

Article

High Accuracy and Wide Range Recognition of Micro AR Markers with Dynamic Camera Parameter Control [†]

Daisuke Haraguchi ^{*,‡}  and Ryu Miyahara [‡]

National Institute of Technology, Tokyo College, 1220-2, Kunugida-machi, Hachioji 193-0997, Tokyo, Japan

* Correspondence: haraguchi@tokyo-ct.ac.jp

[†] This paper is an extended version of our paper published in the abstract book of KRIS 2023.[‡] These authors contributed equally to this work.

Abstract: This paper presents a novel dynamic camera parameter control method for the position and posture estimation of highly miniaturized AR markers (micro AR markers) using a low-cost general camera. The proposed method captures images from the camera at each cycle and detects markers from these images. Subsequently, it performs iterative calculations of the marker's position and posture to converge them to a specified accuracy while dynamically updating the camera's zoom, focus, and other parameter values based on the detected marker's depth distances. For a 10 mm square micro AR marker, the proposed system demonstrated recognition accuracy better than $\pm 1.0\%$ for depth distance and 2.5° for posture angle, with a maximum recognition range of 1.0 m. In addition, the iterative calculation time was 0.7 s for the initial detection of the marker. These experimental results indicate that the proposed method and system can be applied to the precise robotic handling of small objects at a low cost.

Keywords: image recognition; micro AR marker; camera parameter control; iterative recognition



Citation: Haraguchi, D.; Miyahara, R. High Accuracy and Wide Range Recognition of Micro AR Markers with Dynamic Camera Parameter Control. *Electronics* **2023**, *12*, 4398. <https://doi.org/10.3390/electronics12214398>

Academic Editor: Beiwen Li

Received: 8 August 2023

Revised: 10 October 2023

Accepted: 21 October 2023

Published: 24 October 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent years, robots have been used to automate many tasks to improve productivity in manufacturing and other production sites. Real-time object position and posture estimation using image processing is an essential function for autonomous robots that perform object handling in automation. There are various methods for estimating object position and posture using image processing.

Methods using stereo cameras or RGB-D cameras (RGB-D cameras are sensors capable of capturing both color images and the depth distance to objects) can estimate the position, posture, and shape of an object from multiple RGB images or depth images, enabling the handling of general objects without the need to process the handling target. There have been many attempts to estimate the position and posture of objects using machine learning [1–5]. However, in general, the above methods have disadvantages, such as high implementation costs due to the large number of datasets required and the time required for data learning. Therefore, they are not suitable for applications that require low-cost and low-computational resources.

Visual markers are support tools that facilitate object identification and position and posture estimation. Because visual markers use known shape information and image features, they can be used to identify marker IDs and estimate relative positions and postures from a single 2D camera. Although there is the restriction that the marker must be fixed to the target object, this is inexpensive to implement. In addition, various marker projection patterns have been presented for various applications [6–10].

Among visual markers, AR markers have the advantage of easily enabling augmented reality and providing information to users in a more intuitive manner. Most AR markers use the principle of projective transformation to estimate the position and posture of the

marker by finding a homogeneous transformation matrix that represents the position and posture of the marker's coordinate system as seen from the camera coordinate system. The open source library ARToolKit markers [11] is a typical example, and this marker technology is used in self-position estimation [12–17] and mapping [18–20] for mobile robots, and is an essential technology in navigation systems.

AR markers can also be used to handle specific objects [21–23]. However, the size and recognition accuracy of AR markers are problematic when handling small objects or objects with complex shapes. Conventional AR markers require a large amount of space to be attached to the target object, and it is undesirable from an aesthetic point of view for the markers to be too conspicuous in a real environment. Several previous studies have attempted to reduce the size of markers or improve their aesthetics. Zhang et al. [24] developed a curved surface marker that can be attached to cylindrical objects as small as 6 mm in diameter, enabling real-time tracking of ultrasound probes. However, these markers can only be recognized at a depth of 30–125 mm from the camera, making them unsuitable for object handling that requires a large workspace. Costanza et al. [25] created a marker that is unobtrusive to users in their living environment with “d-touch”, an open source system that allows users to create their own markers based on their aesthetic sense. However, there is no concrete verification of miniaturization or recognition accuracy.

It is also difficult to consistently achieve the recognition accuracy required for accurate object handling. To solve this problem, Douchamps et al. [26] improved accuracy and robustness by physically increasing the marker size and using high-density patterns to reduce noise and discretization in marker recognition. This method can recognize markers at a maximum of 0.06–4 ppm; however, the miniaturization of the marker becomes a trade-off issue. Yoon et al. [27] presented a coordinate transformation algorithm to obtain the globally optimal camera posture from local transformations of multiple markers, thus improving the accuracy of pose estimation. Yu et al. [28] presented a robust pose estimation algorithm using multiple AR markers and demonstrated its effectiveness in real-time AR tracking. Hayakawa et al. [29] presented a 3D toothbrush positioning method that recognizes AR markers on each face of a dodecahedron attached to a toothbrush and achieved a motion tracking rate of over 99.5%. However, these methods require multiple markers to achieve high recognition accuracy, which requires a large space to attach them to objects.

There are two methods to improve recognition performance with a single marker while maintaining the marker size: using filters and using circular dots as feature points. The method using a filter [30,31] reduces jitter between frames and stabilizes posture recognition, but does not guarantee accurate recognition. In contrast, Bergamasco et al. [32,33] achieved robustness against occlusion using markers that use the projection characteristics of a circular set of dots and an ellipticity algorithm. In addition to circular dots, Tanaka et al. [34–37] presented an AR marker that uses lenticular lenses or microlens arrays to change the pattern depending on the viewing angle, thereby reducing the posture estimation error and improving robustness against distance and illumination changes. These techniques have dramatically improved the recognition performance of a single marker, thereby enhancing its practicality. Therefore, these techniques are also promising for marker miniaturization but have not yet been demonstrated.

However, even if marker miniaturization and high recognition accuracy can be achieved, a commonly used camera system limits the range of marker recognition, making practical operation difficult. Because the markers are small, they cannot be recognized with high accuracy from an overhead view of the workspace. Conversely, a magnified view narrows the field of view, making it difficult to recognize the surrounding environment necessary for handling. To solve this problem, Toyoura et al. [38] presented a monospectrum marker that enables real-time detection of blurred images, thereby extending the recognition range. However, the recognition of translational positions has an average error of 5 to 10 mm, which does not meet the level of recognition accuracy required for object handling. Another disadvantage is that the system requires a high-performance GPU for real-time detection.

Based on the background described above, this study will develop a prototype micro AR marker of 10 mm per side [39] that is compatible with the high-accuracy recognition method of Tanaka et al. [37], and construct a low-cost and high-accuracy marker recognition system using a general-purpose web camera. The micro AR marker is printed on a glass substrate by photolithography with a high resolution so that the marker image is not easily degraded even when the marker is magnified by a camera. First, we demonstrate that this AR marker inherently has very high accuracy in position and posture recognition despite its ultra-compact size. But, we also reveal the problem of insufficient recognition range for practical use with a conventional camera system. Next, to solve this problem, we present a dynamic camera parameter control method that can maintain high recognition accuracy over a wide field of view and demonstrate its effectiveness through a series of experiments.

This paper is organized as follows. Section 1 describes the background and objectives of this study. Section 2 describes the overall system configuration. Section 3 describes the process of the proposed camera control system, i.e., the algorithm for camera parameter optimization. Section 4 describes the results of the evaluation experiments of the proposed camera control system. Section 5 discusses the results of the evaluation experiments. Finally, Section 6 describes the summary of this paper and future issues.

