The effect of ArUco marker size, number, and distribution on the localization performance of fixed-point targets

1st Renpei Wang
Harbin Institute of Technology
Weihai, China
2201020122@stu.hit.edu.cn

2nd Hongtu Guo
Harbin Institute of Technology
Weihai, China
2200200813@stu.hit.edu.cn

3rd Xin Wang*
National Ocean Technology Center
Tianjin, China
wxtjnk022@126.com

4th Liang Han
Harbin Institute of Technology
Weihai, China
hanliang@hit.edu.cn

2023 6th International Conference on Robotics, Control and Automation Engineering (RCAE) | 979-8-3503-1530-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/RCAE59706.2023.10398770

***Abstract*—ArUco markers have a lower false alarm rate and higher robustness in pose estimation than other AR libraries and have a wide range of applications in target tracking and UAV positioning. In this paper, we propose an evaluation system for evaluating the accuracy, precision, and processing time of the localization system to address the lack of design guidance for ArUco marker patterns in practice and rely on a homemade test system to give the influence curves between factors such as the number, size, and distribution of ArUco markers and the localization performance based on several comparative experiments. The results show that the size of the marker is the most influential factor in the accuracy and precision of the system. Increasing the marker size from 300 pixels to 1400 pixels improves the accuracy and precision by 134.19% and 738.89%, respectively, while the size does not affect the processing time. Increasing the number of markers from 1 to 9 improves the accuracy and precision by 52.96% and 563.59%, respectively, but increases the system processing time. Compared to the other factors analyzed, the distribution of markers does not significantly impact the positioning system's performance. The conclusions of this paper are instructive for the subsequent autonomous landing of related UAS platforms.**

***Index Terms*—ArUco positioning, Marker pattern, Size, Number, UAV landing**

I. Introduction

In recent years, automation devices have been increasingly used in military and civilian projects in different fields, such as unmanned aerial vehicles [1], underwater surveys [2], robotics [3], surgical navigation [4], etc. For these automated devices, an accurate positional estimation system is indispensable. In the case of UAVs, for example, when they perform tasks such as cargo delivery [5], environmental detection and control [6], and structural inspection and monitoring [7], they need to take off at the beginning of the operation and land precisely after the mission is completed. Therefore, an accurate positional estimation system is needed to solve the above problems. Position estimation based on visual sensory data is often used nowadays and it has many successful applications, such as localization [8], navigation [9], ranging [10], etc.

Position estimation based on visual sensory data is done by finding the correspondence between feature points in the real environment and their projection on a 2D image (e.g., an optical camera) to obtain information about the position of the target or camera. Since the computational cost becomes prohibitive when using arbitrary images as features, synthetic markers are usually used to extract the appropriate feature points [11]. Regular and contrastingly colored squares are one of the easiest markers to think of, and most square markers are derived from ARToolkit [12], which uses a global threshold to identify regions that can be fitted as quadrilateral contours. But it has a higher probability of detection failure and false detection. To solve this problem, ARTag [13] was created to improve the reliability of marker detection and to achieve detection in partial occlusion cases. And ArUco [14] used in this paper is developed based on the above two libraries ARToolkit and ARTag. It is similar to ARTag, but it allows users to choose their own generation libraries according to their needs, thus reducing the computation time. Since ArUco markers possess a low false alarm rate and high robustness with a short detection time, they are widely used in target tracking [15], [16], UAV localization [17]–[19], and others [20].

However, many studies have only used ArUco markers as tools in practice, paying little attention to the design of ArUco marker patterns and lacking in-depth studies of related factors. Reference [15] implemented target tracking using 5 ArUco markers, but the effect of the number and size of markers on bit pose detection is not given. Reference [17], [18] and [19] introduced multiple combinatorial and embedded distribution patterns of ArUco codes, respectively, but neither mentioned the rationale for selecting the ArUco tag distribution nor compared the advantages and disadvantages of the two different schemes. Reference [20] compared the effect of different ArUco marker sizes on the positioning effect in their work, but only 2 sets of comparisons were made, and the rule could not be completely represented. To this end, this paper will analyze the impact of the ArUco marker on localization performance based on OpenCV ArUco implementation theory for the number, size, and distribution of markers, and build a low-cost test system for further comparison experiments to verify the impact of different factors. Its conclusions are instructive for ArUco applications in related fields.

