Journal of information and communication convergence engineering
http://jicce.org

J. Inf. Commun. Converg. Eng. 20(1): 65-72, Mar. 2022

Regular paper

A Study on Marker-based Detection Method of Object Position using Perspective Projection

Minjoo Park* **and Kyung-Sik Jang**

Department of Computer Engineering, Korea University of Technology and Education, Cheonan 31253, South Korea

Abstract

With the mark of the fourth industrial revolution, the smart factory is evolving into a new future manufacturing plant. As a human-machine-interactive tool, augmented reality (AR) helps workers acquire the proficiency needed in smart factories. The valuable data displayed on the AR device must be delivered intuitively to users. Current AR applications used in smart factories lack user movement calibration, and visual fiducial markers for position correction are detected only nearby. This paper demonstrates a marker-based object detection using perspective projection to adjust augmented content while maintaining the user's original perspective with displacement. A new angle, location, and scaling values for the AR content can be calculated by comparing equivalent marker positions in two images. Two experiments were conducted to verify the implementation of the algorithm and its practicality in the smart factory. The markers were well-detected in both experiments, and the applicability in smart factories was verified by presenting appropriate displacement values for AR contents according to various movements.

Index Terms: Augmented reality, Marker-based recognition, Perspective projection, Smart factory

I. INTRODUCTION

The paradigm of factories is changing from human-centered to smart factories in the wake of the fourth industrial revolution. Machines are used to build flexible and self-adapting production in smart factories [1]. Paradigm replaces workers with machines, but a complete replacement is impossible in the current Industry 4.0 trend. Workers and machines need to communicate and interact with each other to achieve better results [2].

Industry 4.0 provides workers with integrated processing and communication capabilities. They are equipped with devices to enable human-machine interaction and to support decision-making in industrial environments [2]. A powerful human-machine technology currently used in the Factory 4.0 paradigm is augmented reality (AR), which helps workers acquire the necessary proficiency [3, 4] in the field. Maintenance information, production monitoring, technology control, and production design visualization are some AR applications utilized by workers [5]. AR in smart factories is a highly promising user interface [6]. Therefore, AR information and virtual object processing should be presented as intuitively as possible so that there is no discrepancy between the real world and the digital world [3].

Fig. 1 demonstrates the current use of AR in a smart factory. Augmented content is delivered to users as infotext, graphs, and three-dimensional modeling. Fig. 1(a) is a view straight from the mechanical part immediately after scanning the marker and launching the AR application. The contents are placed on the sides of the screen to prevent blocking the actual production and assembly lines behind. The view after rotating the mobile device to the right is shown in Fig. 1(b). Although the position of the infographic has changed, the orientation and size of the content are not reflected with the

Received 17 February 2021, **Revised** 24 February 2022, **Accepted** 10 March 2022
***Corresponding Author** Minjoo Park (E-mail: cathymjp@koreatech.ac.kr, Tel: +82-41-560-1352)
Department of Computer Engineering, Korea University of Technology and Education, Cheonan 31253, South Korea.

Open Access https://doi.org/10.6109/jicce.2022.20.1.65 print ISSN: 2234-8255 online ISSN: 2234-8883

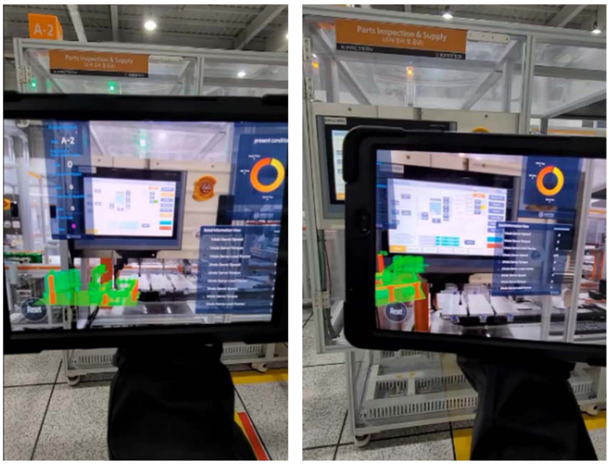


Fig. 1. Example of augmented reality usage in a smart factory.

user's movements, blocking the actual working space.

Virtual objects are superimposed on the real scene in AR. The visualized augmented data can align with the real world once the user's position, that is, the position of the camera, is recognized in the real world [7, 8]. The information displayed on the screen should be perfectly displayed by reflecting the movement of the worker. Predicting camera movement is important to adjust content visualized via mobile devices. In this study, we propose a method using perspective projection for determining the change in the user's position by using the change in captured view. The difference of the equivalent marker point in the reference and the moved image is used to calculate the new angle, size, and position of the virtual object in AR.

