

Study of Marker Array List Method for Augmented Reality Service Based Smart Home

Donghyun Kim, Yohwan So, Seoksoo Kim¹

*Dept of Multimedia Engineering, Hannam University, Daejeon, South Korea
Kimdh1986@hnu.kr, artso@hnu.kr, sskim0123@naver.com*

Abstract

More recently, the demand for smart home services is on the rise due to the advanced computing devices and network technology, calling for more stable and efficient visualization and simulation. This has led to a lot of research efforts on AR for virtual smart home services. AR may use a non-marker-based method or a marker-based method. In case of AR services for the indoor environment, however, powerful detection of an object cannot be done solely by a computerized perspective and thus a marker-based technique is applied to smart home services in order to detect an object more accurately. Nevertheless, the marker-based AR may detect markers of similar patterns as the same markers, which causes repeated detection of markers. In this study, therefore, we have explored a method of augmenting a large number of objects without creating a multitude of markers but by creating a list of a marker array, which can increase marker efficiency and avoid repeated detection of markers.

Keywords: *Smart Home, Augmented Reality, Multi Marker Detection, Marker Array*

1. Introduction

A smart home is a personal house supporting automation. That is, various automatic functions such as illumination of an intelligent building, temperature control, door and window control, security systems, and so on, are applied to such a personal house. Home theater control, efficient automatic intelligent air conditioners, crime prevention, and medical system access can be example services also.

The demand for smart home services is on the rise due to the advanced computing devices and network technology, calling for more convenient user interface technologies. Thus, more research is being made in order to apply augmented reality (AR) to smart home services, offering more stable and efficient visualization and simulation. In addition, a lot of research efforts are underway on AR for an indoor environment [1].

AR is an area of virtual reality in which a virtual object (or information) is combined with the real world so that it may look like an object already existing in the real world. The computer graphic technique is also called mixed reality, and it could be based on a marker or not [2]. A non-marker-based method decides a direction of a camera according to features that may be created in the real world such as a dot, line, edge, and texture. In the marker-based method, the direction of a camera is decided using markers on the plane. On the white plane there are black squares and various patterns. In case of AR services for the indoor environment, however, powerful detection of an object cannot be done solely by a computerized perspective and thus the

¹ Corresponding author

direction of a camera cannot be easily decided. Thus, a marker-based technique is applied to smart home services in order to detect an object more accurately.

Nevertheless, the marker-based AR may detect markers of similar patterns as the same marker. So, when marker patterns are created in order to augment a large number of objects, the patterns should not be similar, which limits the number of markers.

In order to solve this problem, this study explores a method of augmenting a large number of objects without creating a multitude of markers but by using a marker array with no similar patterns. In addition, a method of increasing marker efficiency and making a marker array, avoiding repeated detection of markers, is introduced in this study.

2. Related Works

2.1. Template Matching

For template matching, a basic image of $T=\{T(i,j); 0\leq i\leq N, 0\leq j\leq N\}$ is moved in every direction in a search image of $S=\{S(x,y); 0\leq x\leq N, 0\leq y\leq N\}$ as depicted in Fig 1 in order to calculate correlation. We could find a location with the greatest CC (correlation coefficient) by calculating correlation K^2 times and using f , which calculates correlation from $SS(x, y)$ to $S(x+K-1, y+K-1)$. Here, the computation amount for the entire images is $K^2(N+K+1)^2$. In case the search image is bigger than the basic image, $K\ll N$ is valid and an approximate expression for the computation amount is K^2N^2 [3]. formula (1) is a general expression for a template matching algorithm.

$$SS(x, y) = \sum_{i=0}^{K-1} \sum_{j=0}^{K-1} f(S(x+i, y+j), T(i, j)) \quad (1)$$

CC(Correlation Coefficient)[4] and SAD(Sum of Absolute Difference)[5] are methods of finding correlation for template matching For CC, formula f is multiplication in formula (1) as in the case of formula (2). For SAD formula f is subtraction and the absolute value in formula (1) as in the case of formula (3).

$$SS_{CC}(x, y) = \sum_{i=0}^{K-1} \sum_{j=0}^{K-1} f(S(x+i, y+j) \cdot T(i, j)) \quad (2)$$

$$SS_{SAD}(x, y) = \sum_{i=0}^{K-1} \sum_{j=0}^{K-1} f(S(x+i, y+j) - T(i, j)) \quad (3)$$

However, both methods require a great amount of computation and, thus, a large amount of time and a gate is unavoidable if applied to data with great bit depth. So, we used a statistical method as pre-processing. Hence, template matching is carried out after applying Binary Map[6], Low-Level Feature Map[7] using a grade and a boundary line of an image, and MPPM(Moment-Preserving Pattern Matching) [8] using 2-bit quantization.

