

Implementation of Zhang's Camera Calibration Algorithm on a Single Camera for Accurate Pose Estimation Using ArUco Markers

Junardo Herdiansyah¹, Febi Ariefka Septian Putra^{2,*}, Dwi Septiyanto³

^{1,2,3} Department of Electrical Engineering, Polytechnic State of Bandung, Indonesia

Email: ¹junardo.herdiansyah.toi20@polban.ac.id, ²febi.ariefka@polban.ac.id, ³dwi.septiyanto@polban.ac.id

*Corresponding Author

Abstract—Pose estimation using ArUco markers is a method to estimate the position of ArUco markers relative to the camera lens. Accurate pose estimation is crucial for autonomous systems to navigate robots effectively. This study aims to achieve an ArUco Marker pose estimation accuracy of at least 95% using a single camera. The method employed to obtain accurate ArUco pose estimation results is by calibrating the camera with the Zhang camera calibration algorithm. This calibration is necessary to obtain the camera matrix and distortion coefficients, thereby enhancing the accuracy of the pose estimation results. The results of this study include achieving a cumulative calibration error of 0.0180 pixels and pose estimation errors at a distance of 50 cm between the marker and the camera lens. The accuracy on the X-axis was 100%, the Y-axis was 100%, and the Z-axis was 99.823%. At a distance of 70 cm, the pose estimation accuracy on the X-axis was 99.349%, on the Y-axis was 99.462%, and on the Z-axis was 99.066%. At a distance of 100 cm, the pose estimation accuracy on the X-axis was 96.349%, on the Y-axis was 97.641%, and on the Z-axis was 99.344%.

Keywords—Camera Calibration; Pose Estimation; ArUco Marker; Zhang; Image Processing

I. INTRODUCTION

In the current digital age, using robots to assist human tasks is becoming increasingly common. These robots are designed to perform their tasks independently, without human intervention, known as autonomy. One of the critical components of an autonomous system is computer vision. Computer vision is a technology used to process digital images on robots for navigation purposes. Like eyes, cameras are beneficial and effective in robotics as vision allows for non-contact measurements in various fields, including object recognition, localization, and manipulation [1].

In computer vision, the geometric parameters of a camera, including intrinsic and extrinsic matrices, are calculated for the calibration process [2]. Camera calibration is fundamental in various computer vision applications, encompassing robotics, augmented reality (AR), and autonomous systems [3]. The calibration process aims to determine the camera's intrinsic and extrinsic parameters, which are crucial for transforming 2D images into accurate 3D measurements [4][5]. Camera calibration techniques using structured patterns and deep learning methods have been introduced to enhance the accuracy and efficiency of the calibration process [6][7].

One of the main challenges in camera calibration is ensuring precision under various image capture conditions and different environments. Recent research indicates that camera calibration with dynamic patterns and adaptive methods can significantly improve accuracy [8][9]. Techniques for calibrating single-camera systems enhance the accuracy of real-time 3D mapping [10]. This research demonstrates that adapting more advanced calibration methods can improve the outcomes of applications dependent on high visual accuracy [11]-[14].

One of the camera calibration techniques is the Zhang camera calibration algorithm. This process aims to derive the intrinsic and extrinsic parameters of the camera used for pose estimation. Accurate pose estimation is crucial for robot navigation. However, several factors affect camera calibration accuracy and reliability. These factors include:

- **Lens distortion:** radial and tangential distortions of the camera lens can affect the calibration accuracy and pose estimation. Although the Zhang algorithm is designed to address these distortions, variations in environmental conditions often lead to inconsistent calibration results. Lens distortion frequently results in errors in calculating the position and orientation of objects, especially when objects are at the edge of the camera's field of view.
- **Occlusion and lighting:** occlusion conditions, where part of the marker is obscured by other objects, and extreme lighting variations can affect the detection and identification of ArUco markers. Uneven lighting or sharp shadows can obscure markers, leading to detection and feature extraction errors. This poses a significant challenge in real-time applications requiring fast, accurate detection and measurement.

Camera calibration is crucial for obtaining more accurate measurements in digital image processing. These measurements often use pose estimation to estimate the marker's position relative to the camera [15]. One of the fluid markers for pose estimation is the ArUco marker [16]-[20]. 3D pose estimation is a vital component in computer vision systems, where determining the position and orientation of objects in 3D space is based on 2D images. One key objective is the accuracy of pose estimation [21]. Therefore, accurate pose estimation requires camera calibration [22][23]. One of the problems in pose estimation includes:

- **Pose estimation instability:** in real-world conditions, pose estimation can experience instability due to environmental variations and marker detection errors.

Low accuracy in pose estimation can lead to errors in robot navigation, impacting the overall performance of the autonomous system.

Therefore, this paper aims to achieve accurate pose estimation of ArUco markers using the Zhang camera calibration algorithm, with an average pose estimation accuracy of at least 95% across all measurement distances on the X, Y, and Z axes.

II. SYSTEM DESIGN

In conducting this research, the author used the Python programming language with the OpenCV library [24]. Fig. 1 shows the schematic of the applied system.

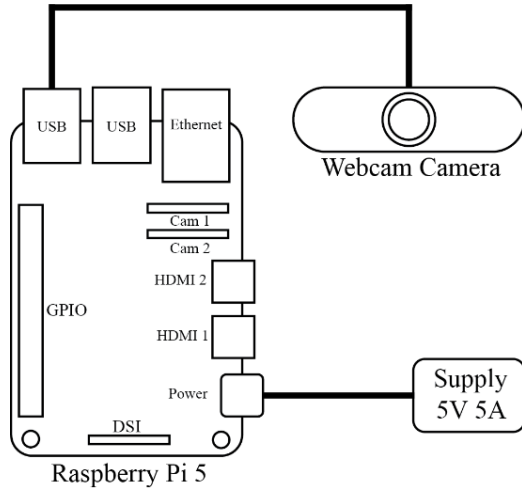


Fig. 1. Schematic of the single camera pose estimation system

Based on Fig. 1 the specifications of the applied system are listed in Table 1.

Table 1. Specifications of the single camera pose estimation system

No	Components	Specifications
1	Raspberry Pi 5	SBC 2.4Ghz, 40 GPIO 25-Watt RAM 4GB, microSD 32GB
2	Webcam Camera	0.9 MP 720P/30FPS
3	Power Adaptor (Supply)	5V 5A

A. Zhang Camera Calibration Algorithm

Camera calibration is a series of processes that obtain the transformation parameters between the camera lens used for capturing images of objects. In other words, camera calibration is a process in digital image processing that accounts for distance, object rotation, and object translation relative to the camera. Camera calibration is crucial for performing distance estimation along three axes: x, y, and z from the camera to the object through visual means. It is essential in 3D computer vision for extracting information from 2D images. A 2D point is denoted as $M = [u, v]^T$. A 3D point is denoted as $M = [x, y, z]^T$, With the z-axis being a zero vector [25]. This calibration process is used to determine camera parameters such as the camera matrix, distortion parameters, image translation vector, and image rotation vector, along with the cumulative calibration error. Mathematically, these parameters are expressed as follows:

$$\text{Camera Matrix} = \begin{bmatrix} F_x & S & C_x \\ 0 & F_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

Where, F_x is the focal length along the X-axis of the camera, F_y is the focal length along the Y-axis of the camera, C_x is the coordinate the principal point along the X-axis of the camera, C_y is the coordinate the principal point along the X-axis of the camera, and S is the Skew parameter that defines the skew between the X and Y axes of the camera.

Based on equation (1), the skew parameter is zero because the camera axes along the X and Y directions are always perpendicular, resulting in a skew value of 0. Thus, the camera matrix is as follows:

$$\text{Camera Matrix} = \begin{bmatrix} F_x & S & C_x \\ 0 & F_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

The mathematical representation of the lens distortion coefficients or parameters is as follows:

$$\text{Distortion parameters} = [K_1 \ K_2 \ P_1 \ P_1 \ K_3] \quad (3)$$

Where, K_1, K_2, K_3 is the Radial distortion coefficients and P_1, P_2 is the Transient distortion coefficients.