2. System Configuration

2.1. Hardware Configuration

The hardware of the camera control system proposed in this paper consists of three pieces of hardware: an AR marker, a single RGB camera, and a PC as a processing unit. The AR marker used is a micro AR marker of 10 mm per side, which is compatible with the Tanaka et al. [37] high-precision recognition method shown in Figure 1. As shown in Figure 1, this marker is equivalent in size to a USB Type-A. The marker was printed on a glass substrate by photolithography with high resolution so that the marker image was not easily degraded even when viewed under a camera's magnification. The camera captures reference points at the four corners of this marker, and by processing the images arithmetically, as shown in Figure 2, it is possible to estimate the relative position and posture of the marker as seen from the camera with high accuracy. A USB 3.0 webcam was used as the single RGB camera. Table 1 shows the camera performance. The proposed dynamic camera control system requires zoom and focus adjustment functions and a wide diagonal viewing angle. Therefore, a BRIO C1000eR[®] (Logitech International S.A., Lausanne, Switzerland, Figure 3) with digital zoom and focus adjustment functions and a maximum diagonal viewing angle of 90 degrees was used. However, when the camera's diagonal viewing angle was set to 90 degrees, the output image from this camera was highly distorted, which required removal. To prepare the camera calibration data that would serve as a reference for removing the distortion, calibration was performed using OpenCV's camera calibration function [40], with the camera's autofocus function turned on. In addition, for the camera control system described in Section 3.1, calibration was performed with the camera's autofocus function turned off to prepare a set of calibration data for each focus value. A Think-Pad X1 Carbon (i7-6600U 2.6GHz, 16 GB memory, Lenovo, Hong Kong, China) was used as the processing PC.

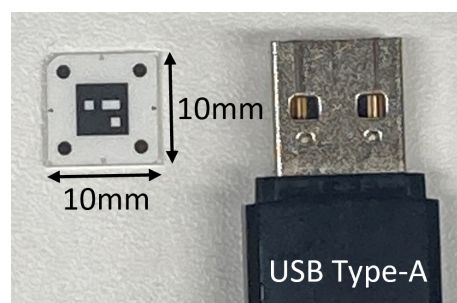


Figure 1. The 10 mm square micro AR marker prototyped in this study.

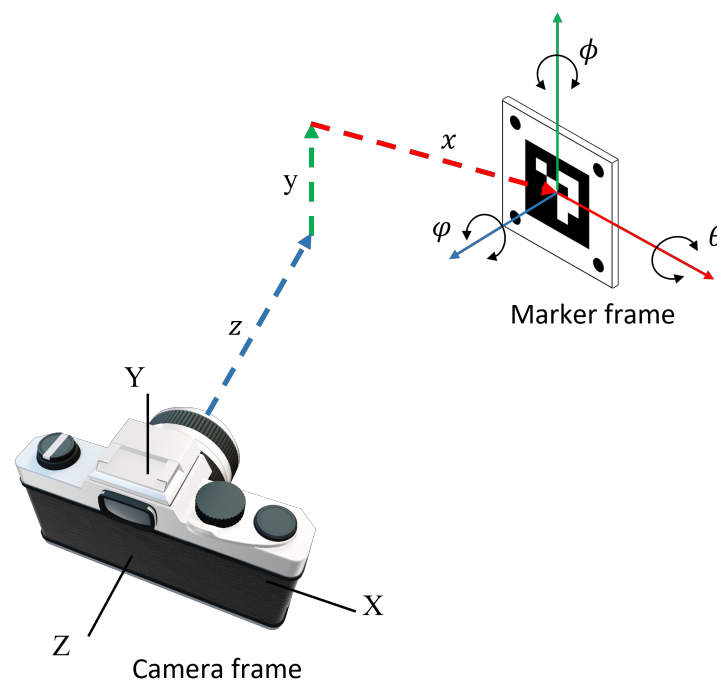


Figure 2. Definition of marker position and posture with respect to camera.

Table 1. The camera specifications.

Product name	BRIO C1000eR [®]
Output resolution	1920 × 1080 (FHD)
Frame rate	30 fps
Diagonal FOV	90°
Digital zoom	1×–5×
Size (mm)	102 × 27 × 27



Figure 3. BRIO C1000eR[®].

2.2. Software Configuration

Figure 4 shows the software configuration of the camera control system proposed in this paper. ROS Melodic was used as the middleware for the development of this study. The software used for AR marker recognition was LEAG-Library from LEAG Solutions Corp. This library is compatible with the high-accuracy recognition method of Tanaka et al. [37]. It also uses OpenCV as an image processing library and uvc-camera as a library for acquiring images from USB cameras. The Relay node relays the AR marker values received from the LEAG Library. It identifies marker IDs and performs type conversion. The Iteration node determines camera parameters such as zoom, focus, and calibration data based on the AR marker values received from the Relay node and provides input to the camera and feedback to the LEAG-Library. It also determines the convergence of the AR marker position and posture and the final AR marker value. Topic communication was used to send and receive AR marker information between nodes. Figure 5 shows the correlation diagram of

topic communication between nodes obtained from the ROS function `rqt_graph`. The mid node acts as the Relay node shown in Figure 4, passing AR marker values to the iteration node using a topic named `tf_mid`. Each node is written in C/C++ language for high-speed processing. The cycle of topics delivered from the LEAG Library is approximately 30 Hz, which is the same as the frame rate of the camera.

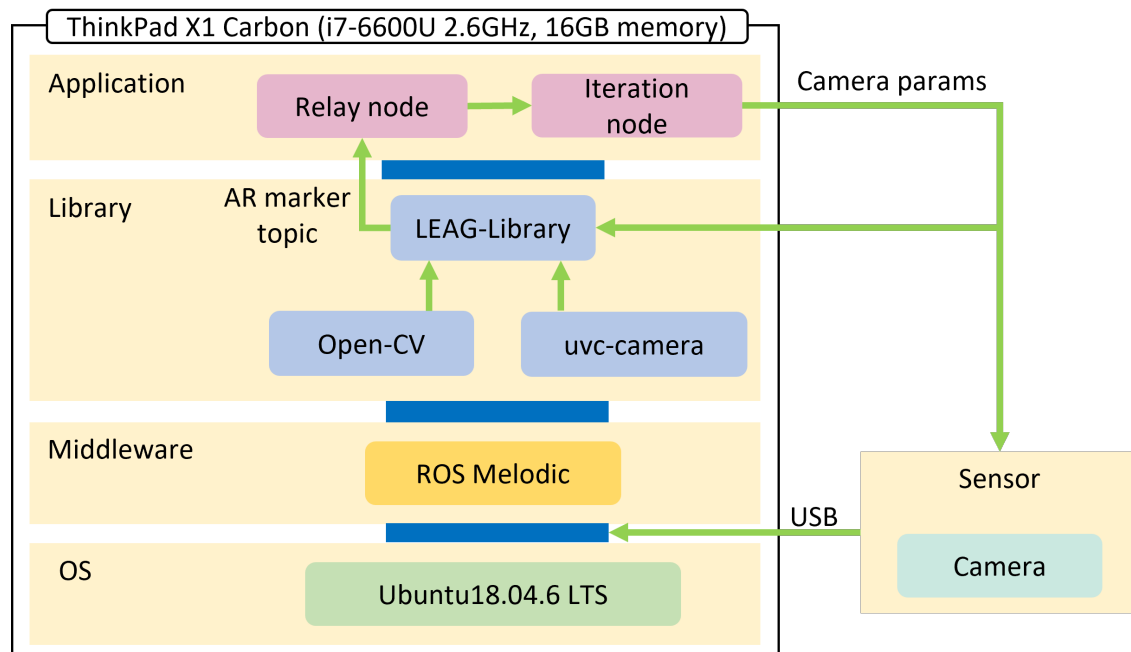


Figure 4. System architecture.

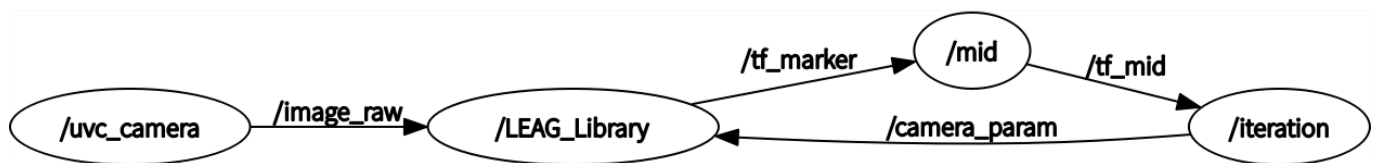


Figure 5. Correlation diagram of the ROS nodes.

