camera calibration is UAV control (Skov et al, 2021), which combines feature detection on a vertical

† Corresponding author, kimhs@kmou.ac.kr 051)410-4334

concrete surface with a camera-based distance estimator, enabling a UAV to autonomously track and approach a user-defined target with a limited margin of error. Another result also applies the vision based method for the 3D Mapping to improve the safety of autonomous driving in container terminals (Vinh et al, 2023).

Chessboard calibration has been widely applied in computer vision fields to increase accuracy when determining a camera's intrinsic parameters, and this parameter can be used to calculate distance using perspective geometry (Xu et al., 2012). This method, which uses images of a chessboard pattern at different angles, has a high performance for the mono camera system and provides a practical solution with high reliability for distance estimation in mobile robots. Several works have focused on and shown that mono camera-based systems can reach trustable results in controlled environments, particularly when calculated properly (Kuramoto, 2018). However, there remain difficulties in extending these methods to more complex environments where real-time performance is required.

This paper describes the problem of distance estimation using a mono camera by applying chessboard-based calibration to find intrinsic camera parameters. These parameters are then used in a perspective geometry framework to estimate the distance between the camera and objects in its field of view. The proposed method provides a cost-effective, lightweight alternative to stereo vision systems, with the added advantage of being easier to implement and integrate into existing robotics platforms. This paper also demonstrates the system provides competitive accuracy with minimal error in controlled environments, making it suitable for a wide range of robotics applications.

The remainder of this paper is organized as follows: Section 2 reviews related work on distance estimation from a mono camera. Section 3 details the approach, methodologies applied, camera calibration, and distance estimation. Section 4 presents experimental results from several tests, while Section 5 concludes with a discussion and potential future work.

2. Literature Review

Distance estimation systems are crucial for performing autonomous navigation in mobile robots. Various methods have recently been developed, from laser-based solutions to vision-only approaches. This paper provides an overview of the relevant research contributing to this area.

Among the well-known approaches for distance estimation in robotics systems, laser rangefinders are among the highest-performance methods. By analyzing the laser beams and their reflections, this system gives LIDAR a high accuracy in distance measurements. Although they can provide great precision, they come at a high cost, making them unsuitable for budget-conscious applications (Muzal et al., 2021). Researchers have investigated alternative methods for tackling this challenge by integrating monocular cameras with laser pointers. For instance, a system for measuring and reconstructing targets using four lasers and a visual camera has been proposed to achieve high-accuracy geometry (Wang et al., 2016). Motion vibrations and computational errors affect the system's performance despite its effectiveness.

Besides laser-based solutions, camera calibration approaches have been employed to estimate distance extensively in monocular camera setups. Chessboard calibration has been widely used to determine camera intrinsic parameters and allow precise perspective projections (Escalera et al., 2010). Various research has also applied this chessboard calibration to estimate distances by calculating the displacement of the image's known reference objects. For example, Xu et al. (2017) proposed a novel visual measurement method using a single camera to estimate 3D positions of objects on the floor, leveraging extrinsic camera parameters and a chessboard pattern for calibration, achieving higher accuracy than the traditional estimation method. However, these methods often struggle with lens distortion, which introduces errors at longer distances.

For depth estimation, several types of methods depend on vision-only approaches. One such method is depth-from-motion, which uses a camera's relative motion to measure depth. Based on that approach, Zhuang et al. (1994) uses a Kalman filter to improve predictions and morphological filtering to lower noise and increase accuracy. This method computes depth maps from monocular image sequences by combining direct depth estimation with optical flow techniques. Although promising, the methods typically involve a sequence of images and extensive calculations, making real-time difficult to execute on mobile robots. Using probabilistic

geometry and object, this system combines local object detection, capturing the dependency of objects, surface orientations, and camera pinhole points. This allows for highly accurate objects and distance estimation (Hoiem et al., 2008). However, this approach has drawbacks, especially when operating in difficult environmental conditions, as its performance is limited. They also require known object sizes or detailed knowledge about the scene.