K_1 and K_2 are the radial distortion coefficients that indicate the extent of radial distortion in the lens. Radial distortion arises from differences in magnification (scaling) along the radius from the projection center. K_3 is an additional radial distortion coefficient for more complex distortion models (e.g., significant barrel or pincushion distortion).

In contrast, P_1 and P_2 are the tangential distortion coefficients that represent distortion due to the lens's shift from the optical axis. This distortion occurs because the lens is not aligned with the image plane.

Image rotation vector

$$\text{Image rotation vector} = \begin{bmatrix} \Phi \\ \Theta \\ \Psi \end{bmatrix} \quad (4)$$

Where, Φ is the Rotation angle of the image along the X-axis, Θ is the Rotation angle of the image along the Y-axis, and Ψ is the Rotation angle of the image along the Z-axis

The image rotation vector represents the rotation from the object's coordinate system (e.g., a planar pattern such as a checkerboard) to the camera's coordinate system along the X, Y, and Z axes. Image translation vector

$$\text{Image translation vector} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad (5)$$

Where, t_x is the Translation along the X axis, t_y is the Translation along the Y axis, and t_z is the Translation along the Z-axis

The image translation vector is a vector that represents the translation from the object's coordinate system (a planar pattern such as a checkerboard) to the camera's coordinate system along the X, Y, and Z axes. The flowchart for Zhang's camera calibration and the calibration process using Zhang's algorithm are shown in Fig. 2 and Fig. 3, respectively.

Based on the flowchart in Fig. 2, the initial step in calibrating the camera using Zhang's algorithm is to prepare a camera with a planar pattern, such as a checkerboard with 9 x 6 corners. Next, activate the camera to capture RGB images with a 640 x 480 pixels resolution. Then, 35 images of the checkerboard with different orientations relative to the camera were taken.

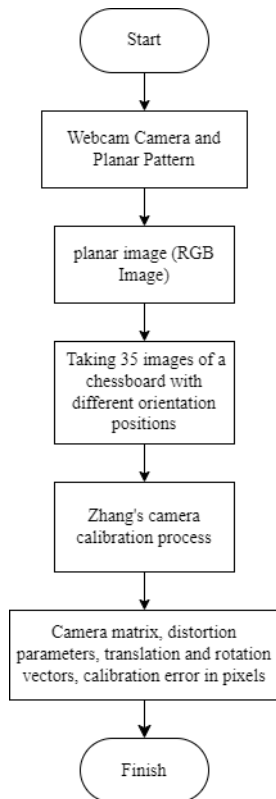


Fig. 2. Camera calibration process using Zhang's algorithm

Proceed with the calibration process using Python with the OpenCV library. Start by declaring the image size as 210mm x 290mm and the checkerboard dimensions as corners (9x5). In this study, the values used are `cv.TERM_CRITERIA_EPS` of 100 and `cv.TERM_CRITERIA_MAX_ITER` of 0.001. Then, configure the program to read images from a folder containing the 35 images. Convert the RGB images to grayscale.

Next, refine or perfect the detected corner locations in the grayscale images. Input the grayscale images and find the corners from the previously detected grayscale images for refinement. Set the window size or search area in pixels for each corner refinement to 11 x 11 pixels. The zero zone, which represents the central area excluded from the search, is set to -1, -1, meaning no area is excluded.

Then, input the criteria for stopping the iterative algorithm, determining when the corner refinement process should stop. The criteria are based on

“`cv.TERM_CRITERIA_EPS`” of one hundred (100) and “`cv.TERM_CRITERIA_MAX_ITER`” of 0.001, where the number 100 represents the iteration count and 0.001 represents the epsilon value.

Perform the calibration with these parameters, yielding results including the camera matrix, distortion parameters, translation and rotation vectors, and calibration error in pixels. Once Zhang's camera calibration is complete, the camera matrix and distortion parameters will be used to improve the accuracy of ArUco marker pose estimation. Fig. 3 shows that the checkerboard corners have been successfully detected, as indicated by the colors displayed.



Fig. 3. Camera calibration process display

B. ArUco Marker and ArUco Marker ID Detection

ArUco Marker is a binary square image with black and white colors used for camera pose estimation. Each ArUco Marker has a unique ID value and supports a maximum of 1024 IDs. An ArUco Marker's ID can be calculated manually or using computer vision. Manual calculation can use schemes such as Hamming code, which involves parity and data bits. Parity bits are error-checking, while data bits are the actual information. An ArUco Marker consists of a maximum 7 x 7 binary grid, with each cell of the ArUco image encoded with bits.

Based on Fig. 4, once the marker is created, the next step is to decode it using the Hamming code scheme to calculate the data bits and account for the parity bits within the marker. Fig. 5 is a representation of the placement of data bits and parity bits in the original ArUco marker.

Fig. 5 shows that the ArUco ID can be calculated by summing the data bits of the binary number. The ID contains the binary number 0000011110, which, when converted to decimal form, equals 30.

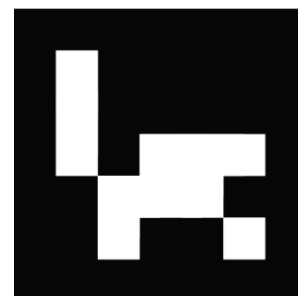


Fig. 4. ArUco marker

Bit Parity	Bit Data	Bit Parity	Bit Data	Bit Parity
1	1	2	2	3
0	0	0	0	0
0	0	0	0	0
0	0	1	1	0
1	1	1	1	0
1	1	0	0	0

Fig. 5. Decoding ArUco marker

C. Image Processing

Digital image processing is a technology that processes digital images or pictures according to the embedded program. In this research, digital image processing aims to read ArUco markers with unique IDs. This processing involves color changes to read ArUco markers. Additionally, detection and pose estimation can be used as a pose estimation marker to obtain the estimated position of the ArUco marker relative to the camera. Therefore, digital image processing for ArUco must include a 3-axis pose estimation for measuring translation along the x, y, and z axes from the center of the marker to the center of the camera lens. The following flowchart illustrates the image processing for identifying the ArUco marker ID and performing pose estimation of the ArUco marker relative to the camera, as shown in Fig. 6.

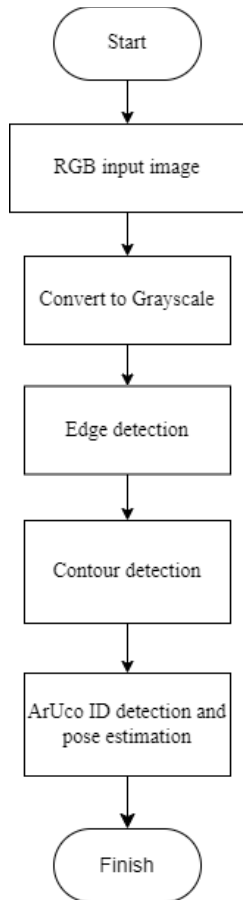
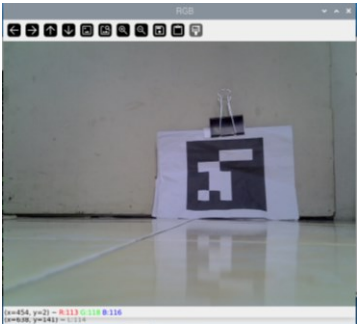
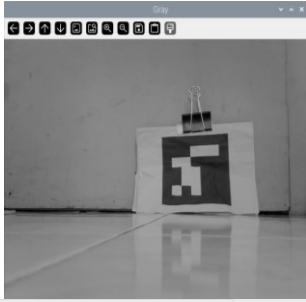
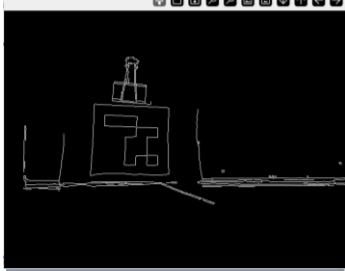
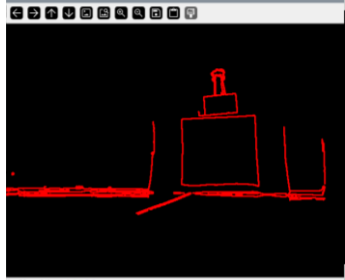


Fig. 6. Flowchart for image processing to identify ArUco marker

Based on Fig. 6, the first step in image processing is to capture an RGB image from the camera, which is then converted to grayscale to reduce data dimensions. Next, the image undergoes edge detection to outline the binary data obtained from the image. In contrast, contour detection focuses on the areas or objects identified in the image and processes them so that the objects can be recognized and understood by the system. The output of this image processing is to determine the ArUco ID and perform pose estimation of the ArUco marker. This process utilizes the OpenCV library.