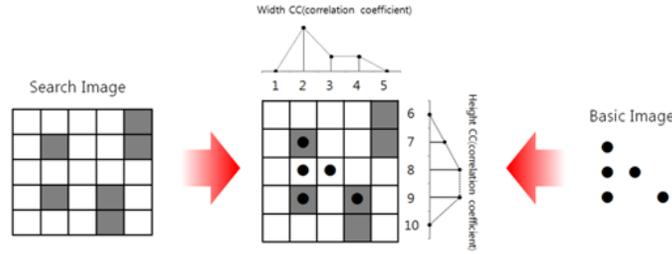


Figure 1. Template Matching Concept

2.2. Hough Transform

The Hough transform is a method for detecting straight lines, circles or other simple shapes in an image[9]. The Hough transform is based on the fact that there are countless straight lines passing through any point in the binary image[10]. If a line has a y-intercept of b and a slope of a ($y = ax + b$), a point on the (x, y) coordinate plane is expressed as a straight line on the (a, b) coordinate plane. In the case where no zero pixels in the input image are all expressed as straight lines on the (a, b) coordinate plane image and the pixel values located where the lines pass through are accumulated, a line on the (x, y) coordinate plane is shown to have a local maximum value on the (a, b) coordinate plane. Since all loci of the lines expressed by each point are summed, the (a, b) coordinate plane is commonly called an accumulator plane[11].

However, the method of expressing a straight line with a slope and a y-intercept is not appropriate to express all straight lines on the (x, y) coordinate plane. This is because to express the common straight lines of which slopes range from $-\infty$ to $+\infty$ in the binary image is difficult. Therefore, in practical realization, it is expressed as a point, (ρ, θ) on the polar coordinate system by means of another method. A line expressed as (ρ, θ) on the polar coordinate system means a straight line perpendicular to a line passing through this point and the origin, which is expressed in the following numerical formula (4):

$$p = x \cos \theta + y \sin \theta \quad (4)$$

A point (x_0, y_0) on the image coordinate system, as shown in Figure 2(A), becomes a point in Figure 2(B) where many straight lines expressed as (ρ, θ) on the polar coordinate system intersect and it can be expressed as a line in Figure 2(C) on the (ρ, θ) plane as follows

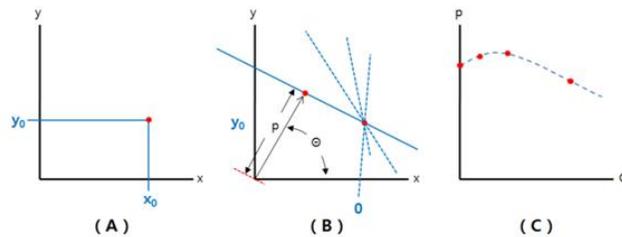


Figure 2. Common accumulator plane expressed on the polar coordinate system. (A)A point (x_0, y_0) . (B) The image coordinate system becomes a point. (C) The polar coordinate system intersects and it can be expressed as a line.

3. Reference marker detection

Figure 3 shows the flow of basic reference marker detection suggested in this study. There are 4 steps in the process. First, the histogram of an input image is smoothed and the image is converted to a binary image using Otsu critical values(pre-processing of an image). Second, a marker is detected from the pre-treated image using a boundary tracing algorithm. Third, the marker is recognized by the PPHT(Progressive Probabilistic Hough Transform) technique. Fourth, markers are matched by template matching in order to detect a reference marker.

3.1. Image Preprocessing

The greatest challenge of marker detection is the recognition rate of a small-size marker is low in case of high brightness (high visibility) while a small-size marker detection is difficult in case of low brightness. Therefore, pre-treatment of an image, which includes smoothing of a histogram and binarization, is necessary for marker detection.

In this study, we converted an input image to a gray image and histogram equalization[12] so as to minimize detection errors that may arise from external factors such as light and illumination. Then, the image is converted to a binary image using Otsu critical values[13].

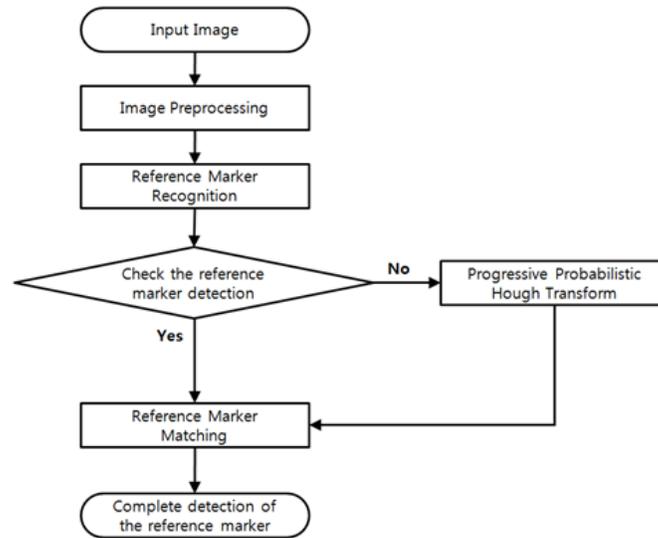


Figure 3. Flowchart of Reference Marker Detection

3.1.1. Histogram equalization: Smoothing of a histogram is a method of improving an image quality by evenly distributing histograms of an input image as follows.