3. Proposed Methodology

The methodology section outlines the process taken to estimate the distance and coordinates of the camera in the mobile robot of known objects in the camera's field of view. This process is divided into three main stages: camera calibration using a chessboard pattern, calculating the pixel-to-real dimension conversion, and estimating distance in the X, Y, and Z coordinates. These stages are detailed as follows.

3.1 Camera Calibration using a Chessboard Pattern

Camera calibration is an important step in finding the intrinsic parameters of a camera, such as focal length, lens distortion, and optical center, to enhance the accuracy of distance and coordinates measurement. This research applies a chessboard calibration method, a widely used approach due to its simplicity and accuracy.

A classical challenge in computer vision is three-dimensional (3D) reconstruction, which involves extracting 3D structural information from two-dimensional (2D) images of a scene (Forsyth and Ponce, 2015). Since real-world cameras are complex devices, photogrammetry techniques are employed to model the relationship between the measurements captured by the camera's image sensor and the actual 3D world. In the widely used pinhole camera model, the connection between world coordinates X and the widely used pinhole camera model establishes the connection between world coordinates X and image (pixel) coordinates coordinates x is established through perspective transformation by Eq. (1).

$$x = K[R\,t]X \quad , \quad x \in P \quad , \quad X \in P^3 \tag{1}$$

where: P is the projective space of dimension n.

Multiplane calibration is a method of camera auto-calibration that enables the computation of a camera's parameters from two or more views of a flat, planar surface. The foundational work in this area was pioneered by Zhang (2000). The author's technique calibrates cameras by solving a homogeneous linear system that encapsulates the homographic relationships between several perspective views of the same plane. This Multiview approach has gained popularity due to its practical simplicity—it is easier to capture multiple views of a flat surface, such as a chessboard, than to construct a precise 3D calibration rig, which is necessary for Direct Linear Transformation (DLT) calibration. The Figs below illustrate a practical example of multiplane camera calibration using multiple views of a chessboard.

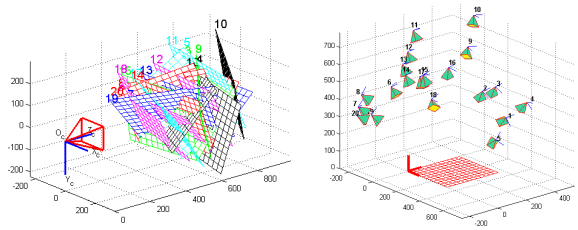

Fig. 1 Reconstructed orientations

Some pinhole cameras provide considerable distortion to images, with two primary types being radial and tangential. Radial distortion results in curved, straight lines, with the effect becoming more pronounced as points move far from the center of the image. For example, within an image, two edges of a chessboard are marked with red lines. However, the actual border of the chessboard does not align with the red lines, illustrating the radial distortion. The expected straight lines bulge outward, highlighting the curvature caused by the distortion. Then, the radial distortion can be calculated as follows:

$$x_{distorted} = x\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right) \tag{2}$$

$$y_{distorted} = y\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right) \tag{3}$$

When the camera lens is not perfectly parallel to the imaging plane, tangential distortion will happen. This misalignment causes certain areas in the image to appear closer to the farther away than expected. Tangential distortion typically results in a slight shift or tilt in the image, making objects appear distorted along the edges. The amount of tangential distortion can be mathematically illustrated by Eq. (4) and Eq. (5):

$$x_{distorted} = x + \left[\left(2p_1 y + p_2\left(r^2 + 2x^2\right)\right)\right] \tag{4}$$

$$y_{distorted} = y + \left[\left(p_1\left(r^2 + 2y^2\right) + 2p_2 x\right)\right] \tag{5}$$

where p_1 and p_2 are tangential distortion coefficients, and r is the radial distance from the center of the image. These formulas account for the deviation caused by the misalignment between the lens and the imaging plane. In short, to correct the distortions in the captured image, there are five distortion coefficients needed to determine, which are typically represented as:

$$Distortion\ coefficients = (k_1\, k_2\, p_1\, p_2\, k_3) \quad (6)$$

where:

- k_1 and k_2: radial distortion coefficients that account for the bulging effect in the image.
- p_1 and p_2: tangential distortion coefficients, which handle the shift due to the lens misalignment
- k_3: an additional radial distortion coefficient that further refines the correction, especially for higher-order distortions.