II. ArUco Positioning Principles

A. ArUco marker detection and identification

As shown in Fig. 1, the ArUco marker provided by OpenCV ArUco library is a square 2D code consisting of an outer black border and an internal binary matrix code, where the outer border is used to locate marker corner points and the binary code is used for ID identification, message verification, and error correction.

As shown in Fig. 2, the steps for the detection and identification of a single ArUco marker are as follows.

a) Thresholding: Converts incoming color images from the camera to grayscale images and adaptively thresholds the grayscale images.

b) Contour filtering: Extract all edge contours, detect and filter the quad contours that are closest to the marker, and eliminate the quad contours that are closer together for processing.

c) Acquire marker encoding: Perform perspective transformation on the region image after contour filtering and separate black and white pixels using Otsu thresholding.

d) Determine the ID and corner point order: Encode the obtained image for recognition, obtain the ID of the marker, and determine the order of corner point arrangement.

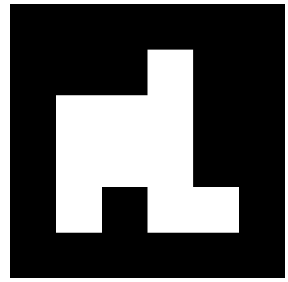


Fig. 1. ArUco marker.

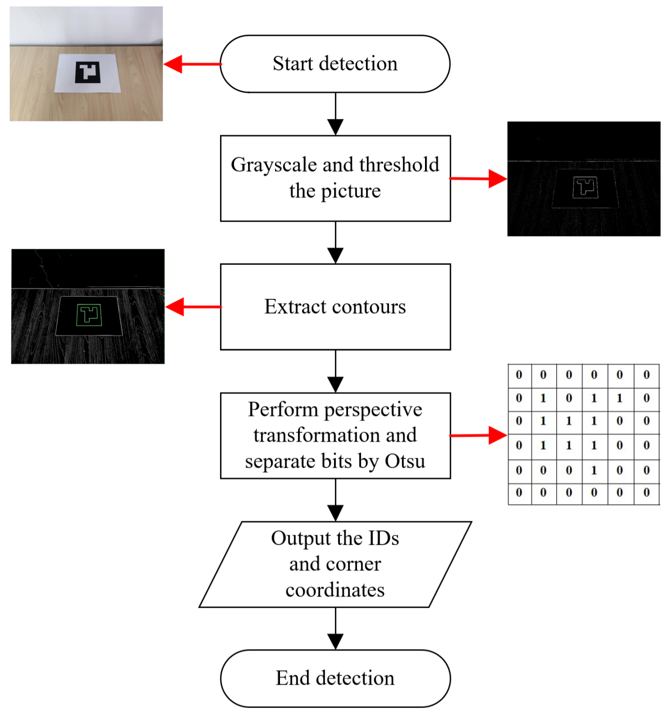


Fig. 2. Detection and identification process of ArUco markers.

B. ArUco Board localization based on L-M algorithm

ArUco Board is another form of pattern offered by ArUco and consists of several ArUco markers. In this paper, we use the solvePnP function in the OpenCV ArUco library to implement the bit pose estimation of the ArUco Board. The required parameters are the world coordinates of the four corner points of each marker in ArUco Board and the pixel coordinates of its projection into the image, the internal parameters matrix, and the distortion vector of the camera. The definition of world and pixel coordinates and the relationship between them are given in Fig. 3, and their mathematical expression is given by (1).

$$S_i \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \begin{bmatrix} K & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \tag{1}$$

where S_i is the depth value of the corner point P_i. X_i, Y_i, and Z_i are the world coordinates of P_i. u_i and v_i are the pixel coordinates of the projection point p_i. K is the internal parameters matrix of the camera. R, T are the rotation matrix and translation vector of the camera also known as the external parameters of the camera.

solvePnP function returns the external camera parameters R, T. Since the external camera parameters describe the transformation relationship between the world coordinate system and the camera coordinate system, a simple algebraic transformation of the external camera parameters can obtain the coordinates of the camera optical center O_c in the world coordinate system.