Formula (5) shows a histogram $h(X_k)$ for an image $X=\{X_k(i, j)\}$ with a coordinate (i, j) and $\{X_0, X_1, \dots, X_{L-1}\}$ which is number of brightness. PDF(Probability Density Function) is $p(X_k)$ as depicted by formula (6).

$$h(X_k) = n_k, (X_0 \leq X_k \leq X_{L-1}) \quad (5)$$

$$p(X_k) = \frac{n_k}{n} \text{ for } k = 0, 1, \dots, L - 1 \quad (6)$$

In formula (2) n and n_k refer to the total pixel number of X , which is the original image, and the pixel number of X_k , which is brightness. $p(X_k)$ is a histogram showing the frequency of brightness X_k . $c(X)$ is a CDF(Cumulative Density Function) based on $p(X_k)$, a PDF, as depicted in formula (7).

$$c(X) = \sum_{i=0}^k p(X_i), \text{ for } k = 0, 1, \dots, L-1 \quad (7)$$

In order to smooth a histogram $f(X_k)$, a conversion function of the original image is created by using $c(X)$, a CDF obtained from formula (7), as depicted in formula (8). Then, an image with a histogram of uniform distribution is mapped. formula (9) shows Y , a conversion image obtained from the mapping.

$$f(X_k) = X_0 + (X_{L-1} - X_0) \cdot c(X) \quad (8)$$

$$Y = f(X(i, j)) | \forall X(i, j) \in X \quad (9)$$

Figure 4 shows the process of histogram smoothing for an input image. (A) of Figure 4 is an original image obtained from a web cam and (C), histogram distribution of the original image. (B) of Figure 4 shows the result of histogram smoothing for the original image and (D), histogram distribution of a smoothed image.

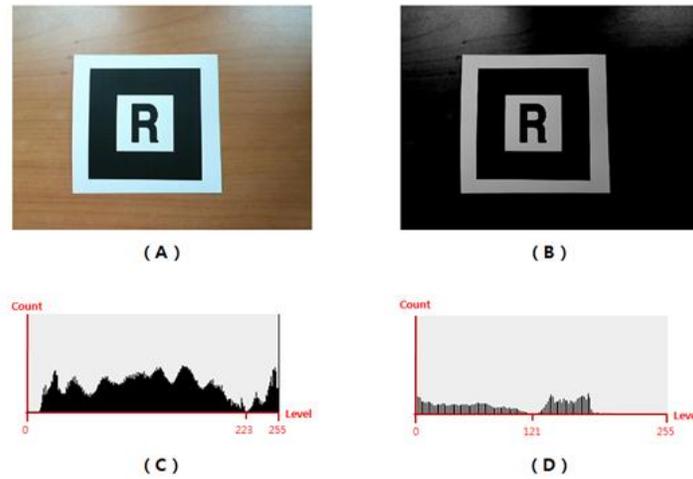


Figure 4. Histogram equalization (A) Original image (B)Histogram equalization image (C) Histogram distribution of origin image (D) Histogram distribution of histogram equalization image

3.1.2. Binarization: Ostu critical value is a binarization method of obtaining bimodal distribution from a frequency histogram of a grey image without using space cohesion information. The frequency histogram obtained by formula (10) from a smoothed image is normalized as depicted in formula (11).

$$\begin{aligned} w_1 &= \sum_{z=0}^T H(z), & w_2 &= \sum_{z=0}^{L-1} H(z) = 1 - w_1, \\ u_1 &= \sum_{z=0}^T \frac{zH(z)}{w_1} = 1, & u_2 &= \sum_{z=T+1}^{L-1} \frac{zH(z)}{w_2}, & u_t &= \sum_{z=0}^{L-1} zH(z) \end{aligned} \quad (10)$$

$$H(z) = \frac{\text{Histogram}(z)}{\sum_{z=0}^{L-1} \text{Histogram}(z)} \quad (11)$$