These coefficients transform the distorted image into its undistorted form, allowing for more accurate measurements and 3D reconstructions of the camera's images.

Intrinsic parameters are specific to a camera and describe its internal characteristics. These include the focal lengths as (f_x, f_y) and the optical center (c_x, c_y). The focal lengths determine how the camera converges light onto the image sensor, while the optical center indicates where the principal axis intersects the image plane. These parameters are combined into a camera matrix, which can be used to calculate lens distortion and increase the accuracy of mapping 3D world coordinates to 2D image coordinates. The camera matrix is unique to a particular camera, so once calculated, it can be applied to all images taken, eliminating the need to repeat the calibration process for future photos. The camera matrix K is demonstrated as a 3x3 matrix by Eq. (7):

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (7)$$

where:

- f_x and f_y are the focal lengths in the x and y directions, respectively.
- c_x and c_y are the optical center coordinates, also known as the principal point.
- The last row maintains the matrix format for homogeneous coordinates.

Extrinsic parameters define the position and orientation of the camera to the world coordinate system

3.2 Calculating Pixel to Real Dimension Conversion

It was necessary to establish a connection between these two scales to convert pixel dimensions into the captured image to real-world units (centimeters). For this process, a label or chessboard pattern with known physical dimensions - w_{lab} in width and h_{lab} in height- was used as a reference object. The camera captured the image of this chessboard pattern (label), and its dimensions in pixels, denoted as L_x (width in pixels) and L_y (height in pixels), were measured from the image.

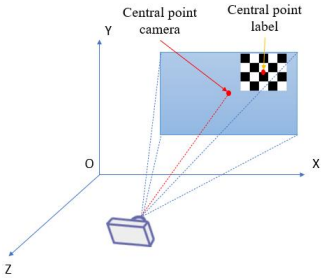

Fig. 2 The relative between the camera position and the label/chessboard

Using this kind of reference chessboard (label) in Fig 2, the pixel-to-real dimension conversion factors for both the x and y directions were computed. The conversion factor p_x for the x direction was calculated from Eq. (8):

$$p_x = \frac{L_x}{w_{lab}} \quad (8)$$

Similarly, the conversion factor p_y for the y direction was computed as Eq. (9):

$$p_y = \frac{L_y}{h_{lab}} \quad (9)$$

These conversation factors represented the physical size of one pixel in centimeters for both directions and were used to transform pixel dimensions into centimeters units. This conversion was important for accurately estimating distances in the real world.

3.3 Estimating Distance in Z, X, and Y Coordinates

The camera position estimation is built from the triangular similarity principle and some equations to convert pixels to real-world dimensions. This section provides the concept and equations used to estimate the

distances in the *z*, *y*, and *x* coordinates, along with the steps to calculate the camera's position relative to the detected objects (chessboard pattern, label).

The focal lengths *fx* and *fy* are calculated by using the triangular similarity mentioned above, which is presented by Eq. (10) and Eq. (11), which relates the size or dimension of an object (this case is a chessboard or label) in the real world to its image in the camera's field of view:

$$\frac{d_{x_{img}}}{f_x} = \frac{d_{x_{obj}}}{R} \tag{10}$$

$$\frac{d_{y_{img}}}{f_y} = \frac{d_{y_{obj}}}{R} \tag{11}$$

where:

- $d_{x_{img}}$: the dimension of the object is in pixels on the x-axis.
- $d_{x_{obj}}$: the known real-world I dimension of the object.
- $d_{y_{img}}$: the dimension of the object is in pixels on the y-axis.
- $d_{y_{obj}}$: the known real-world y dimension of the object.
- R: is the known distance from the camera to the object, measured once.