The purpose of this research is as follows. First, we verify the effect of calculating the change in the user's viewpoint using simple color markers that have not been performed before. Our algorithm uses four marker points to calculate the user's displacement. Second, we verify the effect of our application without initial setup. The algorithm quickly calculates the displacement of the marker points in two images. Third, we verify the effect of locating markers in a smart factory. From a distance, the current markers are obscured by mechanical parts, but with the smart factory demonstration, we have proven otherwise.

II. THEORY OF DETECTION METHOD

A. Perspective Projection

As one of the most popular technologies today, countless AR applications are on the market to realize a fantasy world. Determining the optimal orientation and UI placement for virtual content [7] is one of the research interests. The key to AR content calibration is the degree of adjustment of the content with the camera position. UI content should blend so perfectly that users do not feel that AR is unrealistic.

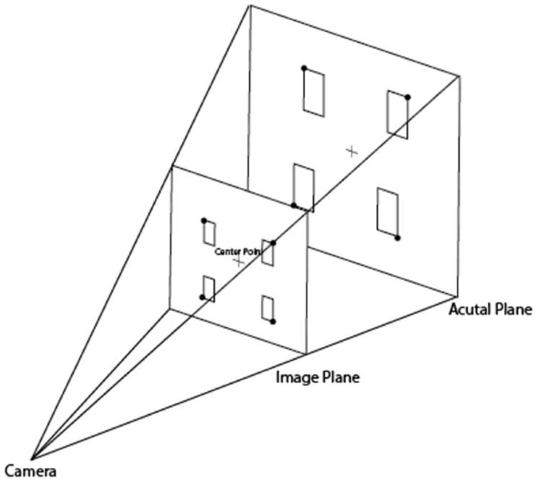


Fig. 2. Perspective projection of markers in the actual plane to the image plane from the camera.

Perspective projection establishes the camera's look parameters by mapping three dimensions into two dimensions. Given three-dimensional point coordinates, the corresponding position of the camera can be estimated [9]. Using perspective projection, the world coordinate system (X, Y, Z) for each three-dimensional scene point is projected onto the image plane as (x, y), losing depth information.

Perspective projection is similar to the way the human eye sees the world. Objects further away from the camera appear smaller, and objects closer to the camera appear larger. Everything in the three-dimensional plane is projected onto the two-dimensional image plane with unique coordinates. For our algorithm, we used these coordinates to calculate the camera displacement. Fig. 2 depicts how a three-dimensional plane is projected onto a two-dimensional plane. The camera on the device captures the actual plane, and the scene is projected onto the image plane where the AR content is drawn.

B. Marker Methodology

AR draws digital information and virtual objects in harmony with the user's real-life in real-time using the camera of a mobile device. Knowing what the user is looking at is critical to realizing this fascinating world. Various algorithms have been proposed to estimate the user's position. One approach uses accelerometer and gyroscope IMU sensors built inside mobile devices to estimate the orientation of the device. However, it is difficult to determine its global position in space as a standalone system and as a large integration drift [10]. Devices like the Kinect sensor estimate the depth lost in the scene projection. However, it struggles detection in various lighting environments and small fields of view [10]. Wang et al. [11] used infrared markers to calculate the camera's position and orientation, which required an additional infrared projector and infrared camera for detection and calculation. These methods are not suitable for real-time calculations in smart factories.

Marker-based detection was introduced to understand the camera position and orientation of the marker in AR [12]. It is the most widely used system in AR due to its high accuracy. Markers are designed differently depending on their intended use [13]. Pattern matching is used to determine the degree of rotation from the referenced image. Identification based on correlation, digital codes, and topology has also been proposed to extract features from markers [13]. The most used visual markers in AR today are fiducial markers such as images or QR codes [14]. Some examples are topological region adjacency-based TopoTag [15], digital code signal-based Fourier tag [16], and correlation-based marker ARToolkit [17]. These markers share a common design by having a unique pattern inside, which is converted into a matrix and decoded by dictionary matching to redeem the marker ID and orientation [18-20].