The pattern shown in Table 2 is the same as the image in Fig. 4 but rotated 90 degrees clockwise. For example, the representation of the rotated ArUco marker is shown in Fig. 7. The image of the detected ArUco with its ID is shown in Fig. 8.

Table 2. Digital image processing process for ArUco marker detection

No	Picture	Explanation
1		RGB Image The RGB image is the input image used for image processing.
2		Grayscale Image The grayscale image is the result of converting from RGB, which serves to reduce data dimensions [26].
3		Edge Detection The edge detection image, or Canny edge detection, is a processed image that functions to represent the binary data obtained from the image
4		Contour Detection The contour detection image is a processed image that focuses on the area or object produced by the image. It is further processed so that the object can be viewed and understood by the system.

Based on the image in Fig. 8, the system reading after image processing to identify the ArUco marker ID matches the manual calculation of the ArUco ID using the Hamming code scheme, with an ID value of 30.

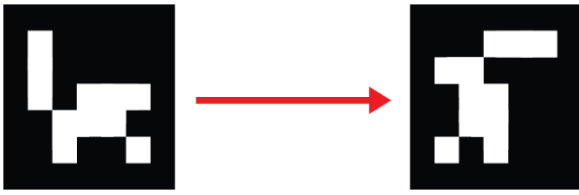


Fig. 7. Rotation of the ArUco Marker 90 Degrees Clockwise

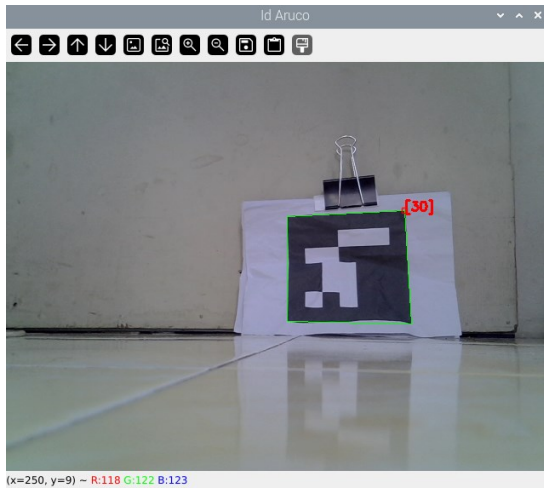


Fig. 8. Reading the ID of the ArUco marker

D. Accuracy Pose Estimation ArUco Marker

Pose estimation is a technique in computer vision used to determine the 3D spatial coordinates of a point, including coordinates on the x, y, and z axes. The use of ArUco markers aims to make it easier for computer systems to recognize them, as ArUco markers contain unique embedded codes. These codes enable the system to accurately identify the marker and estimate the object's pose relative to the camera on the x, y, and z axes. This process involves using camera calibration results, which include intrinsic camera parameters such as the camera matrix and distortion parameters, to achieve more accurate ArUco pose estimation results.

Based on Fig. 9, the display shows the result of pose estimation for the ArUco marker read on the X, Y, and Z axes. Fig. 9 is an example of the pose estimation result for the ArUco marker at a distance of 86 cm. The accuracy formula for ArUco pose estimation is:

$$Accuracy\ pose\ estimation = 100\% - Error\ Relative \tag{6}$$

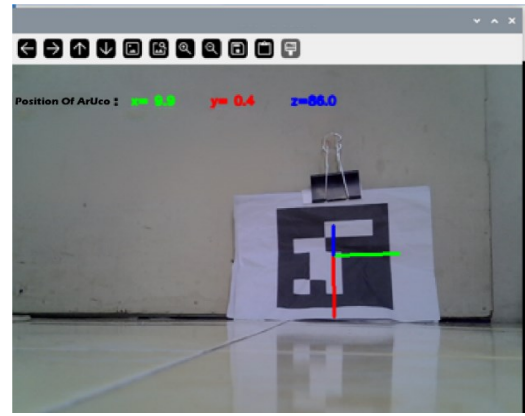


Fig. 9. Pose estimation of ArUco marker

III. RESULT AND DISCUSSION

Based on the results of the design and experiments, the pose estimation results are as follows:

A. Result Zhang's camera calibration algorithm

Based on the experimental results using the Zhang camera calibration algorithm, the intrinsic and extrinsic parameters of the camera are obtained. The intrinsic parameters consist of the camera matrix and lens distortion parameters, while the extrinsic parameters consist of the rotation and translation vectors of the image. The calibration results can be found in Table 3, Table 4, Table 5, and Table 6.

Table 3. Result of matrix camera

No	Parameters	Result		
1	Matrix Camera	698.5167477	0	322.7119836
		0	698.0543131	236.2582937
		0		1

Based on Table 3, the focal length on the camera's X-axis is 698.5167477, the skew parameter is 0 because the X and Y axes of the camera are perpendicular to each other, the focal length on the Y-axis of the camera is 698.0543131, the X coordinate of the camera's principal point is 322.7119836, and the Y coordinate of the camera's principal point is 236.2582937.

Table 4. Result of parameter distortion

No	Parameters	Result
1	Distortion Parameters	-0.11785701583978955
		1.5199508775591315
		-0.016984594698011175
		-0.0014976151455554812
		-5.275252972815059

Based on Table 4, the distortion parameters represent the distortion values of the camera according to formula 3, where the explanations for the numbers are as follows: -0.11785701583978955 is the value of K1, 1.5199508775591315 is the value of K2, -0.016984594698011175 is the value of P1, -0.0014976151455554812 is the value of P2, and -5.275252972815059 is the value of K3. These values were obtained through the camera calibration process with 35 images based on different rotations and translations of the chessboard.

Based on Table 5 and Table 6, these are the results of the calibration values for the camera's extrinsic parameters using the Zhang camera calibration method and for the calibration error using the Zhang calibration algorithm, the calibration error obtained is 0.018006660341099965 pixel.

Table 5. Result of image rotation vector







No	Parameters	Result
1		0.11857564984538077
		-0.01741652085423842
		0.011824271435988583
2		0.11920255053169126
		-0.01810630272594103
		0.011752351712174236
⋮	⋮	⋮
35		-0.5300666165992208
		-0.016183223270571596
		-0.04777238388228286

Table 6. Result of image translation vector

No	Parameters	Result
1		-6.145302070997344
		-5.739821204416082
		28.84203665605388
2		-8.011859774771649
		-2.7335036717816075
		30.459382441788545
⋮	⋮	⋮
35		-5.37787608030147
		-4.472781953410902
		32.6486636145591

B. Pose Estimation Results at a Distance of 50cm from the Camera

The experiment was conducted using three axes: the x-axis, y-axis, and z-axis. For the ArUco pose estimation measurements, the x and y axes were measured at a distance of 50 cm, while the z-axis was measured from 50 cm to 59 cm. Based on the experimental results, the pose estimation results for the x and y axes at a distance of 50 cm can be found in Table 7 and Table 8. Meanwhile, the pose estimation for the z-axis, measured from 50 cm to 59 cm, can be found in Table 9.

Table 7. Results of ArUco pose estimation at a distance of 50 cm on the x-axis

No	Computer (cm)	Actual (cm)	Error	
			Error absolute (cm)	Error relative (%)
1	-10.0	-10.0	0	0
2	-8.0	-8.0	0	0
3	-6.0	-6.0	0	0
4	-4.0	-4.0	0	0
5	-2.0	-2.0	0	0
6	0.0	0.0	0	0
7	2.0	2.0	0	0
8	4.0	4.0	0	0
9	6.0	6.0	0	0
10	8.0	8.0	0	0
11	10.0	10.0	0	0
Mean error			0	0

Based on the experimental results presented in Table 7, at a distance of 50 cm from the camera, the ArUco marker measurement along the X-axis of the marker's center relative to the camera resulted in an average absolute error of 0 cm and a relative error of 0%. According to equation (6), the pose estimation accuracy of the ArUco marker at these distances is 100%. This is because the distance between the marker and the camera, along with the lighting conditions, was adequate for the camera to perform pose estimation accurately, resulting in a high level of accuracy.