In the process of calculating the camera's external parameters, the calculation results can be optimized by minimizing the reprojection error. Minimizing reprojection error is the process of minimizing the difference between the projected pixel coordinates and the actual detected pixel coordinates by iterative optimization after the world coordinates of corner points are projected onto the image plane through the camera's internal and external parameters matrices to obtain the pixel

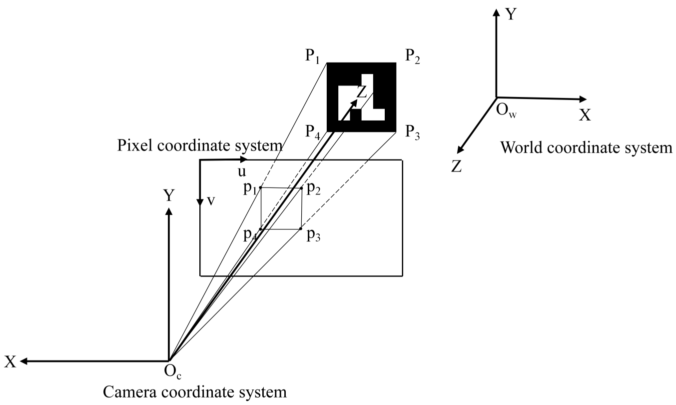


Fig. 3. Perspect metric coordinate system ive projection model

coordinates.

Considering n three-dimensional spatial points P_i and their projection p_i in the image, the mathematical representation of the reprojection error can be written from (1), given by

$$\sum_{i=1}^{n} \left\| P_i - \frac{1}{s_i} \begin{bmatrix} K & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} P_i \right\|^2 \tag{2}$$

where $p_i = [u_i, v_i, 1]^{-1}$ and $Pi = [X_i, Y_i, Z_i, 1]^{-1}$. It is important to note that the coordinate values in p_i are obtained by the image detection and recognition step.

In the process of minimizing the reprojection error, an iterative optimization method is usually used, for example, using the Levenberg-Marquardt algorithm [21]. The specific steps are as follows.

a) Initialize the external camera parameters: The initialization of the external parameters can be done by some heuristics, for example, by using the PnP algorithm to estimate.

b) Calculating reprojection error: The world coordinates of the marked corner points are projected onto the image plane through the internal camera parameters matrix and the current external parameters to obtain the reprojected pixel coordinates, and the difference between them and the detected pixel coordinates is calculated as the loss function of the current optimization target.

c) Update the external camera parameters: Adjust the rotation matrix and translation vector of the camera according to the loss function to minimize the reprojection error.

d) Repeat step b and c: Keep iteratively updating the external camera parameters and calculating the reprojection error until the error reaches a certain threshold or the optimization times reach a preset maximum.

III. Analysis of the Effect of Marking Patterns on Positioning Performance

The following section is a discussion of how the ArUco marker pattern affects the localization performance based on the theory in Section I. The discussion is on factors including the number, size, and distribution of markers in the marker pattern. The discussion of the performance of the localization system in this paper focuses on the accuracy, precision, and processing time of localization.

A. Effect of marker numbers on positioning performance

Due to the iterative approach of the L-M algorithm and the detection and recognition theory of ArUco markers, the number of markers on the Board has an impact on the accuracy and precision of the bit pose estimation and the processing time. As the number of markers increases, the Board can provide more detection constraints and information, which improves the accuracy and precision, but also increases the computation and detection time. The detailed discussion is as follows.

As the number of markers increases, the corresponding number of corner points also increases, which leads to a larger condition number of the Hessian matrix (the second-order derivative matrix of the loss function in the L-M algorithm), thus bringing the results of the bit pose estimation closer to the actual value.