Current marker recognition is easy to implement and can accurately estimate the camera position [21]. However, these are suited for stationary objects under a controlled environment, [22] and a limited number of matrix combinations affect real-time coordination. These markers must be properly positioned in advance to ensure the content is displayed in the correct position [12]. Fiducial markers are recognized by placing them on a simple background up close to the camera. However, these are not suitable for work environments with complex backgrounds. In a smart factory full of objects and colors, markers can be misplaced easily. In addition, a certain distance must be secured to capture the desired production and assembly line on the screen. Subakti et al. [22] proposed a markerless AR system to detect and visualize machine information in indoor smart factories. Despite the high recognition accuracy, AR content did not correlate with the user's movements.

Color markers function as reference points for camera displacement. Color markers have the advantage of ensuring detection in low-quality images and complex backgrounds, and being small with appropriate color selection [23]. Perspective projection is used to get the marker coordinates projected onto the image plane. Coordinate differences between the reference markers and the moved markers reflect changes in depth and distance from the camera to the scene. These are utilized to calculate the new rotation, size, and location values of the AR content on the UI.

III. MARKER-BASED POSITION RECOGNITION

A. Overview

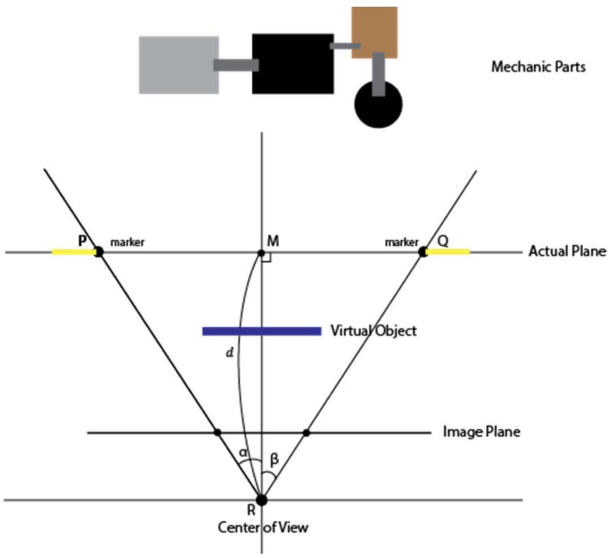


Fig. 3. Overview of the marker-based position recognition model. The marker coordinates are used for adjusting the attribute of virtual objects.

The proposed three-layer marker-based recognition model is shown in Fig. 3 for two markers. Markers in the three-dimensional Actual Plane are projected onto the two-dimensional image plane via perspective projection. Between the two planes is a virtual object on which AR content is displayed. Adjustment values for virtual objects are estimated from the displacement of the original and moved marker positions.

The camera is located at the center of view R, the center of two marker points P and Q projected onto the image plane. In the reference image, the angles α and β of each marker point to R are used to compute the angle values on both the x- and y-axes using basic trigonometry. The angle from the marker coordinate (x_2, y_2) to R (x_1, y_1) is calculated using equation (1) for the x-axis and (2) for the y-axis.

$$\alpha_x = \tan^{-1}\frac{x_1 - x_2}{y_1 - y_2} \tag{1}$$

$$\alpha_y = \tan^{-1}\frac{y_1 - y_2}{x_1 - x_2} \tag{2}$$

The new size and position are computed using the point M between P and Q located on a line PQ perpendicular to R. The difference between the distance *d* from M to R and *d'* from moved point M' to R is the scaling value, and the coordinate change between M and M' is the new location value. The new layout of the AR content is determined by α, β, d, and M in the original position and the corresponding values in the moved image.

B. Angle Adjustment

As the user moves the device to a new location, the angle of the camera towards the workstation changes. Content must be rotated accordingly to provide an equivalent view in its original location. The centerline of the image serves as a reference for where the user was facing in the original image. The angle change between the original image and the moved image is the angle adjustment value.

The α and β values from the original image and the angles α' and β' for the moved marker points P' and Q' calculated using equations (1) and (2) are used in the calculations. Use equations (3) and (4) to find α^* and β^* for two markers, respectively. The new angle values α' and β' must maintain the ratio of α and β to the old R.

$$\alpha^* = \frac{\alpha}{\alpha + \beta} \times \alpha' + \beta' \tag{3}$$

$$\beta^* = \frac{\beta}{\alpha + \beta} \times \alpha' + \beta' \tag{4}$$

The difference between α', β' and α^*, β^*, γ, indicates how much the virtual object needs to be rotated to maintain the viewing angle. Fig. 4 shows the change in the virtual object with γ applied.