3. Camera Control System

This section describes the dynamic camera control processes that properly adjust the parameters of zoom, focus, and calibration data to determine the position and posture of the AR markers. The three parameters of the zoom, focus, and calibration data are collectively referred to as “camera parameters”.

3.1. Marker Recognition Process

As shown in Figure 6, the proposed camera control system consists of two processes: the Scanning process scans the camera’s shooting range to detect AR markers; the Iteration process optimizes camera parameters based on the detected AR marker positions to determine the final AR marker position and posture. Here, it is assumed that the AR marker is still present within the camera’s shooting range when the camera parameters are changed.

Table 2 shows the camera parameters used in this system and the parameters used to calculate the camera parameters. As will be discussed later in Section 4.4, when recognizing micro AR markers, if the camera zoom function is not used, the size of the AR marker in the image will be very small and the recognition range of the AR marker will be narrow. In addition, when the zoom magnification is large, the recognition accuracy of the position of AR markers in close proximity to the camera is poor. Therefore, it is necessary to dynamically control the zoom value to achieve a wide recognition range. The zoom value W is proportional to the depth distance z from the camera of the AR marker shown in

Figure 2 and is calculated by Equation (1). The camera magnification n is also proportional to W , as shown in Equation (2).

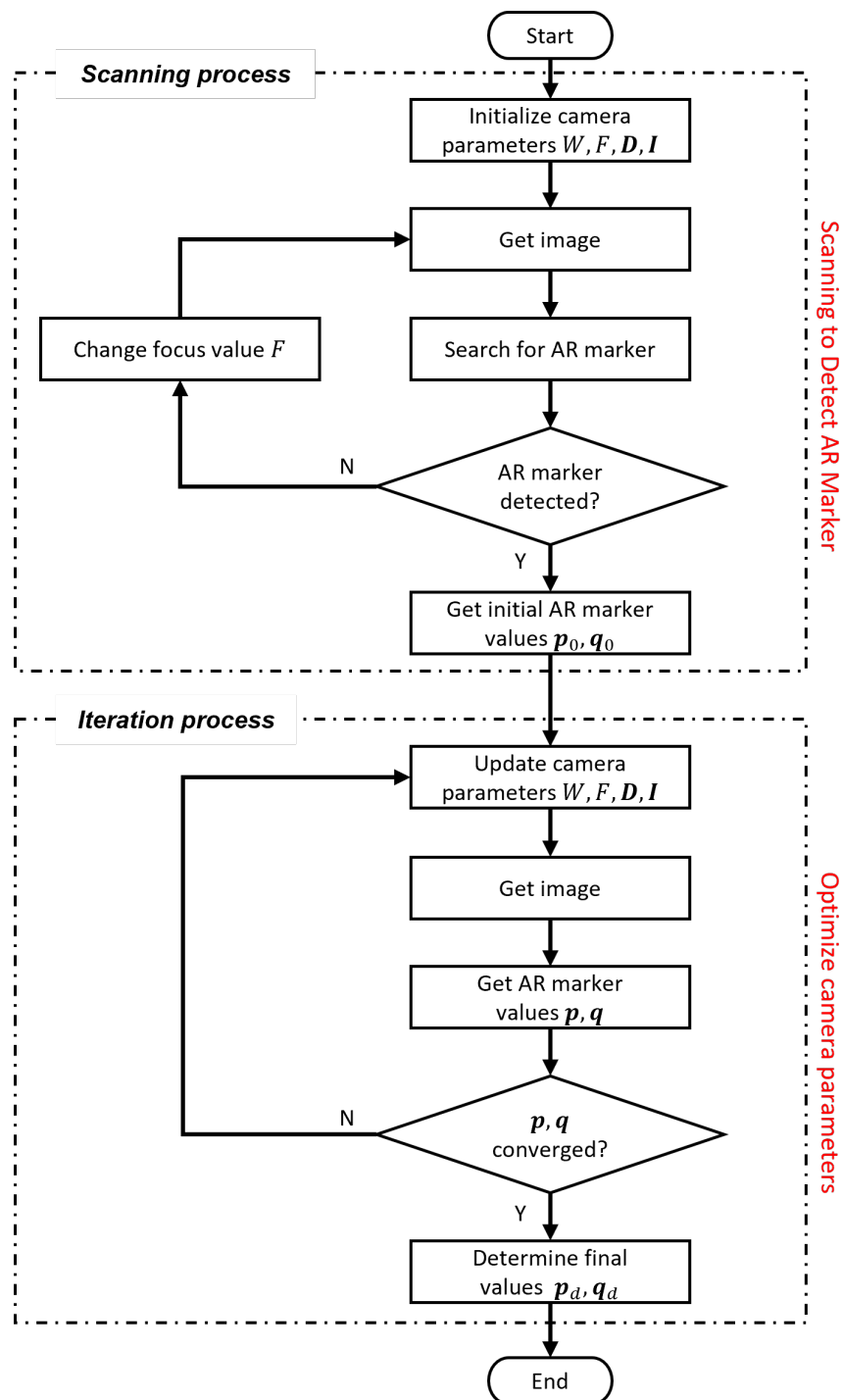


Figure 6. Flowchart of the proposed method.

$$W = C(z - z_{min}) + W_{min} \quad (1)$$

$$n = \frac{W}{100} \quad (2)$$

As in the data presented in Section 4.3, a fixed-focus camera has a narrow range for recognizing AR markers of only 0.2 m in the depth distance. Thus, we attempted to recognize AR markers using autofocus cameras. However, with this approach, the autofocus tended

to focus on the background, which often had higher contrast than the markers themselves. As a result, we could not consistently achieve an accurate focus on the small 10 mm markers. Therefore, it was necessary to explicitly control the focus to achieve a wide recognition range. The focus value F is also determined by the marker's depth distance z using Equation (3). The smaller the focus value, the farther away the camera. The constants α and β used in Equation (3) are constants obtained by measuring the optimum focus value according to the depth distance from the camera in advance and by exponential approximation of the experimental results.

$$F = -\alpha \ln z + \beta \quad (3)$$

Table 2. Parameters used in the proposed camera control system.

Parameter	Symbol	Value
Zoom value	W	Variable (100~500)
Maximum zoom value	W_{max}	500
Minimum zoom value	W_{min}	100
Depth distance of the AR marker	z	Variable (m)
Minimum depth distance	z_{min}	0.05 (m)
Conversion coefficient of the zoom value	C	1000 (1/m)
Camera magnification	n	Variable
Focus value	F	Variable (0~150)
Constants of the focus value	α	42.11
Constants of the focus value	β	4.0
Distortion coefficient matrix	$D_{1 \times 5}$	Determined by calibration
Radial distortion coefficient	k_1, k_2, k_3	
Tangential distortion coefficient	l_1, l_2	
Internal parameter matrix	$I_{3 \times 3}$	Determined by calibration
Focus distance	f_x, f_y	
Optical center	c_x, c_y	
AR marker position	$p = [x \ y \ z]^T$	Iterative output variable
AR marker posture	$q = [\theta \ \phi \ \psi]^T$	Iterative output variable
Threshold of position error	t_p	Arbitrally setting
Threshold of posture error	t_q	Arbitrally setting

The calibration data also include the distortion coefficient matrix D and the internal parameter matrix I of the camera. The distortion coefficient matrix D refers to the lens distortion and is represented by a 1×5 matrix containing the radial distortion coefficients (k_1, k_2, k_3) and tangential distortion coefficients (l_1, l_2) as in Equation (4). The internal parameter matrix I refers to camera-specific parameters and is represented as shown in Equation (5) in a 3×3 matrix containing the focus distance (f_x, f_y) and optical center (c_x, c_y). In this study, 31 calibration data sets for every increment of 5 in the focus value were prepared. The system selects the most appropriate calibration data based on the calculated focus value and applies them to the next recognition process.

$$D = [k_1 \ k_2 \ l_1 \ l_2 \ k_3] \quad (4)$$

$$I = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (5)$$

The two processes shown in Figure 6 are described in detail below.

In the Scanning process, AR markers are initially detected by scanning within the camera's shooting range; the Scanning process follows the steps below.