Based on the experimental results presented in Table 8, at a distance of 50 cm from the camera, the ArUco marker measurement along the X-axis of the marker's center relative to the camera resulted in an average absolute error of 0 cm and a relative error of 0%. According to equation (6), the pose estimation accuracy of the ArUco marker at these distances is 100%. This is because the distance between the marker and the camera, along with the lighting conditions, was adequate for the camera to perform pose estimation accurately, resulting in a high level of accuracy.

Based on Table 9, the experimental results show that at distances ranging from 50 cm to 59 cm, measurements of the ArUco marker along the X-axis from the camera resulted in an average absolute error of 0.197 cm and a relative error of 0.177%. According to equation (6), the pose estimation accuracy of the ArUco marker at these distances is 99.823%. This high accuracy is due to the suitable distance of the marker from the camera and adequate lighting conditions, which allow the camera to perform pose estimation effectively. Additionally, the accuracy is affected by the camera's low resolution of 0.9 MP.

Based on Fig. 10, as the distance between the marker and the camera increases, both the absolute and relative errors also increase. This is due to the lens's focal distance from the

marker becoming greater, causing the marker to appear increasingly out of focus. Increasing the camera's megapixel resolution beyond the current 0.9 MP could help mitigate this issue.

The relationship between actual readings and system readings is illustrated in Fig. 11. As the marker moves farther away, the error readings increase, as indicated by the linear regression equation. The regression equation has a constant term of -1.376 and a positive regression coefficient of 1.027, suggesting that as the marker moves farther from the camera lens, the absolute error increases. The value of 1.027 represents the gradient of the calibration results. This may indicate that the axial error of the single-camera pose estimation algorithm can be addressed by dividing the given error by this gradient. A better approach to minimize this error might involve reconsidering the camera calibration constants, potentially by increasing the focal length of the camera lens.

Table 8. Results of ArUco pose estimation at a distance of 50 cm on the y-axis

No	Computer (cm)	Actual (cm)	Error	
			Error absolute (cm)	Error relative (%)
1	-6.5	-6.5	0	0
2	-5.2	-5.2	0	0
3	-3.9	-3.9	0	0
4	-2.6	-2.6	0	0
5	-1.3	-1.3	0	0
6	0.0	0.0	0	0
7	1.3	1.3	0	0
8	2.6	2.6	0	0
9	3.9	3.9	0	0
10	5.2	5.2	0	0
Mean error			0	0

Table 9. Results of ArUco pose estimation at distances from 50 cm to 59 cm along the z-axis

No	Computer (cm)	Actual (cm)	Error	
			Error absolute (cm)	Error relative (%)
1	50.0	50.0	0	0
2	51.0	51.0	0	0
3	52.0	52.0	0	0
4	53.1	53.0	0.1	0.188
5	54.1	54.0	0.1	0.185
6	55.1	55.0	0.1	0.182
7	56.2	56.0	0.2	0.357
8	57.2	57.0	0.2	0.351
9	58.2	58.0	0.2	0.345
10	59.2	59.0	0.2	0.339
Mean error			0.197	0.177

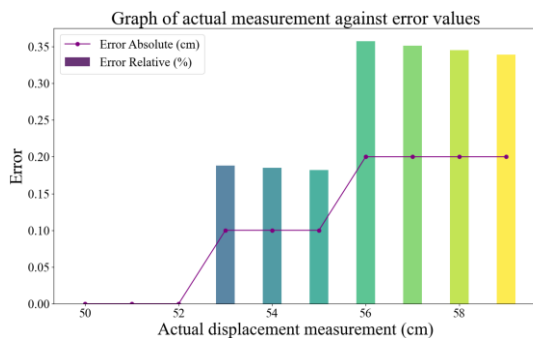


Fig. 10. Graph of absolute and relative errors along the Z-axis at distances from 50 cm to 59 cm

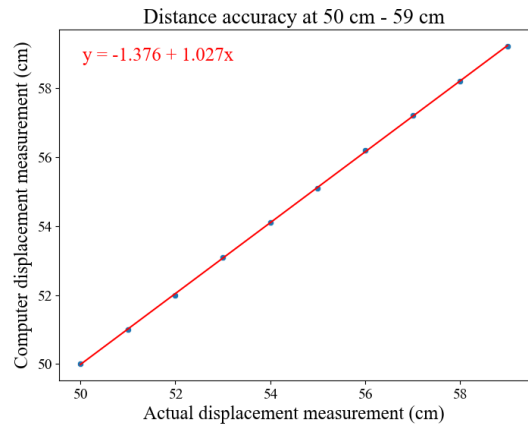


Fig. 11. Graph of absolute and relative errors along the Z-axis at distances from 50 cm to 59 cm

C. Pose Estimation Results at a Distance of 70 cm from the Camera

The experiment was conducted using three axes: the x-axis, y-axis, and z-axis. For the ArUco pose estimation measurements, the x and y axes were measured at a distance of 70 cm, while the z-axis was measured from 70 cm to 79 cm. Based on the experimental results, the pose estimation results for the x and y axes at a distance of 70 cm can be found in Table 10 and Table 11. Meanwhile, the pose estimation for the z-axis, measured from 70 cm to 79 cm, can be found in Table 12.

Table 10. ArUco pose estimation results at a distance of 70 cm on the x-axis

No	Computer (cm)	Actual (cm)	Error	
			Error absolute (cm)	Error relative (%)
1	-10.2	-10.0	0.2	2
2	-8.1	-8.0	0.1	1.25
3	-6.1	-6.0	0.1	1.66
4	-4.0	-4.0	0	0
5	-2.0	-2.0	0	0
6	0.0	0.0	0	0
7	2.0	2.0	0	0
8	4.0	4.0	0	0
9	6.1	6.0	0.1	1.66
10	8.1	8.0	0.1	1.25
11	10.2	10.0	0.2	2
Mean error			0.073	0.893

Based on Table 10, the experimental results indicate that at a distance of 70 cm from the ArUco marker to the camera, the measurement of the X-axis of the marker's center to the camera yields an average absolute error of 0.073 cm and a relative error of 0.893%. According to equation (6), the accuracy of ArUco pose estimation at a distance of 70 cm on the X-axis is 99.107%. This is because the distance between the marker and the camera and the lighting conditions are sufficient for the camera to perform pose estimation, resulting in fairly good accuracy. Additionally, this is also due to the camera's small resolution, which is 0.9 MP.

Based on Fig. 12, the farther the marker is from the camera, the greater the absolute and relative error values. This is because the lens's focal length to the marker increases, causing the marker to become more out of focus. This issue can be mitigated by increasing the camera's megapixel count from the current 0.9 MP.

The relationship between actual readings and system readings is shown in Fig. 13. As the marker moves further from the camera lens center (see the figure), the observed error increases. Referring to the linear regression equation, the error has a relationship with a regression constant of 0.000 and a positive regression coefficient of 1.015 (indicating that as the marker moves further from the camera lens, the absolute error increases). According to the regression equation, the value 1.015 represents the gradient of the calibration results. This may suggest that the axial error from the single-camera pose estimation algorithm can be addressed relatively easily by dividing the given value by this gradient. A better approach to handling this error might involve reconsidering the camera calibration constant. This aims to reduce the error on the X-axis, for example, by increasing the focal length of the camera lens.