In case two groups, C_1 , C_2 , are created from the normalized histogram based on T , a grey level, C_1 is composed of grey level $[T+1, L-1]$ Then, dispersion of the two groups is calculated by formula (12), and a location that the two groups become maximum is calculated as depicted in formula (13).

$$\sigma_B^2(T) = w_1 (u_1 - u_t)^2 + w_2 (u_2 - u_t)^2 = w_1 w_2 (u_1 - u_2)^2 \quad (12)$$

$$T^* = \underset{0 \leq T \leq L-1}{\max} \sigma_B^2(T) \quad (13)$$

The Otsu method decides as a critical value the grey level maximizing the dispersion of the two groups obtained by formula (12). Figure 5 shows binarization that the Otsu critical value is applied to the input image and the image with a smoothed histogram

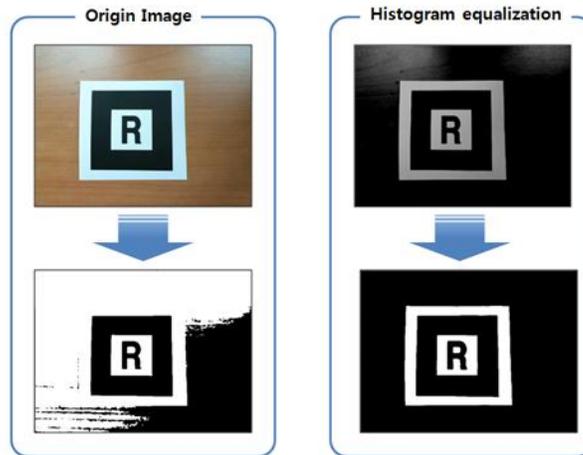


Figure 5. Binarization of Original Image & Histogram equalization image

In Figure 5, we can see that the binary image without histogram smoothing makes difficult detection of a marker, for there is no clear division between the marker area and background area. However, histogram smoothing allows a marker to be easily distinguished.

3.2. Reference Marker Recognition

As depicted in Figure 6, a marker is composed of Crust, Mantle, and Marker ROI(Core + Symbol). Crust is the white edge of the marker and Mantle is used for recognizing the marker as in the case of a finder pattern of QR code. Marker ROI is composed of Core and Symbol. Core is a white area, a contrast color tone of Mantle. Symbol refers to an image of a unique pattern. In order to decide a possible marker area, Mantle (a black area) should be detected first.

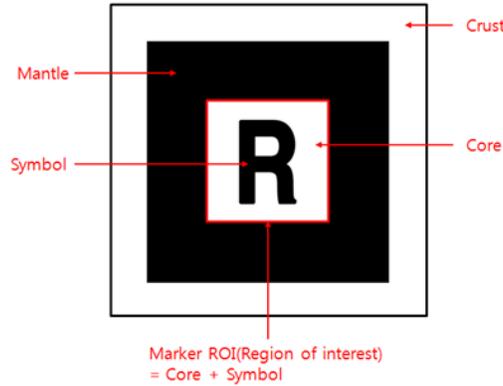


Figure 6. Structure of Marker

Therefore, the 1st marker area is extracted as in Figure 7 (B) by carrying out 1st labeling [14] in a binarized image. The first marker area extracted is reversed and 2nd labeling is executed so as to extract the final possible marker area as in Figure 7 (c). In order to improve computation speed, an area exceeding 90% or within 5% of the image should be excluded based on the length and width data.

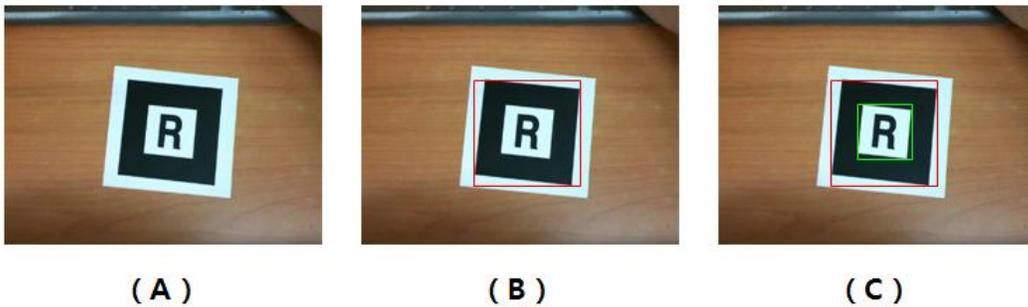


Figure 7. Marker candidate extraction (A) Original Image (B) Primary labeling (C) Secondary labeling