When performing ArUco Board detection, each marker needs to be detected and identified, so as the number of markers increases, the processing time increases accordingly. When parsing a single marker, the processing time of the positioning system is typically between 1-2ms, while when parsing an entire ArUco Board, due to the need to detect and match multiple markers, the processing time is longer, possibly between tens and hundreds of milliseconds.

B. Effect of marker size on positioning performance

When the camera focal length is determined and the camera-marker distance is fixed, the larger the physical size of the marker, the larger the contour in the pixel coordinate system, and the positive correlation between them can be deduced from (1):

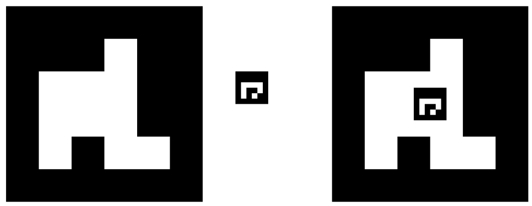
$$n = \frac{1}{s} \times \frac{f}{dx} \times L \tag{3}$$

Where n is the number of pixels contained in the contour edge length in the pixel coordinate system. s is the depth value of the corner point. $\frac{f}{dx}$ is an item in the external camera parameter, and L is the edge length of the marker in the world coordinate system. It should be noted that the derivation of (3) is based on the premise that the origin of the world coordinate system falls on the Z-axis of the camera coordinate system and the two X-axis coincide.

Therefore, n is linearly related to L. Let $K = \frac{1}{s} \times \frac{f}{dx}$. Supposing that the process of detecting contours and extracting corner points is affected by noise so that the pixel coordinates of corner points have an offset of Δn, the resulting error $e = \frac{\Delta n}{n} = \frac{\Delta n}{KS}$. Therefore, the larger the size of the marker being detected, the smaller the error of positioning and the higher the accuracy of positioning.

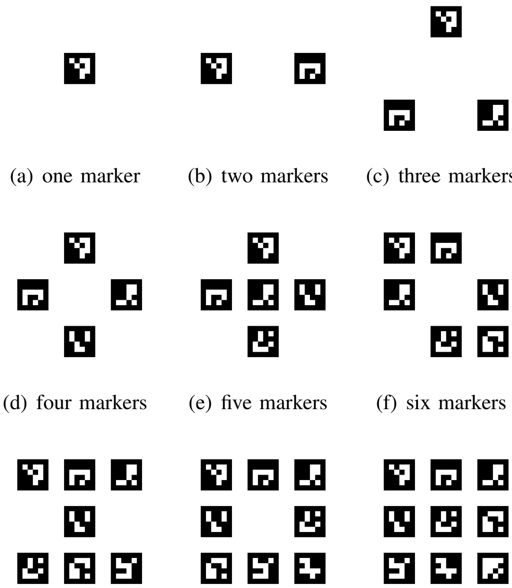
C. Effect of marker distribution on positioning performance

As shown in Fig. 4, combined distribution and embedded distribution are two common distributions of multiple markers, and both have advantages over a single marker:



(a) Combined ArUco Board (b) Embedded ArUco Board

Fig. 4. Two common distributions of ArUco Board.

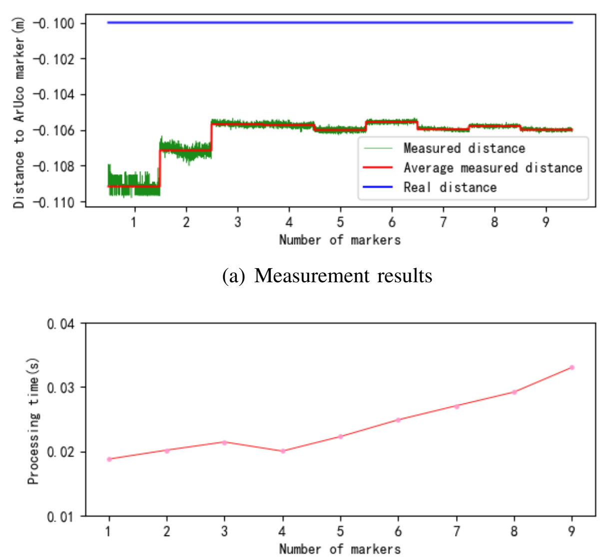


(a) one marker (b) two markers (c) three markers

(d) four markers (e) five markers (f) six markers

(g) seven markers (h) eight markers (i) nine markers

Fig. 5. Marker pattern used to investigate the effect of marker number.