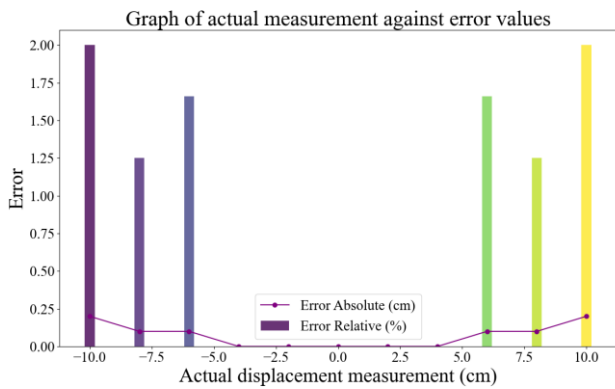


Fig. 12. Graph of absolute and relative errors on the X-axis at a distance of 70 cm

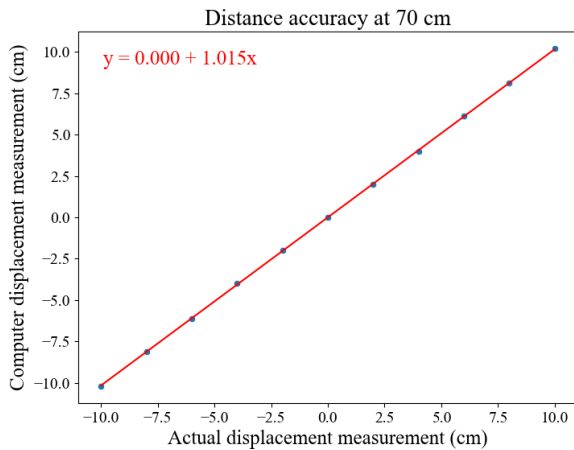


Fig. 13. Graph of the Relationship Between Actual Measurements and System Measurements on the X-Axis at a Distance of 70 cm

Based on Table 11, the experimental results indicate that at a distance of 70 cm from the ArUco marker to the camera, measuring the Y-axis of the marker's center to the camera yields an average absolute error of 0.03 cm and a relative error of 0.538%. According to equation (6), the accuracy of ArUco pose estimation at a distance of 70 cm on the Y-axis is 99.462%. This is because the distance between the marker and the camera and the lighting conditions are sufficient for the camera to perform pose estimation, resulting in fairly good accuracy. This is also due to the camera's small resolution of 0.9 MP.

Table 11. ArUco pose estimation results at a distance of 70 cm on the y-axis

No	Computer (cm)	Actual (cm)	Error	
			Error absolute (cm)	Error Relative (%)
1	-6.6	-6.5	0.1	1.538
2	-5.3	-5.2	0.1	1.923
3	-4.0	-3.9	0	0
4	-2.6	-2.6	0	0
5	-1.3	-1.3	0	0
6	0.0	0.0	0	0
7	1.3	1.3	0	0
8	2.6	2.6	0	0
9	4.0	3.9	0	0
10	5.3	5.2	0.1	1.923
Mean error			0.03	0.538

Based on Fig. 14, the farther the marker is from the camera, the greater the absolute and relative error values. This is because the focal length of the lens to the marker increases, causing the marker to become more out of focus. This issue can be mitigated by increasing the camera's megapixel count from the current 0.9 MP.

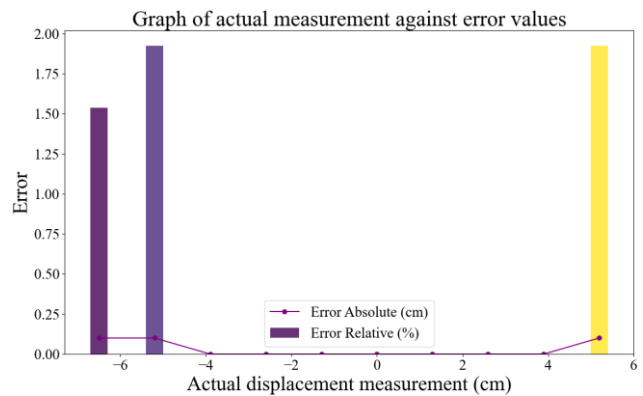


Fig. 14. Graph of absolute and relative errors on the Y-axis with a distance of 70cm

The relationship between actual readings and system readings is shown in Fig. 15. As the marker moves farther from the center of the camera lens (see the figure), the observed error increases. Referring to the linear regression equation, the error has a relationship with a regression constant of 0.001 and a positive regression coefficient of 1.017 (indicating that as the marker moves further from the camera lens, the absolute error increases). According to the regression equation, the value 1.017 represents the gradient of the calibration results. This suggests that the axial error from the single-camera pose estimation algorithm can be easily addressed by dividing the given value by this gradient. A better approach to handling this error might involve reconsidering the camera calibration constant. This aims to reduce the error on the Y-axis, for example, by increasing the focal length of the camera lens.

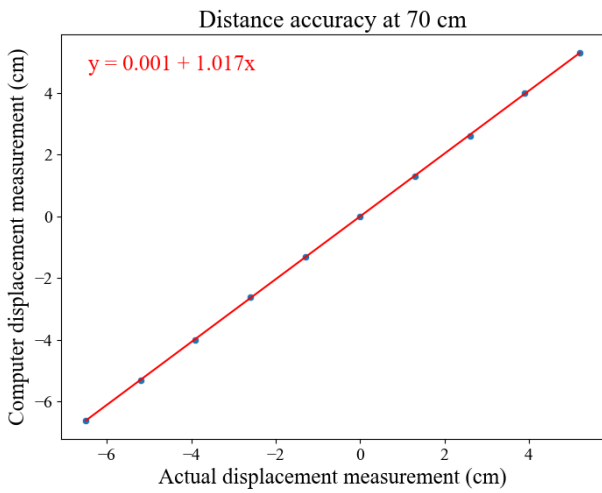


Fig. 15. Graph of the Relationship Between Actual Measurements and System Measurements on the Y-Axis at a Distance of 70 cm

Based on the experimental results presented in Table 12, it was found that at a distance of 70 cm from the ArUco marker to the camera, measuring the Z-axis of the marker's center to the camera yields an average absolute error of 0.77 cm and a relative error of 0.934%. According to equation (6), the accuracy of ArUco pose estimation at a distance of 71 cm to 80 cm is 99.066%. This is because the distance between the marker and the camera and the lighting conditions are sufficient for the camera to perform pose estimation, resulting in fairly good accuracy. This is also due to the camera's small resolution of 0.9 MP.

Table 12. ArUco pose estimation results at a distance of 70 cm to 79 cm on the z-axis

No	Computer (cm)	Actual (cm)	Error	
			Error absolute (cm)	Error Relative (%)
1	70.5	70.0	0.5	0.704
2	71.6	71.0	0.6	0.833
3	72.6	72.0	0.6	0.822
4	73.6	73.0	0.6	0.811
5	74.7	74.0	0.7	0.933
6	75.7	75.0	0.7	0.921
7	76.8	76.0	0.8	1.031
8	77.8	77.0	0.8	1.025
9	78.9	78.0	0.9	1.139
10	79.9	79.0	0.9	1.125
Mean error			0.71	0.934

Based on Fig. 16, the farther the marker is from the camera, the greater the absolute and relative error values. This is because the focal length of the lens to the marker increases, causing the marker to become more out of focus. Alternatively, this issue can be mitigated by increasing the camera's megapixel count from the current 0.9 MP.

The relationship between actual readings and system readings is shown in Fig. 17. As the marker moves farther from the center of the camera lens (see the figure), the observed error increases. Referring to the linear regression equation, the error is related by the equation $-2.586 + 1.044X$ with a positive regression coefficient of 1.044 (indicating that as the marker moves further from the camera lens, the absolute error increases). According to the regression equation, the value 1.044 represents the gradient of the calibration results. This may

suggest that the axial error from the single-camera pose estimation algorithm can be relatively easily addressed by dividing the given value by this gradient. A better approach to handling this error might involve reconsidering the camera calibration constant. This aims to reduce the error on the Z-axis, for example, by increasing the focal length of the camera lens.