- (1a) The camera parameters are set to the initial values $W = 500$ and $F = 150$, with the zoom at its maximum and the focus at its farthest forward. The reason for these settings is that the zoom value is farther away for a wider recognition range, and the focus value can be processed faster if it is shifted from near to far.
- (1b) Detect AR markers within a single frame of the image output from the camera.
- (1c) If no AR marker is found in the image, the focus value F is reduced by 15 to focus on a more distant point.
- (1d) Repeat steps (1b)–(1c) until AR markers are detected.
- (1e) If an AR marker is detected for the first time, obtain the initial position p_0 and posture q_0 of the AR marker to be given as initial values for the next Iteration process.

According to the above algorithm, the scanning takes a maximum of 11 frames of images before the AR marker is detected. Since the frame rate of the camera used is 30 fps, the maximum scanning time is theoretically about 0.33 s. In reality, however, even if the AR marker was at the farthest point within the recognition range, the scanning time was only about 0.3 s. This is because the AR marker could be detected even when the focus position was in front of the AR marker, and detection was possible from the tenth frame of the image.

In the *Iteration process*, the camera parameters are optimized to determine the final marker position p_d and posture q_d with enhanced accuracy; the *Iteration process* follows the steps below.

- (2a) Receive the initial recognition values p_0 and q_0 of the AR marker from the *Scanning process*.
- (2b) Update the camera parameters based on the recognized depth distance z of the AR marker.
- (2c) Get the next recognition values p_k and q_k with the updated camera parameters.
- (2d) Calculate the absolute error value between p_{k-1} and p_k , and between q_{k-1} and q_k . If the error values are larger than the thresholds t_p and t_q , repeat steps (2b)–(2c).
- (2e) If the absolute error values calculated in step (2d) are smaller than the thresholds t_p and t_q , the latest p_k and q_k are output as the final recognition values p_d and q_d .

Software algorithms of the aforementioned two processes are described in Algorithms 1 and 2.

Algorithm 1 Scanning Process

Input: RGB image

Output: p_0, q_0

Initialization:

1: $W \leftarrow 500, F \leftarrow 150, D, I \leftarrow$ Autofocus values

2: Get RGB image

Loop Process:

3: **while** AR marker is **not** detected in RGB image **do**

4: $F \leftarrow F - 15$

5: Get RGB image

6: **if** AR marker is initially detected **then**

7: $p_0, q_0 \leftarrow$ Position, Posture of detected marker

8: $W \leftarrow C(z - z_{min}) + W_{min}$

9: $F \leftarrow -\alpha \ln z + \beta$

10: $D, I \leftarrow$ select ones from dataset of pre-defined data according to the focus value F

11: **Break**

12: **end if**

13: Get RGB image

14: **end while**

15: Go to Iteration Process

Algorithm 2 Iteration Process**Input:** RGB image**Output:** p_d, q_d *Loop Process:*

```

1: while  $|p_k - p_{k-1}| \geq t_p$  &  $|q_k - q_{k-1}| \geq t_q$  do
2:    $p_{k-1}, q_{k-1} \leftarrow p_k, q_k$ 
3:    $W \leftarrow C(z - z_{min}) + W_{min}$ 
4:    $F \leftarrow -\alpha \ln z + \beta$ 
5:    $D, I \leftarrow$  select ones from dataset of pre-defined data according to the focus value  $F$ 
6:   Get RGB image
7:   if AR marker is detected then
8:      $p_k, q_k \leftarrow$  Position, Posture of detected marker
9:     if  $|p_k - p_{k-1}| < t_p$  &  $|q_k - q_{k-1}| < t_q$  then
10:       $p_d, q_d \leftarrow p_k, q_k$ 
11:      Break
12:     end if
13:   end if
14: end while

```

3.2. Dynamic Camera Parameter Controller

Figure 7 shows a block diagram of the system for updating the camera parameter optimization shown in the *Iteration process* in Figure 6. The values p_d and q_d are the final output of the marker's position and posture. The Iterator judges whether the recognized position and posture converge to an accuracy within set thresholds, as in Equations (6) and (7).

$$|p_k - p_{k-1}| < t_p \quad (6)$$

$$|q_k - q_{k-1}| < t_q \quad (7)$$

The Iterator also calculates the marker's depth distance z for updating the camera parameters, which considers the magnification of the image due to zooming. When the image is magnified by a factor of n by zooming, the "apparent" depth distance z' becomes $1/n$ of the real value z . Therefore, the zoom value W is input to the LEAG Library to recognize the real marker position with compensation of the apparent depth distance z' , as shown in Equation (8).

$$z = z' \times n = z' \times \frac{W}{100} \quad (8)$$

After the depth distance z is calculated, the zoom value W and focus value F are updated by Equations (1) and (3), respectively. In addition, the lens distortion coefficients D and the internal parameter matrix I are appropriately selected from the calibration data list based on the focus value F .

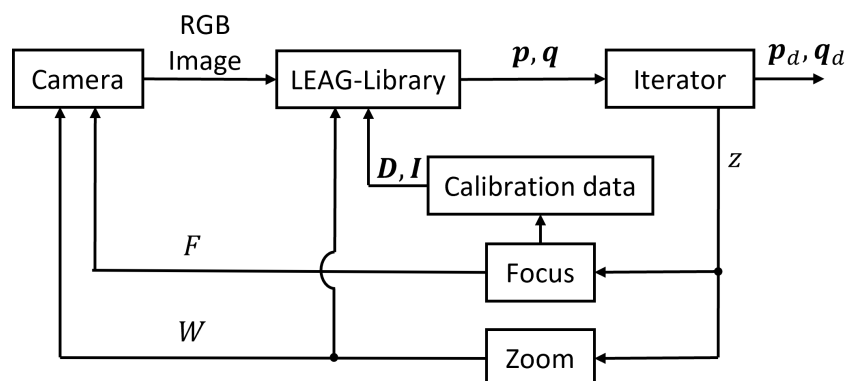


Figure 7. Block diagram of the camera parameter optimization.

4. Experimental Evaluation

4.1. Experimental Setup

Figure 8 shows the experimental setup. Figure 8a shows the definition of the coordinates of the AR marker: the depth distance z and the rotation angle ϕ , which are treated in the experiments in Sections 4.3–4.5. Figure 8b shows a scene of the actual experimental setup. The actual distances and angles were measured using a measuring tape and a protractor with the AR marker fixed to a rotating stand. To maintain consistent experimental conditions for each trial, we established an environment with a solid-colored wall as the marker background and placed the lighting in the same position. Furthermore, for camera parameters not dynamically controlled using our method, such as focus and zoom, we employed fixed parameters.

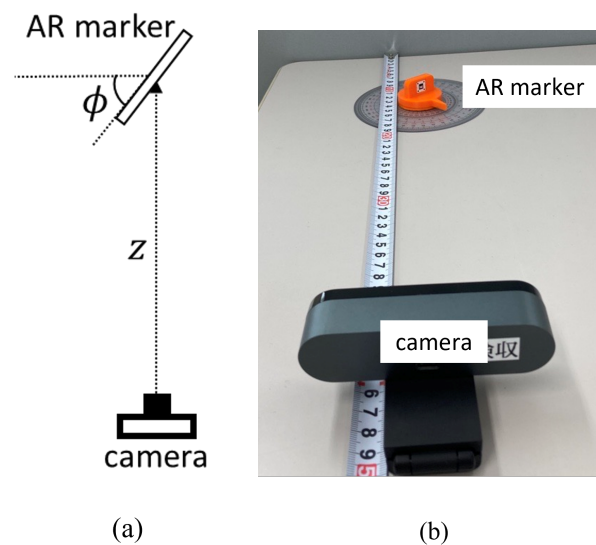


Figure 8. Experimental setup: (a) definition of the AR marker position and angles. (b) scene of the experiment.

4.2. Experimental Conditions

In all experiments, measurements were taken five times for each condition, and the average value was recorded for each condition. Note that the values were recorded only when the marker's ID was correctly recognized five times in a row.