(a) Measurement results

(b) Processing time

Fig. 6. Measurement results and Processing time in dependence of the number of markers.

- more accurate estimation of the bit position.
- capable of solving the problem of partial obscuration.
- capable of solving the problem of too small recognition interval.

For advantage 2 and 3, the differences produced by different distributions are difficult to describe quantitatively, and this paper gives the following qualitative analysis with experience in applications: the combined distribution is more advantageous in coping with partial obscuration of markers because the total area of the markers and the spacing between markers is usually larger compared to the embedded distribution. The embedded distribution is more suitable for applications where the camera height varies vertically, such as autonomous drone landing, because the centers of markers in different sizes can overlap, which can continuously guide the drone toward the center of the marker, and because its area is smaller, it is also more advantageous than the combined distribution in some applications where the marker area is fixed.

For advantage 1, the differences between different distributions can be compared quantitatively, so the exploration of the influence of distribution factors in this paper focuses on advantage 1. Based on the theories in Section 1.1 and 1.2, we argue that adjusting the distribution of ArUco does not affect the accuracy, precision, and processing time of localization with a fixed number and size of ArUco markers. The results of the quantitative comparison experiments are shown in the next section.

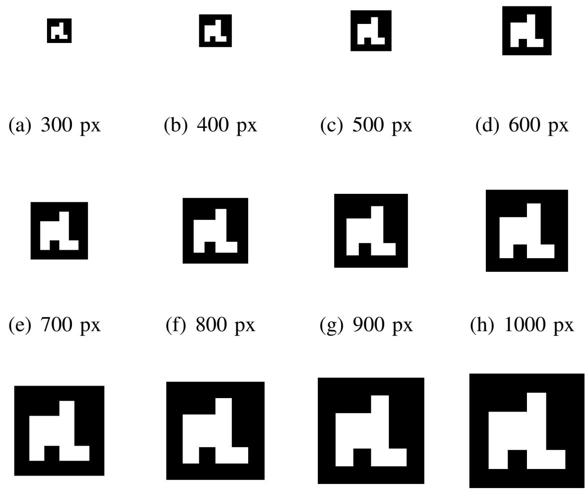
IV. Experiments and Results

A. Investigating the effect of marker number on positioning performance

To investigate the effect of Tag number on localization performance, quantitative experiments were conducted using nine tags containing different Tag numbers, as shown in Fig. 5.

The results of the quantitative experiments are shown in Fig. 6, as the number of detected markers increases from 1 to 9, the relative measurement error decreases from 9.170% to 5.995% and stabilizes around the latter. The standard deviation of 1000 measurements against each ArUco Board decreases from 4.148×10^{-4} to 6.252×10^{-5}. This indicates that the accuracy and precision of localization increase with the number of detected markers increases. However, it is worth noting that the increase in localization accuracy and precision from increasing the marker number is greatly reduced when the number is greater than 3.

Fig. 6(b) shows that the number of markers has an obvious positive correlation with the processing time. The processing time of a single image frame increases from 0.019 s to 0.033 s as the number of markers rises, the increase in processing time is smaller when the number is around 4. Considering that the increase of localization accuracy and precision is not obvious after the number is greater than 3, we recommend that the