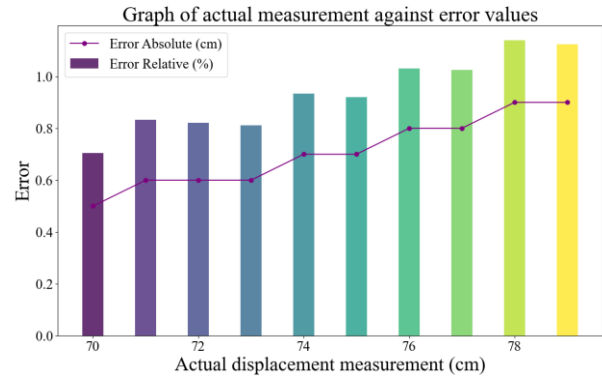


Fig. 16. Graph of Absolute and Relative Error on the Z-Axis at Distances of 70 cm to 79 cm

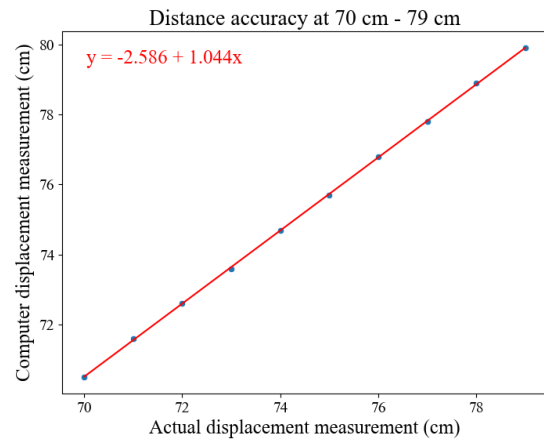


Fig. 17. Graph of the Relationship Between Actual Measurements and System Measurements on the Z-Axis at Distances of 70 cm to 79 cm

D. Pose Estimation Results at a Distance of 100 cm from the Camera

The experiment was conducted using three axes: the x-axis, y-axis, and z-axis. For the ArUco pose estimation measurements, the x and y axes were measured at a distance of 100 cm, while the z-axis was measured from 100 cm to 109 cm. Based on the experimental results, the pose estimation results for the x and y axes at a distance of 100 cm can be found in Table 13 and Table 14. Meanwhile, the pose estimation for the z-axis, measured from 100 cm to 109 cm, can be found in Table 15.

Based on Table 13, the experimental results indicate that at a distance of 100 cm from the ArUco marker to the camera, measuring the X-axis of the marker's center to the camera yields an average absolute error of 0.2 cm and a relative error of 3.651%. According to equation (6), the accuracy of ArUco pose estimation at a distance of 100 cm on the X-axis is 96.349%. This is because the distance between the marker and the camera, as well as the lighting conditions, are sufficient for the camera to perform pose estimation, resulting

in fairly good accuracy. Additionally, this is also due to the camera's small resolution, which is 0.9 MP.

Table 13. ArUco pose estimation results at distance of 100 cm on the x-axis

No	Computer (cm)	Actual (cm)	Error	
			Error absolute (cm)	Error Relative (%)
1	-10.3	-10.0	0.3	3
2	-8.3	-8.0	0.3	3.75
3	-6.2	-6.0	0.2	3.33
4	-4.2	-4.0	0.2	5
5	-2.1	-2.0	0.1	5
6	0.0	0.0	0	0
7	2.6	2.0	0.1	5
8	4.6	4.0	0.2	5
9	6.2	6.0	0.2	3.33
10	8.3	8.0	0.3	3.75
11	10.3	10.0	0.3	3
Mean error			0.2	3.651

Based on Fig. 18, the farther the distance of the marker from the camera, the greater the absolute and relative error values. This is because the focal length of the lens to the marker increases, causing the marker to become more out of focus. Alternatively, this issue could be mitigated by increasing the camera's megapixel count from the current 0.9 MP.

The relationship between actual readings and system readings is shown in Fig. 19. As the marker moves farther from the center of the camera lens (see the figure), the observed error increases. Referring to the linear regression equation, the error is related by the equation $0.082+1.040X$ with a positive regression coefficient of 1.040 (indicating that as the marker moves further from the camera lens, the absolute error increases). According to the regression equation, the value 1.040 represents the gradient of the calibration results. This may suggest that the axial error from the single camera pose estimation algorithm can be relatively easily addressed by dividing the given value by this gradient. A better approach to handling this error might involve reconsidering the camera calibration constant. This aims to reduce the error on the X-axis, for example, by increasing the focal length of the camera lens.

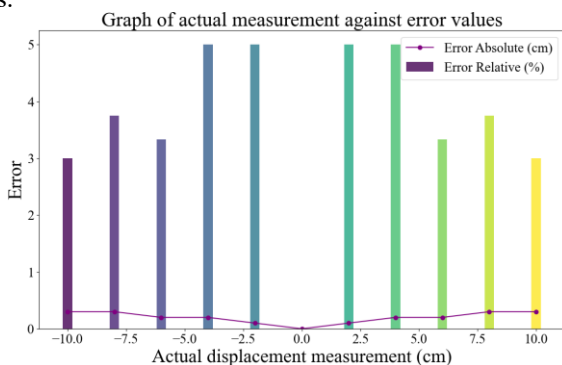


Fig. 18. Graph of Absolute and Relative Error on the X-Axis at a Distance of 100 cm

Based on Table 14, the experimental results indicate that at a distance of 100 cm from the ArUco marker to the camera, measuring the Y-axis of the marker's center to the camera

yields an average absolute error of 0.1 cm and a relative error of 2.359%. According to equation (6), the accuracy of ArUco pose estimation at a distance of 100 cm on the Y-axis is 97.641%. This is because the distance between the marker and the camera, as well as the lighting conditions, are sufficient for the camera to perform pose estimation, resulting in fairly good accuracy. Additionally, this is also due to the camera's small resolution, which is 0.9 MP.

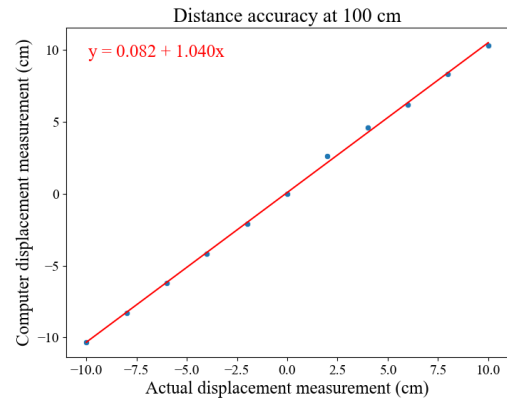


Fig. 19. Graph of the Relationship Between Actual Measurements and System Measurements on the X-Axis at a Distance of 100 cm

Table 14. ArUco pose estimation results at distance of 100 cm on the y-axis

No	Computer (cm)	Actual (cm)	Error	
			Error absolute (cm)	Error Relative (%)
1	-6.7	-6.5	0.2	3.077
2	-5.4	-5.2	0.2	3.846
3	-4.0	-3.9	0.1	2.564
4	-2.7	-2.6	0.1	3.846
5	-1.3	-1.3	0	0
6	0.0	0.0	0	0
7	1.3	1.3	0	0
8	2.7	2.6	0.1	3.846
9	4.0	3.9	0.1	2.564
10	5.4	5.2	0.2	3.846
Mean error			0.1	2.359

Based on Fig. 20, the farther the distance of the marker from the camera, the greater the absolute and relative error values. This is because the focal length of the lens to the marker increases, causing the marker to become more out of focus. Alternatively, this issue could be mitigated by increasing the camera's megapixel count from the current 0.9 MP.

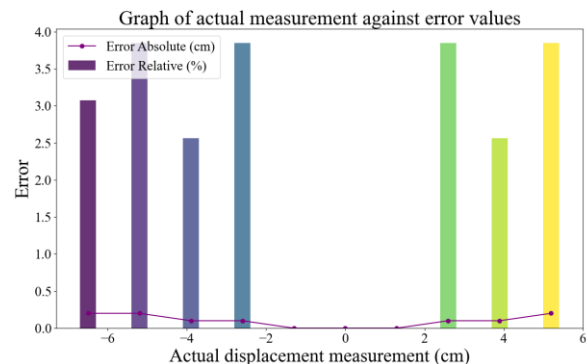


Fig. 20. Graph of Absolute and Relative Error on the Y-Axis at a Distance of 100 cm

The relationship between actual readings and system readings is shown in Fig. 21. As the marker moves farther from the center of the camera lens (see the figure), the observed error increases. Referring to the linear regression equation, the error is related by the equation $0.001+1.033X$ with a positive regression coefficient of 1.033 (indicating that as the marker moves further from the camera lens, the absolute error increases). According to the regression equation, the value 1.033 represents the gradient of the calibration results. This may suggest that the axial error from the single-camera pose estimation algorithm can be relatively easily addressed by dividing the given value by this gradient. A better approach to handling this error might involve reconsidering the camera calibration constant. This aims to reduce the error on the Y-axis, for example, by increasing the focal length of the camera lens.