C. Size Adjustment

In perspective projection, the size of the virtual object is proportional to the distance from the camera to the actual plane. Distance in this context is the distance from the center of the camera to the marker, not the physical distance between the marker and the mobile device. For calculation, we use the calculated point M and the equivalent point M' from the moved image. M' lies on P'Q' with γ angle tilted. The coordinate of M' is calculated with equation (5).

$$M' = \left(\frac{f \sin\alpha\, x_2 + g \sin\beta\, x_1}{f \sin\alpha + g \sin\beta}, \frac{f \sin\alpha\, y_2 + g \sin\beta\, y_1}{f \sin\alpha + g \sin\beta} \right) \tag{5}$$

Using the distance function, the difference between M and M' for R gives the distance change, d^*. This tells how much the camera has moved from its original point. Content must be sized accordingly. Longer d' indicates a farther plane, reducing the content size, and shorter d' indicates a closer plane, increasing the content size. All two pairs of marker changes are summed to the final size of the virtual object. The virtual object size change is shown in Fig. 5.

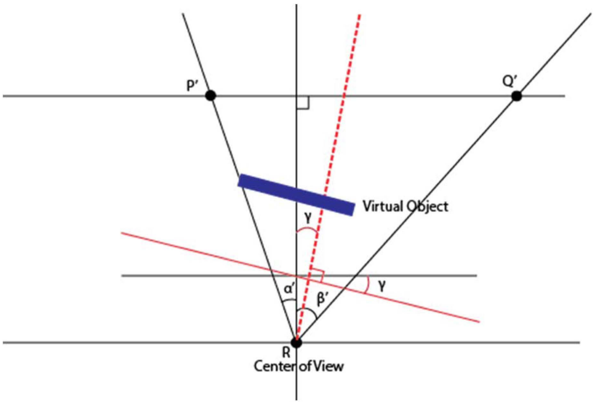


Fig. 4. Angle adjustment of the virtual object with γ value.

D. Location Adjustment

The location of the content on the UI should not be fixed and must change as the user moves. The location adjustment value is equal to the amount of displacement from point M to point M'. The new x-coordinate is the average x-axis change of the top and bottom markers, and the y-coordinate is the average y-axis change of the left and right markers.

Fig. 6 visualizes the x change for the top two markers. Change in y is the same concept using the left and right markers. The coordinates of the top and bottom markers change by y, and the coordinates of the left and right markers change by x. Since α is calculated differently for the top and bottom and left and right markers, changes to the other

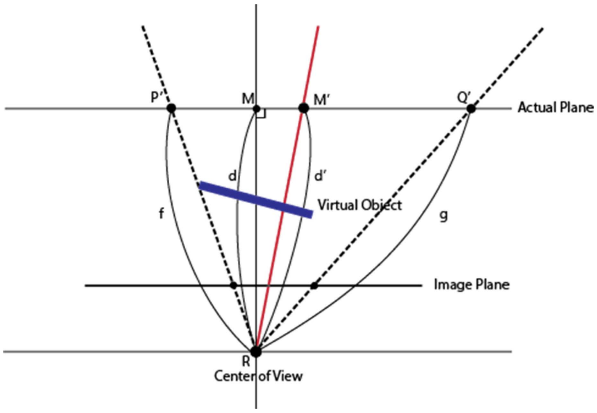


Fig. 5. Size adjustment of the virtual object after calculating the total distance change.

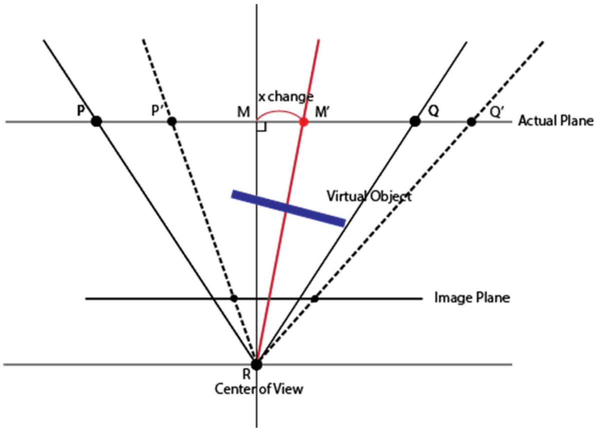


Fig. 6. Location adjustment of the virtual object after calculating the x displacement.

coordinate values are ignored. The final coordinates are the combination of these two values. The new position of the virtual object for the top x-axis is shown below.