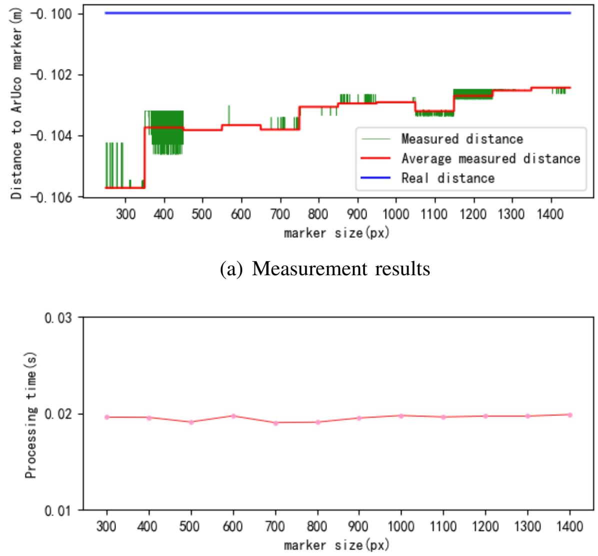


(a) 300 px (b) 400 px (c) 500 px (d) 600 px

(e) 700 px (f) 800 px (g) 900 px (h) 1000 px

(i) 1100 px (j) 1200 px (k) 1300 px (l) 1400 px

Fig. 7. Marker pattern used to investigate the effect of marker size.



(a) Measurement results

(b) Processing time

Fig. 8. Measurement results and Processing time in dependence of the size of markers.

number of markers in ArUco marker pattern should be 3-5.

B. Investigating the effect of marker size on positioning performance

To investigate the effect of marker size on positioning performance, size experiments were conducted using 12 markers with different sizes. The pixel sizes of the 12 markers were evenly distributed between 300 pixels (equal to approximately 2.54cm) and 1400 pixels (equal to approximately 11.85cm), as shown in Fig. 7.

The results of the size experiment are shown in Fig. 8. The relative measurement error decreases from 5.721% to 2.443% as the marker size increases. The extreme difference of 1000 measurements decreases from 0.00151 cm to 0.00020 cm. This indicates that the increase in marker size has an enhancement effect on both the accuracy and precision of positioning. Fig. 8(b) shows that the change in marker size did not have a significant effect on the processing time.

C. Investigating the effect of marker distribution on positioning performance

To investigate the effect of Marker distribution on the positioning performance, two ArUco Boards with different distributions have been designed and shown in Fig. 4, which are the combined Board and the embedded Board.

The results of the distribution experiments are shown in Table I. There is no significant difference between the combined and embedded distributions in the evaluation metrics of relative error, standard deviation, extreme difference, and processing time per frame.

TABLE I
COMPARISON OF THE POSITIONING PERFORMANCE OF TWO DIFFERENT DISTRIBUTIONS

Distribution	Relative error	Standard deviation	Extreme difference (m)	Processing time (s)
Combined Board	3.128%	1.130×10^{-4}	4.2×10^{-4}	0.0203
Embedded Board	2.895%	1.181×10^{-4}	5.8×10^{-4}	0.0201

V. CONCLUSION

Based on theoretical analysis and comparative experiments, this paper investigates the effects of the number, size, and distribution of ArUco markers on positioning performance. Research shows that the number and size of markers are factors that can improve positioning accuracy and precision and that the processing time of the system is only positively related to the number of codes. The above conclusions can be used to guide the pattern design of ArUco markers, and are of good value for application practices in the fields of high-precision target tracking, autonomous UAV landing, and indoor visual localization. The next direction of the research is to conduct outdoor experiments, to verify the correctness and practicality of the conclusions of this paper by testing the success rate, accuracy, and time of landing the UAV onto the ArUco marker.

REFERENCES

[1] Derek Kingston, Steven Rasmussen, and Laura Humphrey. Automated uav tasks for search and surveillance. In *2016 IEEE Conference on Control Applications (CCA)*, pages 1–8. IEEE, 2016.

[2] Andrew Speers, Anna Topol, James Zacher, Robert Codd-Downey, Bart Verzijlenberg, and Michael Jenkin. Monitoring underwater sensors with an amphibious robot. In *2011 Canadian conference on computer and robot vision*, pages 153–159. IEEE, 2011.

[3] Riby Abraham Boby and Subir Kumar Saha. Single image based camera calibration and pose estimation of the end-effector of a robot. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2435–2440. IEEE, 2016.