The accuracy of the marker's translational position recognition is evaluated at the error rate relative to the true value. When the measured value is M and the true value is T , the error rate is defined by Equation (9). On the other hand, the accuracy of posture recognition is evaluated by the error between M and T .

$$\text{Error rate} = \frac{M - T}{T} \times 100 (\%) \quad (9)$$

4.3. Recognition Performance Using Fixed Camera Parameters

Prior to the evaluation of the proposed method, the recognition range and recognition accuracy were examined with camera parameters fixed at specific values. The zoom value W and focus value F were set to three patterns: (1) $W = 100, F = 100$; (2) $W = 500, F = 50$; (3) $W = 500, F = 0$. The distortion coefficient matrix D and internal parameter matrix I were calibrated with the camera's autofocus function turned on. The micro AR marker was placed at a rotation angle $\phi = 40^\circ$ to the camera for the most stable recognition.

Figure 9 shows the recognition range and error rate for the marker's depth distance z for each of the fixed parameter conditions. Under the conditions of this experiment, the recognition range was very narrow, with a maximum of only 0.2 m in the depth direction. Recognition accuracy also tends to significantly deteriorate, especially at short distances, which cannot be acceptable for precise navigation and object handling. This is largely

because a fixed focus value is used, which results in various out-of-focus areas. Therefore, it is necessary to expand the recognition range to properly control the camera focus and calibration data according to the marker's depth distance.

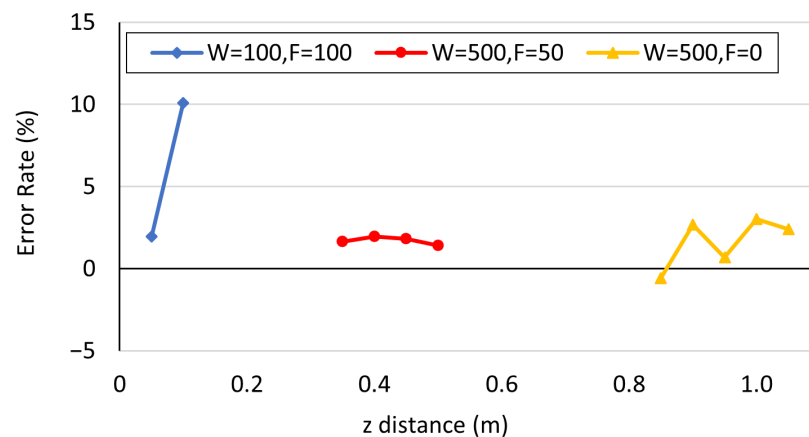


Figure 9. Recognition error rate of marker depth distance (z) with fixed camera parameters.

4.4. Recognition Performance Using Dynamic Focus Control

Next, we investigated the performance improvement in position recognition using the dynamic focus control given by Equation (3). In this experiment, the zoom values W were set to five constant values: 100, 200, 300, 400, and 500. The focus value F was sequentially computed according to the recognized values of depth distance z . The camera calibration data D and I are determined on the basis of the calculated focus values. During the experiment, the micro AR marker was placed at a rotation angle $\phi = 40^\circ$ to the camera for the most stable recognition.

Figure 10 shows the recognition range and error rate of the marker's depth distance z obtained in this experiment. The recognizable range of markers varies depending on the zoom value W . Naturally, the larger the zoom value, the farther away the marker can be recognized. At the maximum zoom value ($W = 500$), the AR markers can be recognized from the minimum distance $z_{min} = 0.05$ m to the maximum distance $z = 1.05$ m in the depth direction. Compared to Figure 9, the recognition range is expanded by adjusting the camera focus. However, when the zoom value W is large, the recognition error rate worsens at close distances. This is thought to be because the marker area in the image is considerably larger because of zooming at close distances, making it more susceptible to lens distortion. On the other hand, the recognition accuracy at close distances is good with small values of W . According to the experimental results, proper control of the zoom value W and the focus value F is necessary to ensure high recognition accuracy over the entire range.

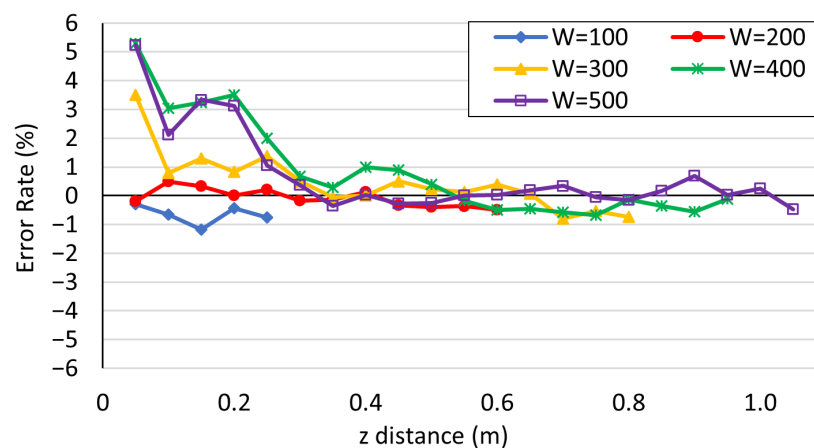


Figure 10. Recognition error rate of marker depth distance (z) using dynamic focus control with constant zoom values (W).

4.5. Recognition Performance Using Dynamic Control of Both Focus and Zoom (Proposed Method)

After understanding the results of the two experiments described above, we investigated and evaluated the recognition performance of the micro AR marker using dynamic control of both focus and zoom values, i.e., applying the proposed camera control method.

Table 3 shows the initial values of the camera parameters and the threshold values t_p and t_q used to determine the convergence of the iterative calculations. The initial zoom value was set to the maximum value ($W = 500$), which has the widest recognition range, as shown in Figure 10. The initial focus value was set at the closest distance from the camera ($F = 150$) so that the scanning process was performed from near to far points from the camera. The threshold values t_p and t_q for convergence judgment were set as absolute errors. In addition, the initial values of the camera calibration data D and I were measured with the autofocus function turned on, as described in Section 2.1.

Table 3. Initial camera parameter values and error thresholds.

Parameter	Symbol	Value
Zoom value	W	500
Focus value	F	150
Threshold of the position error	t_p	1.0 (mm)
Threshold of the posture error	t_q	0.01 (deg)

4.5.1. Performance of Translational Position Recognition

First, the precision and accuracy of the marker's translational position recognition are examined. The evaluation experiment was conducted by measuring the marker's depth distance z under three conditions of the marker's rotation angle: $\phi = 0^\circ, 30^\circ, 60^\circ$. To focus on the performance of the position measurement in this experiment, the convergence judgment of the marker's posture q was not performed; only the convergence judgment of the marker's position p was performed.

Figure 11 shows the recognition error rate of the marker's depth distance z . The error rate is within $\pm 1\%$ for the entire range in which AR markers can be recognized. Compared with the recognition accuracy with focus value control (see Figure 10), the proposed method achieved significant performance improvement. Furthermore, the variation in recognition error rates is not biased toward either positive or negative values, indicating that the accuracy of position estimation is high when using this method. The range of recognition distance becomes smaller at $\phi = 60^\circ$, but the recognition accuracy has not deteriorated as long as the marker position can be recognized.

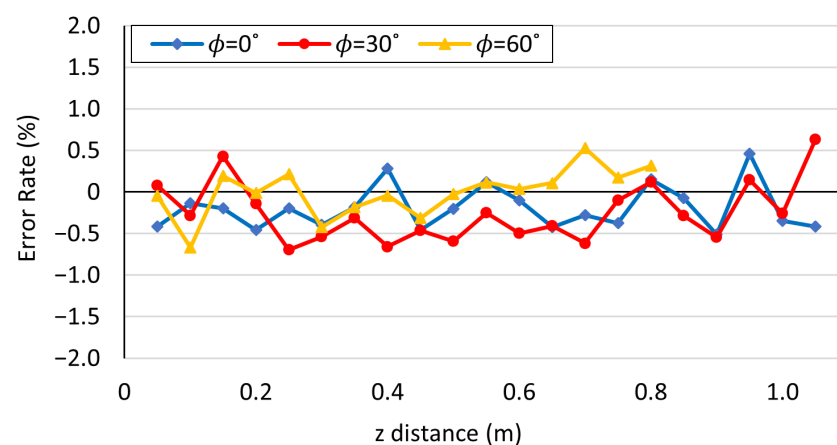


Figure 11. Recognition error rate of depth distance (z) with proposed camera control for different marker angles (ϕ).