Rearranging the equation, the f_x and f_y can be solved by using Eq. (12) and Eq. (13):

$$f_x = R \cdot \frac{d_{x_{img}}}{d_{x_{obj}}} \tag{12}$$

$$f_y = R \cdot \frac{d_{y_{img}}}{d_{y_{obj}}} \tag{13}$$

When the focal length f_x and f_y are defined, the distance to the chessboard or label, R, can be estimated based on the chessboard's dimensions in pixels by using Eq. (14):

$$R = \begin{cases} f_x \cdot \dfrac{d_{x_{obj}}}{d_{x_{img}}} \\ f_y \cdot \dfrac{d_{y_{obj}}}{d_{y_{img}}} \end{cases} \tag{14}$$

Once the distance R is calculated, the camera's position is estimated in the x and y axis relative to the detected object.

Determine the difference in pixels between the center of the object and the center of the image for both the x and y axis by Eq. (15) and Eq. (16):

$$\Delta_{x_{pixel}} = c_x - c_{x_{obj}} \tag{15}$$

$$\Delta_{y_{pixel}} = c_y - c_{y_{obj}} \tag{16}$$

where:

- $c_{x_{obj}}$: the center of the object by x-axis in the pixel
- $c_{y_{obj}}$: the center of the object by y-axis in the pixel

Eq. (17) and Eq. (18) convert these pixel differences into real-world distances using triangular similarity:

$$\Delta_{x_{real}} = R \cdot \frac{\Delta_{x_{pixel}}}{f_x} \tag{17}$$

$$\Delta_{y_{real}} = R \cdot \frac{\Delta_{y_{pixel}}}{f_y} \tag{18}$$

The distance in the z-axis, d_z, represents the object's distance from the camera along the optical axis. In this case, d_z also means the position z of the camera on the mobile robot, which can be estimated by using Eq. (19):

$$z_{cam} = L_x \cdot \frac{f_x}{w_{lab}} \tag{19}$$

where:

- L_x is the known dimension of the object in pixels.
- f_x is the focal length by the x-axis of the camera.

which is derived using triangular similarity principles.

Eq. (20) calculates the camera's position based on the real-world differences for the x-coordinate:

$$x_{cam} = \begin{cases} c_x + \left|\Delta_{x_{real}}\right|, & \text{if } \Delta_{x_{real}} \geq 0 \\ c_x - \left|\Delta_{x_{real}}\right|, & \text{if } \Delta_{x_{real}} < 0 \end{cases} \tag{23}$$

Similarly, Eq. (21) calculates the camera position for the y-coordinate:

$$y_{cam} = \begin{cases} c_y + \left|\Delta_{y_{real}}\right|, & \text{if } \Delta_{y_{real}} \geq 0 \\ c_y - \left|\Delta_{y_{real}}\right|, & \text{if } \Delta_{y_{real}} < 0 \end{cases} \tag{24}$$

These computations can determine the camera's position about the object in real-world coordinates. Knowing the spatial relationship between the camera and the surrounding objects makes more precise 3D object detection and tracking possible.

4. Experiment Results

This experiment focuses on the accuracy of the proposed method for estimating the position of the camera and the chessboard in 3D space. To perform this experiment, the coordinate system with known real-world positions for the camera and chessboard is set up as Fig 3. The camera was placed at various positions, and the

real-world coordinates of the camera and the objects were recorded. Using the proposed method, the estimated positions (including the distance from the camera) of the camera were then calculated based on the image captured by the camera and compared with the actual measurements. Also, the camera's resolution is 5 Megapixel. All images captured by this camera have the same characteristics as below:

- 1080x1080 pixels (WidthxHeight).
- 192 dpi for horizontal resolution and vertical solution.