[4] Wanru Zhou and Baolong Liu. Research scheme of oral implant navigation surgery based on aruco marker. In *2022 International Conference on Computer Network, Electronic and Automation (ICCNEA)*, pages 336–340. IEEE, 2022.
[5] K Han, AM Garcia, IM Leo, MM Campo, C Muhammad, L Ortiz-Valles, GL Donohue, H Redeiss, and L Sherry. Unmanned aerial vehicle (uav) cargo system: senior design capstone project. In *Proceedings of the 2004 IEEE Systems and Information Engineering Design Symposium, 2004.*, pages 121–130. IEEE, 2004.
[6] AS Danilov, Ur D Smirnov, and MA Pashkevich. The system of the ecological monitoring of environment which is based on the usage of uav. *Russian journal of ecology*, 46(1):14–19, 2015.
[7] S Sankarasrinivasan, E Balasubramanian, K Karthik, U Chandrasekar, and Rishi Gupta. Health monitoring of civil structures with integrated uav and image processing system. *Procedia Computer Science*, 54:508–515, 2015.
[8] Andrej Babinec, Ladislav Jurišica, Peter Hubinskỳ, and František Duchoň. Visual localization of mobile robot using artificial markers. *Procedia Engineering*, 96:1–9, 2014.
[9] Mahammad Irfan, Sagar Dalai, Kaushal Kishore, Samarth Singh, and SA Akbar. Vision-based guidance and navigation for autonomous mav in indoor environment. In *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pages 1–5. IEEE, 2020.
[10] Sondre Sanden Tørdal and Geir Hovland. Relative vessel motion tracking using sensor fusion, aruco markers, and mru sensors. *Modeling*, 38(2):79–93, 2017.
[11] Detection of aruco markers. https://docs.opencv.org/4.x/d5/dae/tutorial aruco detection.html.
[12] Hirokazu Kato and Mark Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Proceedings 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR'99)*, pages 85–94. IEEE, 1999.
[13] M. Fiala. Artag, a fiducial marker system using digital techniques. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 590–596 vol. 2, 2005.
[14] Sergio Garrido-Jurado, Rafael Muñoz-Salinas, Francisco José Madrid-Cuevas, and Manuel Jesús Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, 2014.
[15] Wenyao Deng, Xuechao Duan, and Mingyang Jing. Dynamic target teal-time tracking technology tased on aruco marker and sparse sptical flow. *Electro-Mechanical Engineering*, 37(1):59–64, 2021.
[16] Ho Chuen Kam, Ying Kin Yu, and Kin Hong Wong. An improvement on aruco marker for pose tracking using kalman filter. In *2018 19th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, pages 65–69. IEEE, 2018.
[17] Adam Marut, Konrad Wojtowicz, and Krzysztof Falkowski. Aruco markers pose estimation in uav landing aid system. In *2019 IEEE 5th International Workshop on Metrology for AeroSpace (MetroAeroSpace)*, pages 261–266. IEEE, 2019.
[18] Jamie Wubben, Francisco Fabra, Carlos T Calafate, Tomasz Krzeszowski, Johann M Marquez-Barja, Juan-Carlos Cano, and Pietro Manzoni. A vision-based system for autonomous vertical landing of unmanned aerial vehicles. In *2019 IEEE/ACM 23rd International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, pages 1–7. IEEE, 2019.
[19] Zhe-Cheng Xu, Bin-Bin Hu, Bin Liu, XD Wang, and Hai-Tao Zhang. Vision-based autonomous landing of unmanned aerial vehicle on a motional unmanned surface vessel. In *2020 39th Chinese Control Conference (CCC)*, pages 6845–6850. IEEE, 2020.
[20] Bartosz Wieczorek, Łukasz Warguła, Mateusz Kukla, Arkadiusz Kubacki, and Jan Górecki. The effects of aruco marker velocity and size on motion capture detection and accuracy in the context of human body kinematics analysis. *Technical Transactions*, 117(1), 2020.
[21] Kaj Madsen, Hans Bruun Nielsen, and Ole Tingleff. *Methods for Non-Linear Least Squares Problems (2nd ed.).* 2004.