A corner point in the possible marker area, extracted for marker matching, is extracted as follows. First, if all corner points in/outside of the area are visible, contour data are extracted by using a boundary tracing algorithm. Then, one point of the data is selected, and a point farthest from the point is extracted based on the Pythagorean Theorem as depicted in Figure 8(B). Next, another corner point farthest from the extracted point is located as in Figure 8 (C). After the three points are extracted, formula (14) is used to locate a point that maximizes the area of the square as in Figure 8 (D). Finally, the four extracted points are employed to find out Marker ROI(Region of Interest).

$$\begin{aligned}
 \text{RectArea} &= \frac{1}{2}(x_1y_2 - x_2y_1 + x_2y_3 - x_3y_2 + \dots + x_ny_1 - x_1y_n) \\
 &= \frac{1}{2}x_1(y_2 - y_n) + x_2(y_3 - y_1) + \dots + x_n(y_1 - y_{n-1})
 \end{aligned} \tag{14}$$

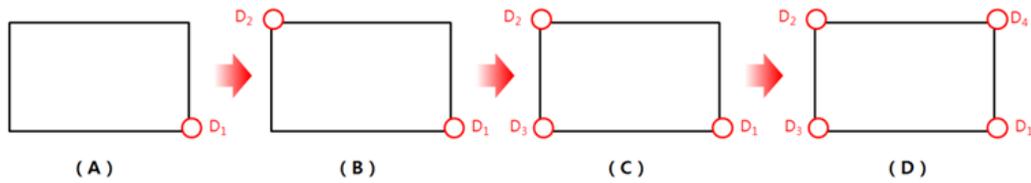


Figure 8. Corner Point Extraction in Marker Area

If point of straight-line is not recognized in marker area, the marker is not detected because corner point is not recognized. Progressive Probabilistic Hough Transform (PPHT)[15] can calculate the beginning and the end of each line as well as the direction of line. Therefore if detection of a marker fails because a corner point in a marker area is not detected, four lines with the highest straight-line intensity extracted using PPHT in the possible marker area as depicted in 7(C). An intersection point is located by applying formula (15), which is an equation for a straight line, to the four lines so as to extract a corner point and then Marker ROI.

$$C_y = \left(\frac{y_3 - y_1}{x_3 - x_1} \right) (C_x - x_1) + y_1 \quad (15)$$

Figure 9 shows the result of marker recognition using PPHT in the pre-treatment image while the red lines show an area detected.

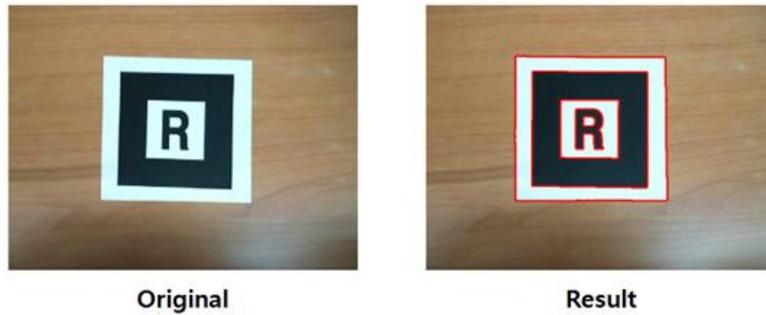


Figure 9. Marker detection using PPHT

3.3. Reference Marker Matching

There are two methods for marker matching: feature-based matching and template matching. Although feature-based matching is simpler than template matching, extraction of features is complicated and difficult to be implemented in a real-time environment. In this study, therefore, the template matching method is used [16].

Nevertheless, when matching is done by warping with registered markers based on the extracted corner points, there may occur central location and size deviation during marker matching. In order to solve this problem, therefore, markers are matched as follows in this study.

When the coordinates of the four corner points(D_1 - D_4) are (x_1, y_1) (x_2, y_2) (x_3, y_3) (x_4, y_4) formula (16) is used to detect the center of the marker.

$$C_y = \left(\frac{y_4 - y_2}{x_4 - x_2} \right) (C_x - x_2) + y_2 \quad (16)$$

The result of the calculation of formula (15) – formula (16) is formula (17), and is obtained by applying formula (17) as in formula (18). Again, this is applied to formula (15) or formula (16) so as to obtain C_y .

$$0 = \left(\frac{y_3 - y_1}{x_3 - x_1} - \frac{y_4 - y_2}{x_4 - x_2} \right) C_x - \left(\frac{y_3 - y_1}{x_3 - x_1} \right) x_1 + \left(\frac{y_4 - y_2}{x_4 - x_2} \right) x_2 + y_1 - y_2 \quad (17)$$

$$C_x = \frac{\left(\frac{y_3 - y_1}{x_3 - x_1} \right) x_1 - \left(\frac{y_4 - y_2}{x_4 - x_2} \right) x_2 - y_1 + y_2}{\frac{y_3 - y_1}{x_3 - x_1} - \frac{y_4 - y_2}{x_4 - x_2}} \quad (18)$$