The camera's estimated position was derived using the triangular similarity method described in the methodology section. The pixel-to-centimeter conversion was applied based on the known dimension of the label. The distance in the X, Y, and Z coordinates was estimated for each camera position using the derived focal length and the known real dimensions of the objects. The error between the real and estimated positions was calculated as follows:

The experiment results are summarized in Table 1, which shows the measured and estimated positions of the camera at various locations. These positions are located at different x and z values but have the same y values. Since the mobile robots do not change their y positions, this experiment chooses to keep y-position value the same for all cases.

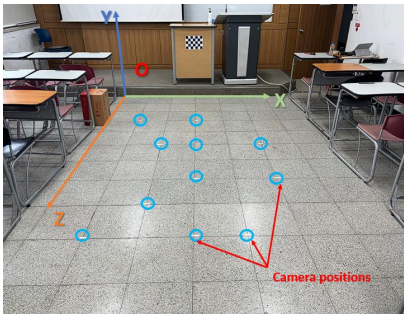

Fig. 3 Camera and label position setup

The average error in the Z-coordinates was 6.5894 cm, while the error in the X and Y coordinates ranged from 3.6312 cm to 9.5887 cm. As seen from the data, the methods provided relatively accurate estimates for positions where the camera was closer to the chessboard; however, the error increased slightly when the camera was positioned at greater distances.

Table 1 The errors between the estimation and the measurement in the normal test

Unit cm	$X_{estimate}$	$Y_{estimate}$	$Z_{estimated}$	$X_{measure}$	$Y_{measure}$	$Z_{measured}$	$\triangle e_x$	$\triangle e_y$	$\triangle e_z$	$\%e_x$	$\%e_y$	$\%e_z$
Image 01	45.9543	19.2300	556.2027	40.2000	36.0000	562.9000	5.7543	16.7700	6.6973	14.31	46.58	1.19
Image 02	125.7963	19.2300	562.9837	110.0000	36.0000	562.8000	15.7963	16.7700	0.1837	14.36	46.58	0.03
Image 03	83.4755	23.3722	522.8723	80.6000	36.0000	528.7000	2.8755	12.6278	5.8277	3.57	35.08	1.10
Image 04	44.1278	16.1212	515.5195	40.2000	36.0000	482.8000	3.9278	19.8788	32.7195	9.77	55.22	6.78
Image 05	166.2333	24.7813	474.8882	160.8000	36.0000	482.8000	5.4333	11.2187	7.9118	3.38	31.16	1.64
Image 06	123.5731	24.9000	441.8613	120.9000	36.0000	442.0000	2.6731	11.1000	0.1387	2.21	30.83	0.03
Image 07	85.7918	25.2871	399.9393	80.2000	36.0000	401.9000	5.5918	10.7129	1.9607	6.97	29.76	0.49
Image 08	205.8873	24.8384	391.0181	201.6000	36.0000	401.9000	4.2873	11.1616	10.8819	2.13	31.00	2.71
Image 09	161.6439	28.2443	360.6929	161.5000	36.0000	361.5000	0.1439	7.7557	0.8071	0.09	21.54	0.22
Image 10	120.4927	27.3124	336.1946	120.8000	36.0000	321.3000	0.3073	8.6876	14.8946	0.25	24.13	4.64
Image 11	204.7919	29.6179	317.9803	201.6000	36.0000	321.8000	3.1919	6.3821	3.8197	1.58	17.73	1.19
Image 12	40.2265	30.5100	278.1305	40.5000	36.0000	281.4000	0.2735	5.4900	3.2695	0.68	15.25	1.16
Image 13	79.2078	30.9127	237.0185	80.1000	36.0000	240.5000	0.8922	5.0873	3.4815	1.11	14.13	1.45
Image 14	164.4307	30.7570	236.2470	160.7000	36.0000	240.9000	3.7307	5.2430	4.6530	2.32	14.56	1.93
Image 15	41.9119	32.9810	196.0874	40.3000	36.0000	201.1000	1.6119	3.0190	5.0126	4.00	8.39	2.49
Image 16	122.6079	34.4862	157.2291	121.0000	36.0000	160.4000	1.6079	1.5138	3.1709	1.33	4.20	1.98