The behavior of iterative calculations during marker position recognition was also investigated. Figure 12 shows the iteration time required for the convergence of position recognition in this experiment. According to the results, the iteration time is within 0.7 s, indicating that the AR marker at an unknown location can be recognized quickly with high accuracy. In addition, as the depth distance z increases, the iteration time tends to increase. Because the threshold value for the position convergence t_p is set to a constant value of 1.0 mm, independent of the depth distance, the convergence accuracy becomes relatively high at large distances.

In relation to the iteration time, Figure 13 shows the convergence of the recognized values of depth distance z by iterative calculations for five experimental trials. In this experiment, the true values of the marker's depth distance were set to (a) $z = 1.0$ m, (b) $z = 0.5$ m, and (c) $z = 0.1$ m with a rotation angle of $\phi = 40^\circ$. According to the results, although the number of convergence calculations tends to increase at large depth distances, recognition with the specified absolute accuracy (1.0 mm) is achieved over the entire range of recognizable depth distances. Note that the first detected values of z were larger than the true value in all cases. This is because the focus position is not perfectly adjusted to the marker immediately after scanning, and the AR marker is recognized as smaller than its actual size in the blurred image.

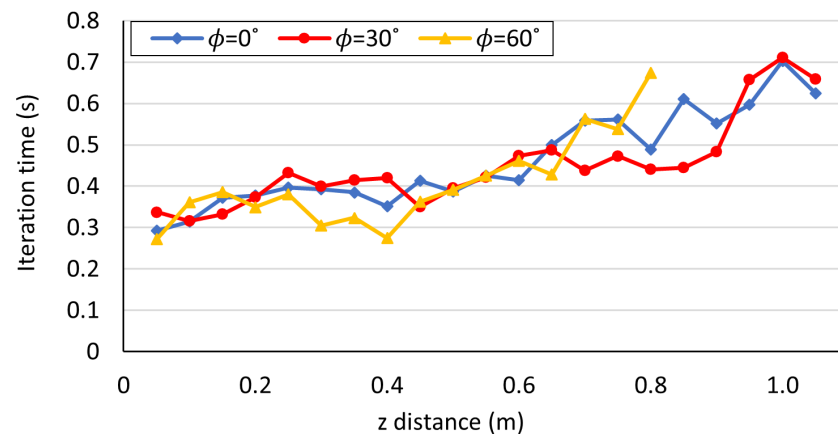


Figure 12. Iteration time of the depth distance (z) to converge with the proposed camera control for different marker angles (ϕ).

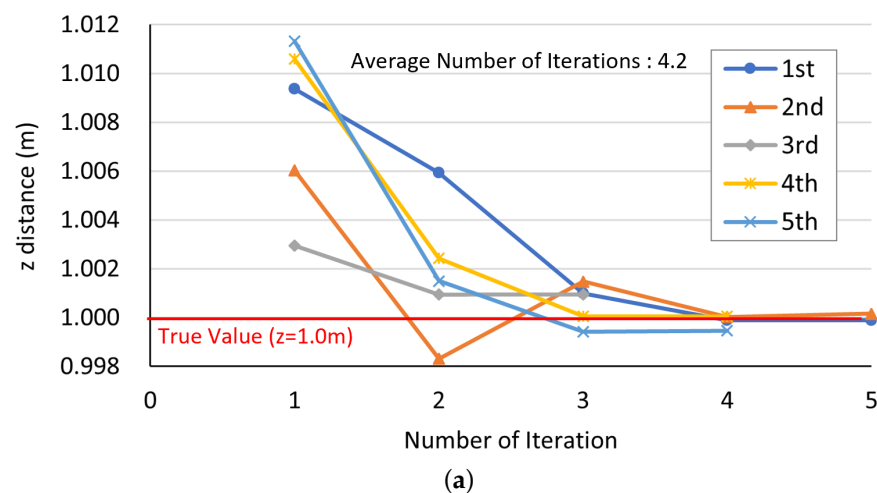


Figure 13. Cont.

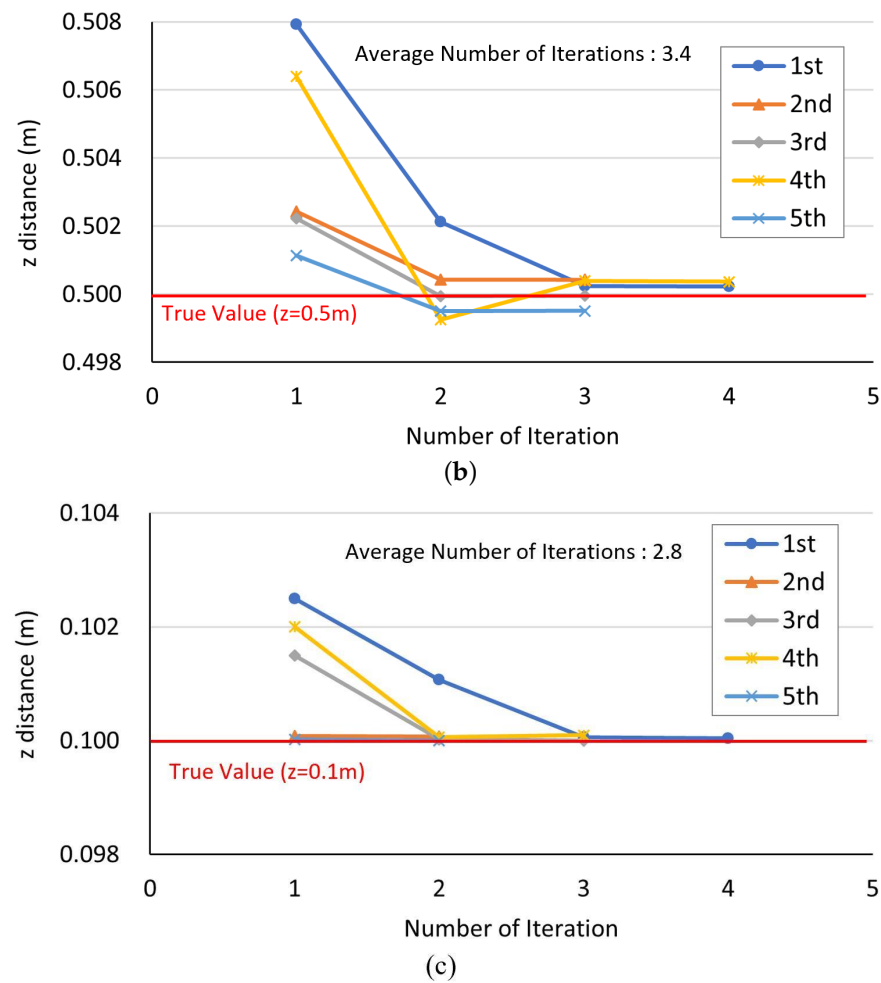


Figure 13. Convergence of the depth distance (z) measurement values to true values by iteration: (a) $z = 1.0 \text{ m}$; (b) $z = 0.5 \text{ m}$; (c) $z = 0.1 \text{ m}$.

4.5.2. Performance of Posture Recognition

Next, the precision and accuracy of the marker's posture recognition are examined. The evaluation experiment was conducted by measuring the marker's rotation angle ϕ under three depth distance conditions: $z = 0.05, 0.55, 1.05 \text{ m}$. To focus on the performance of the posture measurement in this experiment, the convergence judgment of the marker's position p was not performed; only the convergence judgment of the marker's posture q was performed.

As an experimental result, Figure 14 shows the recognition error of the marker's rotation angle ϕ at each condition. When the marker angle is in the range of 20° to 65° , accurate recognition within an error of $\pm 1^\circ$ is achieved. On the other hand, in the range where the marker angle is smaller than 20° , the recognition error is as large as 2.5° at maximum. This is due to the characteristic of AR markers that deteriorates recognition accuracy at angles near the frontal direction of the camera.