When the height of a tag for a registered marker is h_r and that of the marker extracted after warping, the difference is calculated as formula (19).

$$S_{diff} = \frac{h_w}{h_r} \times 100 \quad (19)$$

When the center of a marker is a base point, correction of the size error (T_{total}), caused by warping, can be done by formula (20) and is a warping conversion matrix.

$$T_{total} = \begin{bmatrix} \frac{1}{S_{diff}} & 0 & 0 \\ 0 & \frac{1}{S_{diff}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot W_T \quad (20)$$

Figure 10 shows how to detect a reference marker by marker matching after correcting the size error of a marker and to obtain the coordinates of the corner points of the extracted marker.

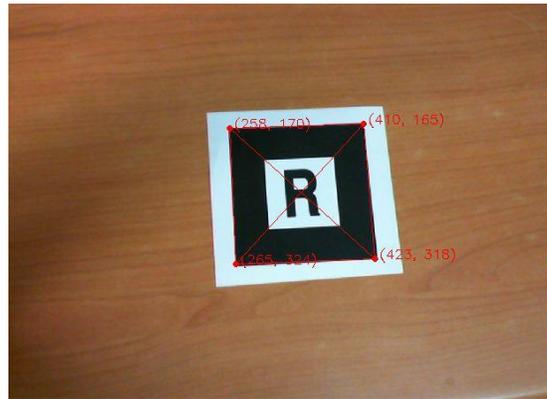


Figure 10. Marker Corner Coordinate Extraction

4. Marker Array List Method

Figure 11 shows the flow of sub marker detection process suggested in this study. Upon completion of reference marker detection, a marker array area is created based on the coordinates of the corner points (D_1 - D_4) in order to recognize a sub marker. In the marker array area a sub marker is detected by applying the method used for recognizing and matching the reference marker. After a sub marker is detected, a marker array is

created by using sub marker ID and marker array ID is verified. Then, an object is augmented by contents warping.

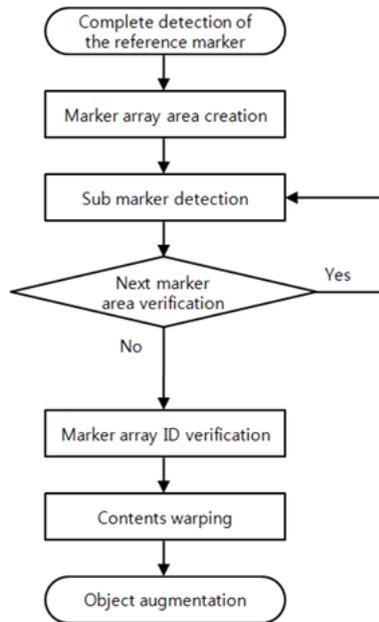


Figure 11. Flowchart of Marker Array List Method

4.1. Marker Array Area Creation

The marker array suggested in this study, as in Figure 12, recognizes the marker area divided into four sections in the form of the 2×2 matrix in which four markers are applied to each section and objects of $4^4=256$ can be augmented.

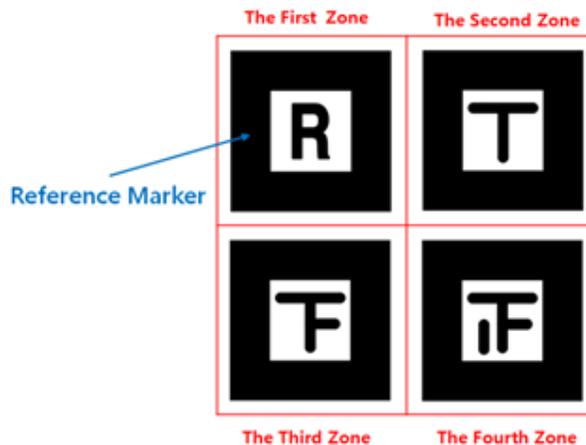


Figure 12. Marker Array Area

The marker array is recognized as follows. The length and width of a marker is measured by using the corner points (D_1, D_4) of the reference marker and then a marker array area, which is 10% bigger than the size, is selected. Next, the 2nd marker array

area is selected based on the left corner point (D_1, D_4) of the reference marker and the 3rd marker area, based on the right corner point (D_2, D_1). Later, the 4th marker array area is selected based on the lower part of the 2nd area and the right side of the 3rd area.

After the marker array area is selected, an area of interest is selected per section. Then, a marker is detected per section as in Figure 13 with the same method used for the reference marker detection. Sub marker ID is verified after the marker is detected.