The new location of the virtual object is determined by combining the values obtained from sections A, B, and C. Adjustments can be easily calculated using the coordinates of each marker point and the center point of the image.

IV. RESULTS AND DISCUSSION

A. Experimental Implementation Overview

Two experiments were performed to verify the execution of the algorithm and its practicality in the smart factory. The displacement of the marker was calculated by applying the algorithm to the image taken from the central location and after moving the camera. Pictures were taken with the smartphone's front camera, and the algorithm was implemented using OpenCV in Python.

Four 7.6 cm × 2.5 cm rectangular sticky notes were used as markers. We calculated the x-movement with makers at the top and bottom and y-movement with two markers on the left and right. Neon yellow, a rare color in life, was chosen as the marker color. For all vertices of the four markers, the vertices closest to the center point of each marker were used as calculation points. We chose four points as they are stronger than three and are the most optimal number for calculating the x and y axes [24, 25].

B. Application in a lab setting

1) Experimental setup

The first experiment was conducted in a lab setting to verify the execution of the algorithm. Images were taken with an LG G7 phone with a focal length of 300 mm fixed on a tripod. For a sophisticated environment, the device was installed on a wooden board. Pictures were taken at 100 cm and 150 cm away parallel to the marker. The different views that users could create with the AR devices were mimicked by varying camera positions. A goniometer was used to measure the rotation, and a digital inclinometer application was used to check the level of the smartphone. The center of the camera was the center of the four markers for all reference images. For better detection and computation, markers were placed on black paper. All markers were placed 10 cm apart on the x- and y- axes from the center point, equally spaced around the center point. At the two furthest vertices, top and bottom horizontal markers were 25 cm apart, and left and right vertical markers were 35.2 cm apart.

2) Experimental results

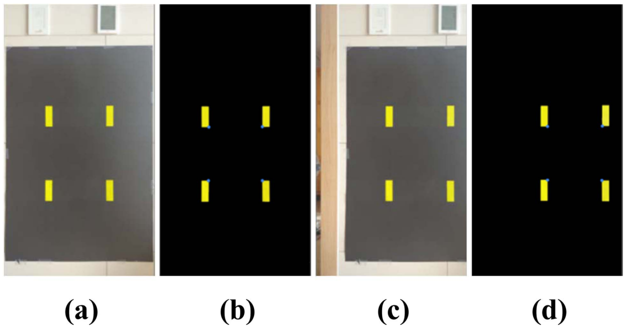


Fig. 7. Visualization of how markers are detected in the input image (a) initial position (b) marker detected in the initial position (c) moved position (d) marker detected in moved position.

Fig. 7 demonstrates the marker detection procedure. The camera was positioned 100 cm away from the markers and was translated 10 cm to the left. (a), (b) are reference images; (c), (d) are moved images. (a), (c) are original photos and (b), (d) are photos with color filters applied. Four yellow markers are detected in the image, and the vertices used in the calculation are marked with blue dots.

For the laboratory experiment, three camera movements were investigated. From the reference position, the camera was translated right, left, down, rotated right, left, down, and zoomed out. All translation and zoom results were calculated every 5 cm, left-right rotation every 2 degrees, and downward rotation every 10 degrees. The results of all the rotation (γ), distance (d), and location (x, y) calculations are in pixels.

Here are the results obtained by setting the camera 100 cm away from the marker. Table 1 shows the left and right translation results and Table 2 shows the down translation results. The increase in d and x indicates that the center point of the

Table 1. Translation of the camera right and left adjustment values

Trans (cm)		Right			Left		
		5	10	15	5	10	15
γ	x	10.89	33.15	46.85	11.39	22.87	46.75
	y	0.19	0.06	0.25	0.01	0.04	0.05
d		4.68	34.93	128.10	4.11	32.46	128.74
x, y	x	22	48	75	-18	-42	-72
	y	0	-1	0	-1	0	0

Table 2. Translation of the camera down adjustment values

Trans (cm)		Down				
		3	6	9	12	15
γ	x	0.59	0.15	0.65	0.20	0.19
	y	8.23	.17.00	28.33	38.31	47.23
d		0.38	7.32	20.36	64.74	132.05
x, y	x	0	0	1	0	0
	y	-10	-21	-36	-53	-73

second image has shifted further horizontally from its original position.

The results of rotating the camera left and right are shown in Table 3, and the results of rotating the camera down are shown in Table 4. Similar to translation, the left-and-right affects the x values and the up-and-down affects the y values.