In addition, Figure 15 shows the iteration time required for the convergence of posture recognition in this experiment. According to the result, the iteration time is within 0.6 s , resulting in fast convergence of the recognized value of ϕ within the specified iteration accuracy ($t_q = 0.05^\circ$) over the entire recognizable angular range. It can also be seen that the larger the depth distance z , the longer it takes to converge the recognized value of the marker angle ϕ .

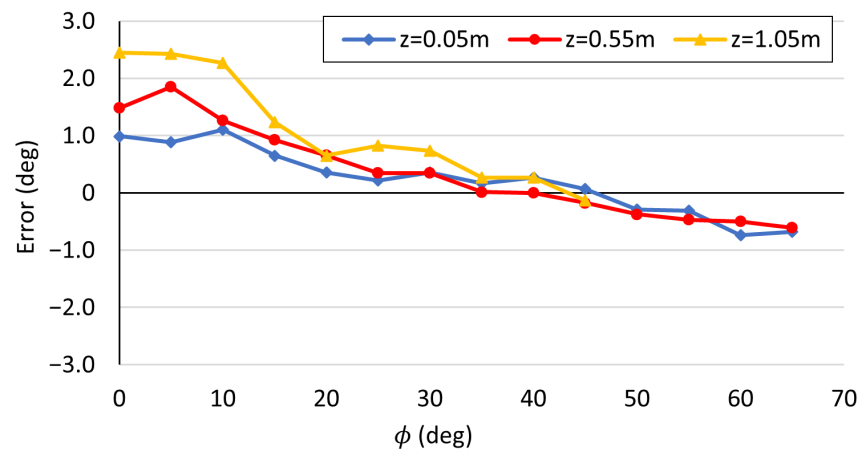


Figure 14. Recognition error of marker angle (ϕ) with the proposed camera control for different depth distances (z).

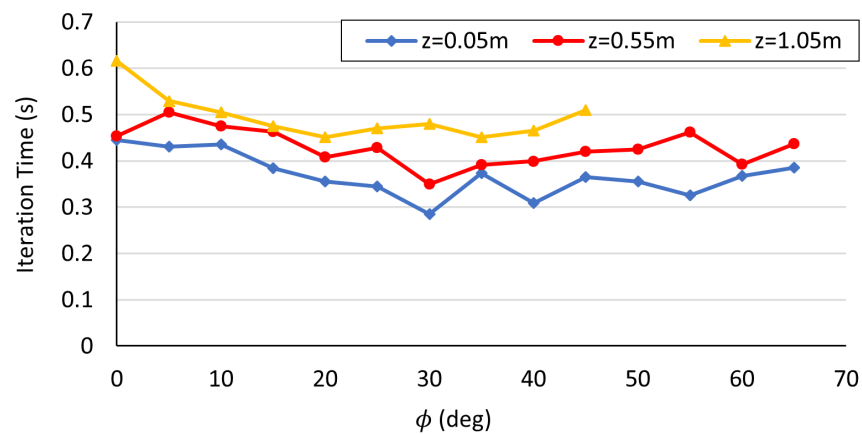


Figure 15. Iteration time of marker angle (ϕ) with the proposed camera control for different depth distances (z).

5. Discussion

Using the proposed dynamic camera control system, a 10 mm square micro AR marker can be recognized with an accuracy better than $\pm 1.0\%$ for depth distance and 2.5° for rotation angle. The depth recognition range is 1.0 m, which is five times greater than the range with fixed camera parameters. In the most recent relevant study, Inoue et al. [41] proposed an AR marker pose estimation system using a $64\text{ mm} \times 64\text{ mm}$ AR marker, in which the recognition accuracy was 4% in the depth direction and 3.9° in the rotation angle. The depth recognition range was 0.1~0.7 m. Compared with the state-of-the-art research described above, the proposed method in this study achieves significantly better performance in terms of marker size, recognition accuracy, and recognition range.

Regarding the iteration time, the proposed system requires a maximum of 0.6–0.7 s for convergence of recognition values for both position and posture. For application to object tracking in a robotic manipulation system, the iteration time can be significantly reduced, except for the initial detection. This is because the previous marker recognition value and the kinematic information of the robot arm can be used to efficiently determine the initial values of the camera parameters at the next recognition.

However, the recognition performance of the proposed system depends on the hardware specifications. A camera with higher zoom magnification capabilities, including an optical zoom, can further expand the recognition range. Iteration time also depends on the computational resources. Considering the low-cost implementation, this study uses a normal laptop PC without a GPU. However, the real-time performance of the sys-

tem can be further improved using an industrial camera with a high frame rate and a GPU-accelerated processor.

6. Conclusions

In this paper, a dynamic camera parameter control method was proposed for the accurate and precise recognition of micro AR markers. In this system, the detected position and posture of the marker are converged to a specified accuracy through iterative calculations by updating the camera parameters. Evaluation experiments have shown the significant superiority of the proposed system in recognition performance over existing methods.

In future work, we aim to utilize the proposed recognition method for position estimation of small or transparent objects using micro AR markers. Subsequently, we plan to apply this technology in various domains, such as constructing a laboratory automation framework and automating tasks related to object organization and tidying. Ultimately, we intend to implement a robotic arm-based object handling system.

Author Contributions: Conceptualization, D.H.; methodology, D.H. and R.M.; software, R.M.; validation, D.H. and R.M.; formal analysis, D.H. and R.M.; investigation, R.M.; resources, D.H.; writing—original draft preparation, D.H. and R.M.; writing—review and editing, D.H.; visualization, R.M.; supervision, D.H.; project administration, D.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments: The authors would like to thank LEAG Solutions Corp. and Dai Nippon Printing Co., Ltd. for their cooperation in prototyping of the micro AR Marker.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Brachmann, E.; Rother, C. Visual camera re-localization from RGB and RGB-D images using DSAC. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 5847–5865. [CrossRef] [PubMed]
- Mathis, A.; Mamidanna, P.; Cury, K.M.; Abe, T.; Murthy, V.N.; Mathis, M.W.; Bethge, M. DeepLabCut: Markerless pose estimation of user-defined body parts with deep learning. *Nat. Neurosci.* **2018**, *21*, 1281–1289. [CrossRef] [PubMed]
- Madec, S.; Jin, X.; Lu, H.; Solan, B.D.; Liu, S.; Duyme, F.; Heritier, E.; Baret, F. Ear density estimation from high resolution RGB imagery using deep learning technique. *Agric. For. Meteorol.* **2019**, *264*, 225–234. [CrossRef]
- Panteleris, P.; Oikonomidis, I.; Argyros, A. Using a Single RGB Frame for Real Time 3D Hand Pose Estimation in the Wild. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018; pp. 436–445. [CrossRef]
- Ichnowski, J.; Avigal, Y.; Kerr, J.; Goldberg, K. Dex-NeRF: Using a Neural Radiance Field to Grasp Transparent Objects. In Proceedings of the Machine Learning Research, Baltimore, MD, USA, 17–23 July 2022; Volume 164, pp. 526–536.
- Kan, T.W.; Teng, C.H.; Chen, M.Y. QR Code Based Augmented Reality Applications. In *Handbook of Augmented Reality*; Furht, B., Ed.; Springer: New York, NY, USA, 2011; pp. 339–354. [CrossRef]
- Kalaitzakis, M.; Carroll, S.; Ambrosi, A.; Whitehead, C.; Vitzilaios, N. Experimental Comparison of Fiducial Markers for Pose Estimation. In Proceedings of the 2020 International Conference on Unmanned Aircraft Systems (ICUAS), Athens, Greece, 1–4 September 2020; pp. 781–789. [CrossRef]
- Ruan, K.; Jeong, H. An Augmented Reality System Using Qr Code as Marker in Android Smartphone. In Proceedings of the 2012 Spring Congress on Engineering and Technology, Xi'an, China, 27–30 May 2012; pp. 1–3. [CrossRef]
- Ikeda, K.; Tsukada, K. CapacitiveMarker: Novel interaction method using visual marker integrated with conductive pattern. In Proceedings of the 6th Augmented Human International Conference, Singapore, 9–11 March 2015; pp. 225–226.
- Uranishi, Y.; Imura, M.; Kuroda, T. *The Rainbow Marker: An AR Marker with Planar Light Probe Based on Structural Color Pattern Matching*; IEEE: Manhattan, NY, USA, 2016; pp. 303–304.
- ARToolKit SDKs Download Website. Available online: <http://www.hitl.washington.edu/artoolkit/download/> (accessed on 23 February 2023).
- Zhao, T.; Jiang, H. *Landing System for AR. Drone 2.0 Using Onboard Camera and ROS*; IEEE: Manhattan, NY, USA, 2016; pp. 1098–1102.