Table 2 The errors between the estimation and the measurement with 20% brightness

Unit cm	$X_{estimate}$	$Y_{estimate}$	$Z_{estimated}$	$X_{measure}$	$Y_{measure}$	$Z_{measured}$	$\triangle e_x$	$\triangle e_y$	$\triangle e_z$	$\%e_x$	$\%e_y$	$\%e_z$
Image 01	45.9543	19.2300	556.1210	40.2000	36.0000	562.9000	5.7543	16.7700	6.7790	14.31	46.58	1.20

Image 02	125.7963	19.2300	562.9000	110.0000	36.0000	562.8000	15.7963	16.7700	0.1000	14.36	46.58	0.02
Image 03	83.4755	23.3722	522.7944	80.6000	36.0000	528.7000	2.8755	12.6278	5.9056	3.57	35.08	1.12
Image 04	47.1602	23.2534	477.9842	40.2000	36.0000	482.8000	6.9602	12.7466	4.8158	17.31	35.41	1.00
Image 05	166.2333	22.8398	482.8904	160.8000	36.0000	482.8000	5.4333	13.1602	0.0904	3.38	36.56	0.02
Image 06	123.5709	24.9000	441.7543	120.9000	36.0000	442.0000	2.6709	11.1000	0.2457	2.21	30.83	0.06
Image 07	85.7918	25.2871	399.8802	80.2000	36.0000	401.9000	5.5918	10.7129	2.0198	6.97	29.76	0.50
Image 08	205.8873	24.8384	390.9599	201.6000	36.0000	401.9000	4.2873	11.1616	10.9401	2.13	31.00	2.72
Image 09	162.4373	28.2269	363.5595	161.5000	36.0000	361.5000	0.9373	7.7731	2.0595	0.58	21.59	0.57
Image 10	120.5380	30.9811	320.1282	120.8000	36.0000	321.3000	0.2620	5.0189	1.1718	0.22	13.94	0.36
Image 11	204.7919	29.8905	317.9333	201.6000	36.0000	321.8000	3.1919	6.1095	3.8667	1.58	16.97	1.20
Image 12	40.2265	30.7500	278.0897	40.5000	36.0000	281.4000	0.2735	5.2500	3.3103	0.68	14.58	1.18
Image 13	79.2078	30.9127	236.9835	80.1000	36.0000	240.5000	0.8922	5.0873	3.5165	1.11	14.13	1.46
Image 14	164.4307	30.7570	236.2116	160.7000	36.0000	240.9000	3.7307	5.2430	4.6884	2.32	14.56	1.95
Image 15	41.9119	32.9810	196.0586	40.3000	36.0000	201.1000	1.6119	3.0190	5.0414	4.00	8.39	2.51
Image 16	122.6079	34.4862	157.2056	121.0000	36.0000	160.4000	1.6079	1.5138	3.1944	1.33	4.20	1.99

Fig. 4 Test cases for camera position estimation with normal brightness

From Fig 4 and the analysis results from Table 1, the estimated positions closely follow the real positions, indicating the proposed method's overall accuracy. The lines connecting the real and estimated points visually represent the error magnitude for each case.

- X-axis: the estimation errors for the X-coordinates are relatively small, and the estimated positions generally remain within a few centimeters of the real positions. The trend line for the X-axis is consistent across the different cases.
- Y-axis: Different from the X-axis, the estimation results differ a larger than X-axis from the real or measured value. These deviations are most noticeable when the camera is far from the chessboard. However, this method was apply for the mobile robot moving on the floors. That means in reality, the y-position robot rarely changes, therefore, this error does not have a strong impact to mobile robot estimation.
- Z-axis (depth): as expected from the above numerical analysis Table 1. The errors on the Z-axis are larger than those on the x and y axes, especially when the camera is farther from the chessboard. However, the overall position trend by the Z-axis still belongs to an acceptable range.