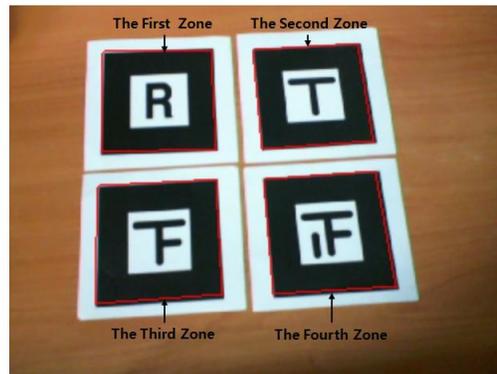


Figure 13. Marker Detection in Marker Array Area

4.2. Marker Array ID Verification

In order to verify marker array ID, one-dimension marker array should be created as shown in Figure 14. To do this, sub marker ID, detected by marker array (2×2 matrix) zones, is applied to the array in sequence.

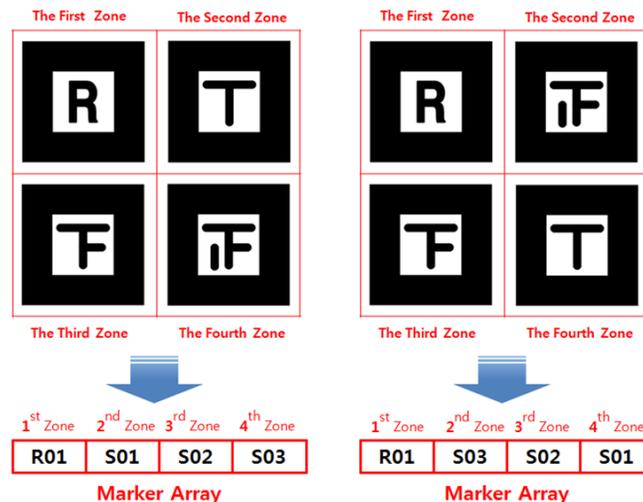


Figure 14. Conversion from marker array area to marker array

The ID of the marker array is verified by using the created marker array. And contents corresponding to the marker array ID are confirmed. Also, contents warping is done by creating a coordinates conversion matrix based on the coordinates of the corner points (of the reference marker), calculating a difference caused by warping, and

correcting the error. Figure 15 shows how an object is augmented according to marker array ID.

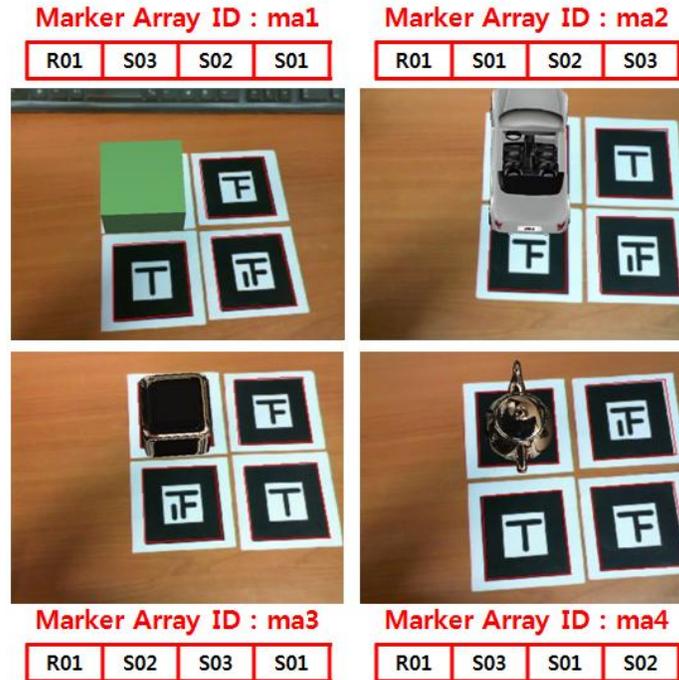


Figure 15. Content Augmentation Using Marker Array

5. Conclusion

In this study, we have explored a method of augmenting a large number of objects for smart home service AR without creating a multitude of markers but by using a marker array (created by a limited number of markers), which is a method of creating a list of a marker array.

In this study, we detected a reference marker by binarizing an input image, carrying out labeling for the converted image in order to select a possible marker area, recognizing a marker using a boundary tracing algorithm and PPHT, and conducting template matching. Then, a marker array area is selected based on the reference marker and a sub marker is detected within the area so as to verify marker array ID and augment an object. By doing so, we could augment a large number of objects with a limited number of markers and no similar patterns, which solves the problem of repeated detection of markers. Nevertheless, there should be further research on detection of multiple markers in a mobile device, small-size marker detection, and extension of a marker array area, for selection of marker array area can be difficult in such a device (e.g. a smart phone) so as to apply the method to marker-based AR services for various environments and devices.