Table 5 shows the results of moving the camera backward in proportion to the reference point. Since the distance was the only variable, rotation and location changes are negligible. As distance decreases, virtual objects should appear smaller.

The purpose of the lab experiment is to see if the r, d, x, and y values change as expected as the camera moves. The result values indicate the amount of movement of AR content to be displayed on the screen compared to the existing location. Therefore, the accuracy of the values does not affect the hypothesis. The consistent trend of this experimental result proves that the algorithm can reflect the camera movement in various positions.

C. Marker Identification at Smart Factory

1) Experimental setup

The second experiment was conducted in an actual smart factory. The objective was to test whether the markers were detectable in an actual smart factory environment using the algorithm confirmed in the previous section. Pictures were taken with the Samsung Galaxy Note 20 in hand. The position of the markers and the distance of the camera were set randomly, and the pictures were taken without rules.

Table 3. Rotation of the camera right and left adjustment values

Rotation (°)		Left			Right		
		2	4	6	2	4	6
γ	x	11.22	21.67	33.87	10.36	22.50	32.95
	y	0.12	0.13	0.22	0.05	0.50	0.30
d		2.64	11.29	37.58	2.81	12.07	35.77
(x, y)	x	-13	-26	-45	13	28	45
	y	1	1	1	0	1	0

Table 4. Rotation of the camera down adjustment values

Rotation (°)		10	20	30	40	50
γ	x	0.35	0.76	0.74	0.42	0.15
	y	12.70	26.58	46.60	59.24	65.04
d		4.34	17.11	126.68	28.722	355.29
(x, y)	x	0	1	1	0	1
	y	15	34	71	117	155

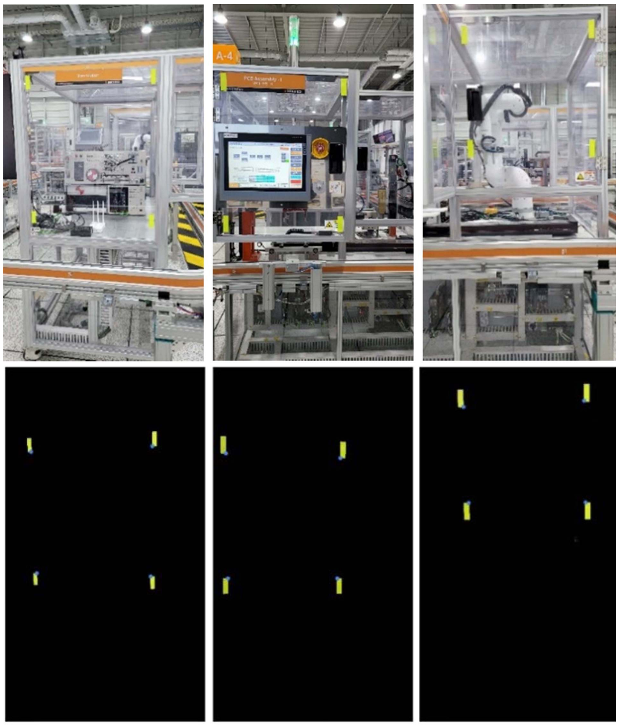
Table 5. Zooming of the camera out adjustment values

Zoom out (cm)		5	10	15	20
γ	x	0.23	0.91	0.21	0.21
	y	0.32	0.79	0.33	0.38
d		-13.47	-24.46	-34.46	-43.46
(x, y)	x	0	0	0	0
	y	1	1	1	0

A great number of production modules make up the smart factory [2]. Production and assembly lines are lined with constantly moving machines and robots. Because every station has a different mechanical design, the marker detection process is challenging. It should not be vulnerable to many factors such as lighting, color, layout, and background noises. Five stations were chosen for detection and verification. The first part of the experiment placed markers on the plastic shields of the workstations, and the second part of the experiment placed the markers on the actual machines. Markers were attached differently according to the status of each station.

2) Experimental results

Lighting can reflect off the plastic shield and cast shadows, making the lighting setup an important consideration for placing markers. To test the phenomenon, markers were applied directly to the plastic shield in three different stations.



(a) (b) (c)

Fig. 8. Marker recognition by attaching the markers on a plastic shield.