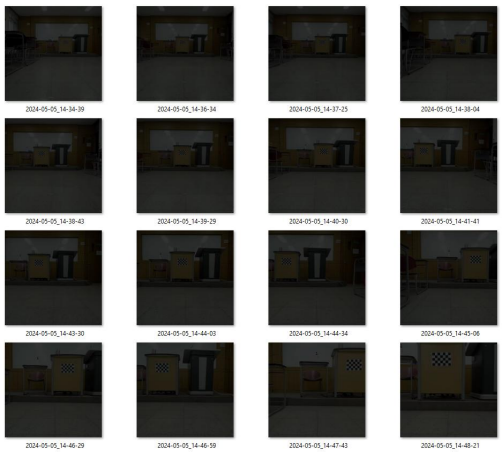

Fig. 5 Test cases for camera position estimation with 20% brightness

The analysis of the test cases above showed that the difference between the estimation and measurement position by the Z-axis was generally larger than in the X and Y coordinates. One of the reasons could be mentioned that depth (Z-axis) estimation strongly depends on small variations in pixel dimensions, which can be impacted by camera distortion and image solution. For instance, when the camera moves far away from the chessboard/labels,

several small changes in pixel dimensions result in larger deviations in the Z-coordinates calculation. This result is consistent with previous studies that underline the drawback of accurate depth estimation from a 2D image.

In contrast, the X and Y coordinate errors were more consistent and comparatively minimal across various camera positions. This can be attributed to the fact that these coordinates rely heavily on the difference between the chessboard's center point and the camera's center point in the image. This calculation is less sensitive to minute pixel changes.

The Fig. 5 illustrates all the test cases with 20% brightness condition serves to visually compare the camera position estimation errors under different lighting conditions. By presenting this figure alongside the numerical data, the purpose is to highlight the impact of reduced brightness on estimation accuracy. The comparison allows for a clearer understanding of how changes in lighting can influence the performance of the estimation model across the X, Y, and Z coordinates.

The results from the camera position estimation evaluation reveal that the average percentage errors for the X, Y, and Z coordinates differ between normal lighting conditions (Table 1) and reduced brightness (20%, Table 2). In Table 1, the average errors across 16 tests are 4.68% for X-axis, 27.81% for Y-axis and 1.71% for Z-axis. Under the 20% brightness condition, as shown in Table 2, the errors across 16 tests are slightly different, with 4.75% for X-axis, a reduced error of 25.01% for Y-axis, and 1.12% for Z-axis.

These results suggest that reducing brightness to 20% had minimal impact on the X-axis estimation accuracy but improved the Y and Z-axis estimations. The significant decrease in error for the Y-axis indicates that lower brightness helped enhance the model's accuracy in estimating positions along this axis. Similarly, the reduced error in Z-axis estimation suggests improved precision in depth estimation under lower brightness. However, since the X-axis error did not show considerable change, it implies that brightness reduction had a limited effect on this coordinate's estimation accuracy. Overall, this analysis demonstrates that brightness conditions can influence camera position estimation accuracy, particularly along the Y and Z axes.

The experiment illustrates the effectiveness of the suggested approach for estimating the camera's position in 3D spaces. The comparatively higher Z-coordinate error indicates that greater improvement could be useful, especially regarding reducing camera distortion and enhancing image quality for a more accurate depth estimate.

5. Conclusions

This research proposed a method for calculating a camera's position in 3D space using chessboard calibration and pixel-to-real unit conversion equation. The experiments demonstrated the proposed approach's effectiveness in accurately estimating distance in the x, y, and z coordinates, focusing on analyzing the errors between the measured and estimated position.

In general, the triangulate similarity principle-based and the pixel-to-world dimension conversion proved reliable methods for estimating the camera's position or camera on mobile robots. Though minor, the error from the experiment indicates the challenges of translating 2D-pixel measurements into accurate 3D world coordinates. Despite these difficulties, this approach proves their accuracy in several scenarios and can be applied to various practical tasks.

In future steps, depth estimation must be improved using high-level approaches, such as multi-point calibration techniques, to minimize estimation errors. Furthermore, some algorithms for updating and optimizing real-time errors could enhance the system's efficiency. These improvements extend the low-cost application's ability to apply to wide and varied scenarios with higher accuracy and performance.