13. Qi, J.; Guan, X.; Lu, X. An Autonomous Pose Estimation Method of MAV Based on Monocular Camera and Visual Markers. In Proceedings of the 2018 13th World Congress on Intelligent Control and Automation (WCICA), Changsha, China, 4–8 July 2018; pp. 252–257. [\[CrossRef\]](#)
14. Aoki, R.; Tanaka, H.; Izumi, K.; Tsujimura, T. Self-Position Estimation based on Road Sign using Augmented Reality Technology. In Proceedings of the 2018 12th France-Japan and 10th Europe-Asia Congress on Mechatronics, Tsu, Japan, 10–12 September 2018; pp. 39–42. [\[CrossRef\]](#)
15. Ababsa, F.-e.; Mallem, M. Robust camera pose estimation using 2d fiducials tracking for real-time augmented reality systems. In Proceedings of the 2004 ACM SIGGRAPH International Conference on Virtual Reality Continuum and Its Applications in Industry, Singapore, 16–18 June 2004; pp. 431–435.
16. Kato, J.; Deguchi, G.; Inoue, J.; Iwase, M. Improvement of Performance of Navigation System for Supporting Independence Rehabilitation of Wheelchair—Bed Transfer. *J. Phys. Conf. Ser.* **2020**, *1487*, 012041. [\[CrossRef\]](#)
17. Nakanishi, H.; Hashimoto, H. AR-Marker/IMU Hybrid Navigation System for Tether-Powered UAV. *J. Robot. Mechatron.* **2018**, *30*, 76–85. [\[CrossRef\]](#)
18. Tsujimura, T.; Aoki, R.; Izumi, K. *Geometrical Optics Analysis of Projected-Marker Augmented Reality System for Robot Navigation*; IEEE: Manhattan, NY, USA, 2018; pp. 1–6.
19. Yu, X.; Yang, G.; Jones, S.; Saniie, J. AR Marker Aided Obstacle Localization System for Assisting Visually Impaired. In Proceedings of the 2018 IEEE International Conference on Electro/Information Technology (EIT), Rochester, MI, USA, 3–5 May 2018; pp. 271–276. [\[CrossRef\]](#)
20. Romli, R.; Razali, A.F.; Ghazali, N.H.; Hanin, N.A.; Ibrahim, S.Z. Mobile Augmented Reality (AR) Marker-based for Indoor Library Navigation. *IOP Conf. Ser. Mater. Sci. Eng.* **2020**, *767*, 012062. [\[CrossRef\]](#)
21. Choi, C.; Christensen, H.I. Real-time 3D model-based tracking using edge and keypoint features for robotic manipulation. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation, Anchorage, AK, USA, 3–8 May 2010; pp. 4048–4055. [\[CrossRef\]](#)
22. Pai, Y.S.; Yap, H.J.; Singh, R. Augmented reality-based programming, planning and simulation of a robotic work cell. *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.* **2014**, *229*, 1029–1045. [\[CrossRef\]](#)
23. Raessa, M.; Chen, J.C.Y.; Wan, W.; Harada, K. Human-in-the-Loop Robotic Manipulation Planning for Collaborative Assembly. *IEEE Trans. Autom. Sci. Eng.* **2020**, *17*, 1800–1813. [\[CrossRef\]](#)
24. Zhang, L.; Ye, M.; Chan, P.L.; Yang, G.Z. Real-time surgical tool tracking and pose estimation using a hybrid cylindrical marker. *Int. J. Comput. Assist. Radiol. Surg.* **2017**, *12*, 921–930. [\[CrossRef\]](#) [\[PubMed\]](#)
25. Costanza, E.; Huang, J. *Designable Visual Markers*; Association for Computing Machinery: New York, NY, USA, 2009; pp. 1879–1888. [\[CrossRef\]](#)
26. Douchamps, D.; Chihara, K. High-accuracy and robust localization of large control markers for geometric camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *31*, 376–383. [\[CrossRef\]](#) [\[PubMed\]](#)
27. Yoon, J.H.; Park, J.S.; Kim, C. *Increasing Camera Pose Estimation Accuracy Using Multiple Markers*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 239–248.
28. Yu, R.; Yang, T.; Zheng, J.; Zhang, X. Real-Time Camera Pose Estimation Based on Multiple Planar Markers. In Proceedings of the 2009 Fifth International Conference on Image and Graphics, Xi'an, China, 20–23 September 2009; pp. 640–645. [\[CrossRef\]](#)
29. Hayakawa, S.; Al-Falouji, G.; Schickhuber, G.; Mandl, R.; Yoshida, T.; Hangai, S. *A Method of Toothbrush Position Measurement Using AR Markers*; IEEE: Manhattan, NY, USA, 2020; pp. 91–93.
30. Uematsu, Y.; Saito, H. Improvement of Accuracy for 2D Marker-Based Tracking Using Particle Filter. In Proceedings of the 17th International Conference on Artificial Reality and Telexistence (ICAT 2007), Esbjerg, Jylland, Denmark, 28–30 November 2007; pp. 183–189. [\[CrossRef\]](#)
31. Rubio, M.; Quintana, A.; Pérez-Rosés, H.; Quirós, R.; Camahort, E. *Jittering Reduction in Marker-Based Augmented Reality Systems*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 510–517.
32. Bergamasco, F.; Albarelli, A.; Rodolà, E.; Torsello, A. RENE-Tag: A high accuracy fiducial marker with strong occlusion resilience. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; pp. 113–120. [\[CrossRef\]](#)
33. Bergamasco, F.; Albarelli, A.; Torsello, A. Pi-tag: A fast image-space marker design based on projective invariants. *Mach. Vis. Appl.* **2013**, *24*, 1295–1310. [\[CrossRef\]](#)
34. Tanaka, H.; Sumi, Y.; Matsumoto, Y. *A Novel AR Marker for High-Accuracy Stable Image Overlay*; IEEE: Manhattan, NY, USA, 2012; pp. 217–218.
35. Tanaka, H.; Sumi, Y.; Matsumoto, Y. *A High-Accuracy Visual Marker Based on a Microlens Array*; IEEE: Manhattan, NY, USA, 2012; pp. 4192–4197.
36. Tanaka, H.; Sumi, Y.; Matsumoto, Y. Avisual marker for precise pose estimation based on lenticular lenses. In Proceedings of the 2012 IEEE International Conference on Robotics and Automation, St. Paul, MN, USA, 14–18 May 2012; pp. 5222–5227. [\[CrossRef\]](#)
37. Tanaka, H.; Ogata, K.; Matsumoto, Y. Improving the accuracy of visual markers by four dots and image interpolation. In Proceedings of the 2016 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS), Tokyo, Japan, 17–20 December 2016; pp. 178–183. [\[CrossRef\]](#)
38. Toyoura, M.; Aruga, H.; Turk, M.; Mao, X. Mono-spectrum marker: An AR marker robust to image blur and defocus. *Vis. Comput.* **2014**, *30*, 1035–1044. [\[CrossRef\]](#)

39. Miyahara, R.; Haraguchi, D. Object Handling System using Ultra-Small and High-Precision AR Markers(KRIS2023). 2023, p. 140. Available online: <https://kris2023.kosen-k.go.jp/ja/> (accessed on 5 July 2023).
40. OpenCV-Python Tutorials. Available online: http://labs.eecs.tottori-u.ac.jp/sd/Member/oyamada/OpenCV/html/py_tutorials/py_calib3d/py_calibration/py_calibration.html# (accessed on 23 February 2023).
41. Inoue, M.; Ogata, M.; Izumi, K.; Tsujimura, T. Posture Estimation for Robot Navigation System based on AR Markers. In Proceedings of the 2021 60th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), Tokyo, Japan, 8–10 September 2021; pp. 628–633.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.