Based on the results in Table 15, the experimental process indicates that at a distance of 100 cm from the ArUco marker to the camera, measuring the Z-axis of the marker's center to the camera yields an average absolute error of 1.77 cm and a relative error of 1.656%. According to equation (6), the accuracy of ArUco pose estimation at a distance of 100 cm to 109 cm is 98.344%. This is because the distance between the marker and the camera, as well as the lighting conditions, are sufficient for the camera to perform pose estimation, resulting in fairly good accuracy. Additionally, this is also due to the camera's small resolution, which is 0.9 MP.

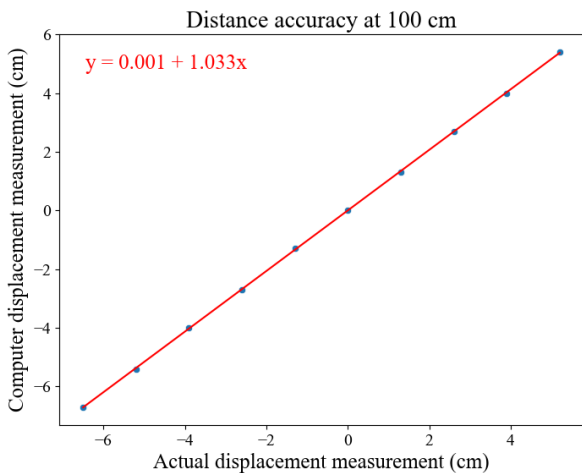


Fig. 21. Graph of the Relationship Between Actual Measurements and System Measurements on the Y-Axis at a Distance of 100 cm to 109 cm

Table 15. ArUco pose estimation results at a distance of 100 cm to 109 cm on the z-axis

No	Computer (cm)	Actual (cm)	Error	
			Error absolute (cm)	Error Relative (%)
1	101.3	100	1.3	1.283
2	102.3	101	1.3	1.270
3	103.4	102	1.4	1.354
4	104.5	103	1.5	1.435
5	105.7	104	1.7	1.608
6	106.7	105	1.7	1.593
7	107.9	106	1.9	1.761
8	109.2	107	2.2	2.014
9	110.3	108	2.3	2.085
10	111.4	109	2.4	2.154
Mean Error			1.77	1.656

Based on Fig. 22, the farther the distance of the marker from the camera, the greater the absolute and relative error values. This is because the focal length of the lens to the marker increases, causing the marker to become more out of focus. Alternatively, this issue can be mitigated by increasing the camera's megapixel count from the current 0.9 MP.

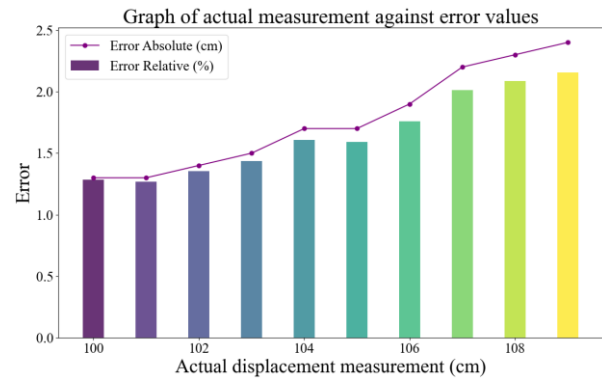


Fig. 22. Graph of Absolute and Relative Error on the Z-Axis at Distances of 100 cm to 109 cm

The relationship between the actual readings and system readings is illustrated in Fig. 23. As the marker moves further from the camera lens center (see the image), the error increases. According to the linear regression equation, the error has a relationship with a regression constant of -12.227 and a positive regression coefficient (indicating that as the marker moves further from the camera lens, the absolute error increases) with a value of 1.134. Referring to the regression equation, the value of 1.134 represents the gradient of the calibration results. This suggests that axial errors from the single-camera pose estimation algorithm can be relatively easily addressed by dividing the given value by this gradient. A better approach to handling this error might involve reassessing the camera's calibration constant. This could involve, for example, increasing the focal length of the camera lens to reduce errors along the Z-axis.

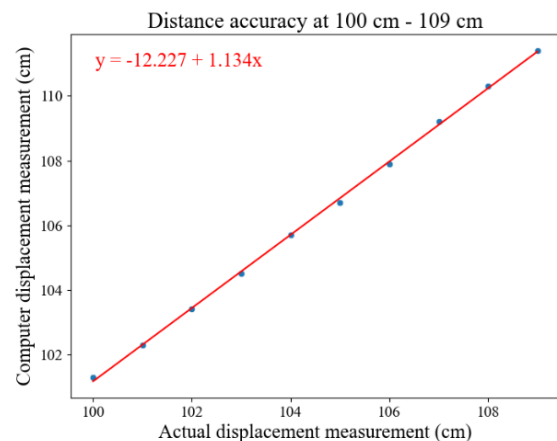


Fig. 23. Graph showing the relationship between actual measurements and system measurements on the Z-axis at distances ranging from 100 cm to 109 cm

Based on previous research [27], at a distance of 50 cm, the accuracy measurements were 99.8% for the x and y axes and 99.5% for the z axis. At a distance of 70 cm, the accuracy was 99.5% for the x-axis, 99.6% for the y-axis, and 98.8% for

the z-axis. At a distance of 100 cm, the accuracy was 95.8% for the x-axis, 96.7% for the y-axis, and 98.7% for the z-axis. In contrast, the current study achieved accuracy values at a distance of 50 cm of 100% for the x and y axes and 99.823% for the z-axis. At 70 cm, the accuracy values were 99.107% for the x-axis, 99.462% for the y-axis, and 99.066% for the z-axis. At 100 cm, the accuracy values were 96.349% for the x-axis, 97.641% for the y-axis, and 99.344% for the z-axis. This indicates that the accuracy in this study has improved by approximately 0.05% to 0.1%. This improvement is likely due to the effective camera calibration using the Zhang method.

IV. CONCLUSION

Based on the conducted experiments, implementing camera calibration using the Zhang algorithm has proven to provide accurate distance measurements by accounting for intrinsic camera values, such as the camera matrix and distortion parameters. The results demonstrate that for a measurement at 50 cm, ArUco pose estimation on both the X and Y axes achieved an average error of 0 cm, resulting in a pose estimation accuracy of 100%. On the Z-axis, for distances ranging from 50 cm to 59 cm, the system recorded an average error of 0.197 cm (1.97 mm) with an accuracy of 99.823%. For a measurement at 70 cm, the X-axis pose estimation showed an average error of 0.073 cm with an accuracy of 99.107%, while the Y-axis had an error of 0.03 cm and an accuracy of 99.462%. On the Z-axis, with distances between 70 cm and 79 cm, the system recorded an average error of 0.71 cm (7 mm) and an accuracy of 99.066%. At 100 cm, the X-axis estimation had an average error of 0.2 cm and an accuracy of 96.349%, while the Y-axis reported a 0.1 cm error with an accuracy of 97.641%. On the Z-axis, the system achieved an average error of 1.77 cm (17.7 mm) with a high accuracy of 99.344%.

ACKNOWLEDGMENT

Acknowledgments to the Polytechnic State of Bandung for providing the facilities that enabled the author to conduct this research.