In Fig. 8(a), markers were placed on a shield with a sta-

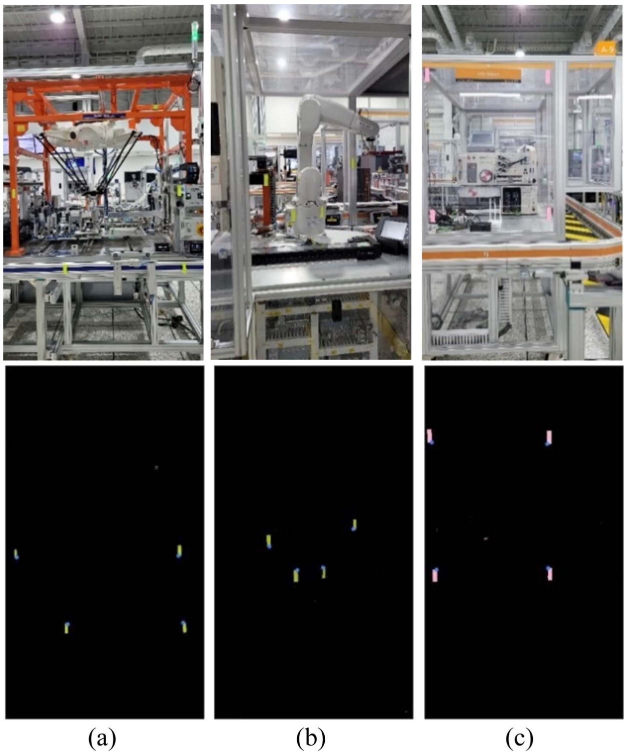


Fig. 9. Marker detection by attaching the markers directly to the machine.

tionary machine behind them. Markers were detected well in thirty-one pictures despite the yellow tag rail line on the back. Operating machines change the background of the photo, which can affect color recognition. This effect was experimented with taking thirty-one pictures shown in (b). The results show that the marker detection was not affected by the moving machines behind it.

Not all stations have just plain shields, such as the shield with controller monitor in (c). The markers must be detected even from these monitor lights. Results from fifteen images taken at this station showed that these lights did not affect marker detection.

Results for three different stations proved marker detection under different light sources. However, attaching markers directly to the shield may obscure the machine and obstruct a person's view. This problem is solvable by attaching the markers directly to the machine. In the second part of the experiment, the markers were attached directly to the operating machines.

Fig. 9(a) and (b) are the results of attaching the markers to the machines. Twenty-three pictures for (a) and fourteen pictures for (b) show that the markers can be detected even under these conditions.

Choosing the right marker color for the environment is important. Neon yellow is not a common color in everyday life, but it is widely used in factories as a warning sign. The range of the yellow had to be narrowed in this experiment to distinguish the two. In the last experiment, pink markers were used to check the applicability of different colors. Fifteen pictures were taken and the results in (c) show that the application is applicable with the right marker color.

V. CONCLUSIONS AND DISCUSSION

In this paper, we presented an approach to compute novel attributes of AR content using the displacement of projected markers in two images without initial configurations. Using colored markers to calculate changes in the field of view has never been presented before. By performing lab experiments, we obtained a good estimate for the new virtual content's angle, size, and location change. Through experiments in the smart factory, we demonstrated the feasibility in a real-world environment. Two experiments suggested a new possibility for position detection using markers in the smart factory. As a future study, we plan to find better border detection, size and marker set-up, and distinction of stations using markers.

ACKNOWLEDGEMENTS

This result was supported by the Regional Innovation Strategy (RIS) through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (MOE) (2021RIS-004).

REFERENCES

[1] A. Syberfeldt, O. Danielsson and P. Gustavsson, "Augmented reality smart glasses in the smart factory: Product evaluation guidelines and review of available products," *IEEE Access*, vol. 5, pp. 9118-9130, 2017.

[2] H. Kagermann, W. Wahlster, and J. Helbig, "Securing the future of german manufacturing industry: Recommendations for implementing the strategic initiative INDUSTRIE 4.0," ACATECH - German National Academy of Science and Engineering Tech. Rep., 2013.

[3] D. Gorecky, M. Schmitt, M. Loskyll, and D. Zühlke, "Human-machine-interaction in the industry 4.0 era," in *2014 12th IEEE International Conference on Industrial Informatics (INDIN)*, pp. 289-294, 2014.

[4] J. Um, J. Popper, and M. Ruskowski, "Modular augmented reality platform for smart operator in production environment," *2018 IEEE Industrial Cyber-Physical Systems (ICPS)*, pp. 720-725, 2018.

[5] A. Ivaschenko, A. Khorina, and P. Sitnikov, "Accented visualization by augmented reality for smart manufacturing applications," *2018 IEEE Industrial Cyber-Physical Systems (ICPS)*, pp. 519-522, 2018.