Acknowledgements

This paper is supported by 2011 Korea Sanhak Foundation.

References

- [1] Yeol Lee, Dongwoo Seo, Gue Won Rhee, Sung Hun Hong, Ji-Seung Nam, "Virtual and Pervasive Smart Home Services Using Tangible Mixed Reality", Parallel and Distributed Processing with Applications, 2008. ISPA '08. International Symposium on
- [2] T. P. Caudell and D. W. Mizell, "Augmented reality: an application of heads-up display technology to manual manufacturing processes", Proc. IEEE Hawaii Int. Conf. on Systems Sciences, vol. 2, pp. 659-669, 1992.
- [3] S. Venugopal, A VLSI Architecture for Image Template Matching, M. Thesis, Dept. of Computer Science and Engineering, University of South Florida, 1994.
- [4] Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing, Prentice- Hall. Inc, Upper Saddle River. NJ, 2006
- [5] C. H. Chou and Y. C. Chen, "A VLSI Architecture for Real-Time and Flexible Image Template Matching", IEEE Trans on Circuits and Systems, vol. 36, no. 10, pp. 1336-1342, 1989.
- [6] A. Venot, J. F. Lebruchec and J. C. Roucayrol, "A New Class of Similarity Measure for Robust Image Registration", Computer. Vision ,Graph, and Image process, vol. 28, Issue. 2, pp. 176-184, 1984.
- [7] M. Svedlow, C. D. McGlem and P. E. Anuta, "Image Registration : Similarity Measure and Preprocessing Math Comparisons", IEEE Transactions on, Aerospace and Electronic Systems, vol, AES-14, Issue. 1, pp. 128-140, 1978 .
- [8] C. H. Chou and Y. C. Chen, "Moment Preserving Pattern Matching", Pattern Recognition, vol. 23, no. 5, pp. 461-474, 1990.
- [9] Duda, R.O, Hart, P.E., Pattern Recognition and Scene Analysis, Wiley: New York, NY, USA, 1973.
- [10] Boiangiu. C.-A., Raducanu, B., "Line detection techniques for automatic content conversion systems", WSEAS Transactions on Information Science and Applications, vol. 5, no. 7, pp. 1200-1209, 2008.
- [11] R. O. Duda, P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures", Communications of the ACM, vol. 15, no. 1, pp. 11-15, 1972.
- [12] Y. Wang, Q. Chen, and B. M. Zhang, "Image Enhancement based on Equal Area Dualistic sub-Image Histogram Equalization Method", IEEE Transactions on Consumer Electronics, vol.45, no.1, pp.68-75, 1999.
- [13] N. Otsu, "A threshold selection method from gray-level histograms", IEEE Transactions on systems, Man and Cybernetics, vol. 9, issue. 1, pp. 62-66, 1979.
- [14] He. Xuming, R.S. Zemel, M.A. "Multiscale Conditional Random Fields for Image Labeling", Carreira-Perpinan, Computer Vision and Pattern Recognition, Proceedings of the 2004 IEEE Computer Society Conference, vol. 2, pp. 696-702, 2004.
- [15] Sungmo Jung, Jae-gu Song, Dae-Joon Hwang, Jae Young Ahn, SeokSoo Kim, "A Study on Software-based Sensing Technology for Multiple Object Control in AR Video", sensors 2010, vol. 10, issue. 11, pp. 9857-9871, 2010
- [16] terwilliger, T. C, "Automated main-chain model building by template matching and iterative fragment extension", Acta Crystallographica Section D: Biological Crystallography, vol. 59, no. 1, pp. 38-44, 2002

Authors



Donghyun Kim is an undergraduate student taking up multimedia engineering from Hannam University, Korea. His research interests include Augmented reality, Information Security and Image processing.



Yohwan So received a B.S. degree in Fine Art from Hongik University, Korea, and M.S. degree in Fine Art from Hongik University, Korea, and M.S degree in Engineering from New York Institute of Technology. And currently, he joined the faculty of Hannam University, Korea. His research interests include multimedia contents, 3D modeling, VFX.



Seoksoo Kim received a B.S. degree in Computer Engineering from Kyungnam University, Korea, 1989 and M.S. degree in Information Engineering from Sungkyun-kwan University, Korea, 1991 and Ph.D. Degree in Information Engineering from Sungkyun-kwan University, Korea, 2002. In 2003, he joined the faculty of Hannam University,

Korea, where he is currently a professor in the Department of Multimedia Engineering. His Research interests include multimedia communication systems, distance learning, multimedia authoring, telemedicine, multimedia programming, computer networking, and information security. He is a member of KCA, KICS, KIMICS, KIPS, KMS, and DCS. He is editor-in-chief of IJMUE.