REFERENCES

- [1] P. Corke. *Robotics and control: fundamental algorithms in MATLAB®*. vol. 141. springer Nature. 2021. <https://books.google.co.id/books?hl=id&lr=&id=NXBJEAAAQBAJ>.
- [2] D. A. Forsyth and J. Ponce. *Computer vision: a modern approach*. prentice hall professional technical reference. 2002. <https://dl.acm.org/doi/abs/10.5555/580035>.
- [3] F. Ababsa and M. Mallem, "Robust camera pose estimation using 2D fiducials tracking for real-time augmented reality systems," *Int. J. Image Graphics*, vol. 4, no. 4, pp. 643-661, 2004, <https://doi.org/10.1145/1044588.1044682>.
- [4] I. A. Aguilar, A. C. Sementille, and S. R. Sanches, "ARStudio: A low-cost virtual studio based on Augmented Reality for video production," *Multimedia Tools and Applications*, vol. 78, pp. 33899-33920, 2019, <https://doi.org/10.1007/s11042-019-08064-4>.
- [5] J. Beltrán, C. Guindel, A. de la Escalera and F. García, "Automatic Extrinsic Calibration Method for LiDAR and Camera Sensor Setups," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 10, pp. 17677-17689, 2022, <https://doi.org/10.1109/TITS.2022.3155228>.
- [6] S. Lee, S. Shim, H. -G. Ha, H. Lee and J. Hong, "Simultaneous Optimization of Patient-Image Registration and Hand-Eye Calibration for Accurate Augmented Reality in Surgery," in *IEEE Transactions on Biomedical Engineering*, vol. 67, no. 9, pp. 2669-2682, 2020, <https://doi.org/10.1109/TBME.2020.2967802>.
- [7] S. Wu, A. Hadachi, D. Vivet and Y. Prabhakar, "NetCalib: A Novel Approach for LiDAR-Camera Auto-calibration Based on Deep Learning," *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 6648-6655, 2021, <https://doi.org/10.1109/ICPR48806.2021.9412653>.
- [8] M. Cao, L. Zheng, W. Jia, H. Lu and X. Liu, "Accurate 3-D Reconstruction Under IoT Environments and Its Applications to Augmented Reality," in *IEEE Transactions on Industrial Informatics*, vol. 17, no. 3, pp. 2090-2100, 2021, <https://doi.org/10.1109/TII.2020.3016393>.
- [9] J. Li, Z. Chen, G. Rao and J. Xu, "Structured Light-Based Visual Servoing for Robotic Pipe Welding Pose Optimization," in *IEEE Access*, vol. 7, pp. 138327-138340, 2019, <https://doi.org/10.1109/ACCESS.2019.2943248>.
- [10] T. Cavallari *et al.*, "Real-Time RGB-D Camera Pose Estimation in Novel Scenes Using a Relocalisation Cascade," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 10, pp. 2465-2477, 2020, <https://doi.org/10.1109/TPAMI.2019.2915068>.
- [11] D. J. Yeong, G. Velasco-Hernandez, J. Barry, and J. Walsh, "Sensor and sensor fusion technology in autonomous vehicles: A review," *Sensors*, vol. 21, no. 6, p. 2140, 2021, <https://doi.org/10.3390/s21062140>.
- [12] L. Tao, R. Xia, J. Zhao, T. Zhang, Y. Chen and S. Fu, "A Convenient and High-Accuracy Multicamera Calibration Method Based on Imperfect Spherical Objects," in *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1-15, 2021, <https://doi.org/10.1109/TIM.2021.3113132>.
- [13] A. Assadzadeh, M. Arashpour, A. Bab-Hadiashar, T. Ngo, and H. Li, "Automatic far-field camera calibration for construction scene analysis," *Computer-Aided Civil and Infrastructure Engineering*, vol. 36, no. 8, pp. 1073-1090, 2021, <https://doi.org/10.1111/micc.12660>.
- [14] B. Nagy, L. Kovács and C. Benedek, "Online Targetless End-to-End Camera-LIDAR Self-calibration," *2019 16th International Conference on Machine Vision Applications (MVA)*, pp. 1-6, 2019, <https://doi.org/10.23919/MVA.2019.8757887>.
- [15] M. Kalaitzakis, S. Carroll, A. Ambrosi, C. Whitehead and N. Vitzilaios, "Experimental Comparison of Fiducial Markers for Pose Estimation," *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 781-789, 2020, <https://doi.org/10.1109/ICUAS48674.2020.9213977>.
- [16] G. Čepon, D. Očepek, M. Kodrič, M. Demšar, T. Bregar, and M. Boltežar, "Impact-pose estimation using ArUco markers in structural dynamics," *Experimental Techniques*, vol. 48, no. 2, pp. 369-380, 2024, <https://doi.org/10.1007/s40799-023-00646-0>.
- [17] A. T. Duchowski, V. Peysakhovich, and K. Krejtz, "Using Pose Estimation to Map Gaze to Detected Fiducial Markers," *Procedia Computer Science*, vol. 176, pp. 3771-3779, 2020, <https://doi.org/10.1016/j.procs.2020.09.010>.
- [18] Y. Wang, Z. Zheng, Z. Su, G. Yang, Z. Wang and Y. Luo, "An Improved ArUco Marker for Monocular Vision Ranging," *2020 Chinese Control And Decision Conference (CCDC)*, pp. 2915-2919, 2020, <https://doi.org/10.1109/CCDC49329.2020.9164176>.
- [19] Z. Zhou, W. Tang, Z. Wang, L. Wang and R. Zhang, "Multi-robot Real-time Cooperative Localization Based on High-speed Feature Detection and Two-stage Filtering," *2021 IEEE International Conference on Real-time Computing and Robotics (RCAR)*, pp. 690-696, 2021, <https://doi.org/10.1109/RCAR52367.2021.9517423>.
- [20] S. Roos-Hoefgeest, I. A. Garcia and R. C. Gonzalez, "Mobile robot localization in industrial environments using a ring of cameras and ArUco markers," *IECON 2021 - 47th Annual Conference of the IEEE Industrial Electronics Society*, pp. 1-6, 2021, <https://doi.org/10.1109/IECON48115.2021.9589442>.
- [21] A. Marut, K. Wojtowicz and K. Falkowski, "ArUco markers pose estimation in UAV landing aid system," *2019 IEEE 5th International Workshop on Metrology for AeroSpace (MetroAeroSpace)*, pp. 261-266, 2019, <https://doi.org/10.1109/MetroAeroSpace.2019.8869572>.
- [22] I. de Medeiros Esper, O. Smolkin, M. Manko, A. Popov, P. J. From, and A. Mason, "Evaluation of RGB-D multi-camera pose estimation for 3D reconstruction," *Applied Sciences*, vol. 12, no. 9, p. 4134, 2022, <https://doi.org/10.3390/app12094134>.

- [23] J. L. Pulloquina, D. Corrata, V. Mata, A. Valera, and M. Vallés, "Experimental Analysis of Pose Estimation Based on ArUco Markers," In *International Conference Innovation in Engineering*, pp. 138-149, 2024, https://doi.org/10.1007/978-3-031-61582-5_12.
- [24] J. Howse and J. Minichino. *Learning OpenCV 4 Computer Vision with Python 3: Get to grips with tools, techniques, and algorithms for computer vision and machine learning*. Packt Publishing Ltd. 2020. https://books.google.co.id/books?hl=id&lr=&id=ef_RDwAAQBAJ.
- [25] B. Huang, Y. Tang, S. Ozdemir and H. Ling, "A Fast and Flexible Projector-Camera Calibration System," in *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 3, pp. 1049-1063, 2021, <https://doi.org/10.1109/TASE.2020.2994223>.
- [26] A. D. A. Zakawali, E. Loniza, M. Safitri, dan M. A. Baballe, "Evaluating the Impact of Cliplimit Parameters and Viewing Distance on Image Clarity in Vein Viewer," *Journal of Fuzzy Systems and Control*, vol. 1, no. 3, 2023, <https://doi.org/10.59247/jfsc.v2i1.173>.
- [27] O. Kedilioglu, T. M. Bocco, M. Landesberger, A. Rizzo and J. Franke, "ArUcoE: Enhanced ArUco Marker," *2021 21st International Conference on Control, Automation and Systems (ICCAS)*, pp. 878-881, 2021, <https://doi.org/10.23919/ICCAS52745.2021.9650050>.