[6] V. Paelke, "Augmented reality in the smart factory: Supporting workers in an industry 4.0. environment," in *Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA)*, pp. 1-4, 2014.

[7] D. Liu and J. W. Park, "A study on the icon and layout design of device UI in augmented reality," in *2019 2nd World Conference on Mechanical Engineering and Intelligent Manufacturing (WCMEIM)*,

pp. 245-248, 2019.

[8] I. Malý, D. Sedláček and P. Leitão, "Augmented reality experiments with industrial robot in industry 4.0 environment," *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*, pp. 176-181, 2016.

[9] R. M. Haralick, "Determining camera parameters from the perspective projection of a rectangle," *Pattern Recognition*, vol. 22, pp. 225-230, 1989.

[10] E. van der Kruk and M. M. Reijne, "Accuracy of human motion capture systems for sport applications; state-of-the-art review," *European Journal of Sport Science*, vol. 18, no. 6, pp. 806-819, 2018.

[11] T. Wang, Y. Liu, and Y. Wang, "Infrared marker based augmented reality system for equipment maintenance," *2008 International Conference on Computer Science and Software Engineering*, 2008, pp. 816-819.

[12] A. López-Cerón, "Accuracy analysis of marker-based 3 D visual localization," in *XXXVII Journadas de Automatica Workshop*, 2016.

[13] J. Kohler, A. Pagani, and D. Stricker, "Detection and identification techniques for markers used in computer vision," *OASIcs-OpenAccess Series in Informatics*, vol. 19, 2011.

[14] H. Subakti and J. Jiang, "A marker-based cyber-physical augmented-reality indoor guidance system for smart campuses," in *2016 IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pp. 1373-1379, 2016.

[15] G. Yu, Y. Hu, and J. Dai, "TopoTag: A robust and scalable topological fiducial marker system," *IEEE Transactions on Visualization and Computer Graphics*, 2020.

[16] J. Sattar, E. Bourque, P. Giguere, and G. Dudek, "Fourier tags: Smoothly degradable fiducial markers for use in human-robot interaction," *Fourth Canadian Conference on Computer and Robot Vision (CRV '07)*, pp. 165-174, 2007.

[17] ARToolKit: http://www.hitl.washington.edu/research/ shared space/ download/.

[18] M. Fiala, "ARTag, a fiducial marker system using digital techniques," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2, pp. 590-596, 2005.

[19] A. Sagitov, K. Shabalina, R. Lavrenov, and E. Magid, "Comparing fiducial marker systems in the presence of occlusion," in *2017 International Conference on Mechanical, System and Control Engineering (ICMSC)*, pp. 377-382, 2017.

[20] T. Kawano, Y. Ban, and K. Uehara. "A coded visual marker for video tracking system based on structured image analysis," *ISMAR*, pp. 262-263, 2003.

[21] M. Fiala, "Designing highly reliable fiducial markers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 7, pp. 1317-1324, 2010.

[22] H. Subakti and J. Jiang, "Indoor augmented reality using deep learning for industry 4.0 smart factories," in *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, pp. 63-68, 2018.

[23] A. Dell'Acqua, M. Ferrari, M. Marcon, A. Sarti, and S. Tubaro, "Colored visual tags: a robust approach for augmented reality," in *IEEE Conference on Advanced Video and Signal Based Surveillance*, 2005, pp. 423-427, 2005.

[24] M. L. Liu and K. H. Wong, "Pose estimation using four corresponding points," *Pattern Recognition Letters*, vol. 20, no. 1, pp. 69-74, 1999.

[25] W. J. Wolfe, D. Mathis, C. W. Sklair, and M. Magee, "The perspective view of three points," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 1, pp. 66-73, 1991.

Minjoo Park

received her Bachelor in Engineering in Information Systems from Stony Brook University, Stony Brook, NY, USA in 2018. She received her Master's in Engineering in Computer Science in Korea Technology and Institute University, Cheonan, Korea in 2021. Her research interests include image processing, computer vision, and smart factory.

Kyung-Sik Jang

received the B.S degree in electronic engineering from Korea University, 1987, and the M.S. degree in electrical and electronic engineering from KAIST, in 1989, and Ph.D degree in electrical and electronic engineering from Tokyo Institute of Technology, 1998.His research interests include embedded system, natural language processing.

https://doi.org/10.6109/jicce.2022.20.1.65