Acknowledgement

This research was supported by “Regional Innovation Strategy (RIS)” through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (MOE)(2023RIS-007).

References

[1] Karthika, K., Adarsh, S. and Ramachandran, K. I.(2020), “Distance estimation of preceding vehicle based on mono vision camera and artificial neural networks”, In 2020 11th International Conference on

Computing, Communication and Networking Technologies (ICCCNT), pp. 1-5, IEEE.

[2] Liu, S., Zhao, L. and Li, J.(2012). "The applications and summary of three dimensional reconstruction based on stereo vision", In 2012 International Conference on Industrial Control and Electronics Engineering, pp. 620-623, IEEE.

[3] Zhang, X., Yang, Y., Liu, Z. and Zhang, J.(2013), "An improved sensor framework of mono-cam based laser rangefinder", Sensors and Actuators A: Physical, 201, 114-126.

[4] Griffin, B. A. and Corso, J. J.(2021), "Depth from camera motion and object detection", In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1397-1406.

[5] Skov, T., Holst, L. B. and Fumagalli, M.(2021), "3D Navigation by UAV using a mono-camera, for precise target tracking for contact inspection of critical infrastructures", In 2021 Aerial Robotic Systems Physically Interacting with the Environment (AIRPHARO), pp. 1-8, IEEE.

[6] Vinh, N. Q., Park, J. H., Shin, H. S. and Kim, H. S.(2023), "3D Mapping for Improving the Safety of Autonomous Driving in Container Terminals", Korean Navigation and Port Research, Vol. 47, No. 5, pp. 281-287.

[7] Xu, H. and Wang, X.(2012), "Camera calibration based on perspective geometry and its application in LDWS", Physics Procedia, 33, pp. 1626-1633.

[8] Kuramoto, A., Aldibaja, M. A., Yanase, R., Kameyama, J., Yoneda, K. and Suganuma, N.(2018), "Mono-camera based 3D object tracking strategy for autonomous vehicles", In 2018 IEEE Intelligent Vehicles Symposium (IV), pp. 459-464, IEEE.

[9] Muzal, M., Zygmunt, M., Knysak, P., Drozd, T. and Jakubaszek, M.(2021). "Methods of precise distance measurements for laser rangefinders with digital acquisition of signals", Sensors, Vol. 21, No. 19, p. 6426.

[10] Wang, F., Dong, H., Chen, Y. and Zheng, N.(2016), "An accurate non-cooperative method for measuring textureless spherical target based on calibrated lasers", Sensors, Vol. 16, No. 12, p. 2097.

[11] De la Escalera, A. and Armingol, J. M.(2010), "Automatic chessboard detection for intrinsic and extrinsic camera parameter calibration", Sensors, Vol. 10, No. 3, pp. 2027-2044.

[12] Xu, L. Y., Cao, Z. Q., Zhao, P. and Zhou, C.(2017), "A new monocular vision measurement method to estimate 3D positions of objects on floor", International Journal of Automation and Computing, Vol. 14, No. 2, pp. 159-168.

[13] Zhuang, H., Sudhakar, R. and Shieh, J. Y.(1994), "Depth estimation from a sequence of monocular images with known camera motion", Robotics and autonomous systems, Vol. 13, No. 2, pp. 87-95.

[14] Hoiem, D., Efros, A. A. and Hebert, M.(2008), "Putting objects in perspective", International Journal of Computer Vision, 80, 3-15.

[15] Forsyth, D. A. and Ponce, J.(2015), Computer Vision: A Modern Approach: International Edition. Pearson Higher Ed, 2015.

[16] Zhang, Z.(2000), "A flexible new technique for camera calibration", IEEE Transactions on pattern analysis and machine intelligence, Vol, 22, No. 11, pp. 1330-1334.

Received 11 October 2024
Revised 22 October 2024
Accepted 28 October 2024