The Journal of Supercomputing (2022) 78:7509–7528
https://doi.org/10.1007/s11227-021-04161-0

A study on recognizing multi-real world object and estimating 3D position in augmented reality

Taemin Lee[1] · Changhun Jung[2] · Kyungtaek Lee[3] · Sanghyun Seo[4]

Accepted: 20 October 2021 / Published online: 15 November 2021
© The Author(s) 2021

Abstract

As augmented reality technologies develop, real-time interactions between objects present in the real world and virtual space are required. Generally, recognition and location estimation in augmented reality are carried out using tracking techniques, typically markers. However, using markers creates spatial constraints in simultaneous tracking of space and objects. Therefore, we propose a system that enables camera tracking in the real world and visualizes virtual visual information through the recognition and positioning of objects. We scanned the space using an RGB-D camera. A three-dimensional (3D) dense point cloud map is created using point clouds generated through video images. Among the generated point cloud information, objects are detected and retrieved based on the pre-learned data. Finally, using the predicted pose of the detected objects, other information may be augmented. Our system estimates object recognition and 3D pose based on simple camera information, enabling the viewing of virtual visual information based on object location.

Keywords Augmented reality · Object recognition · SLAM · Estimating 3D position · Camera tracking

✉ Sanghyun Seo
sanghyun@cau.ac.kr

Taemin Lee
kevinlee@cglab.cau.ac.kr

Changhun Jung
grchjung@gmail.com

Kyungtaek Lee
ktechlee@keti.re.kr

[1] Da Vinci Software Education Institute, Chung-Ang University, Seoul 06974, Korea

[2] Entertainment Technology, Art & Technology Chung-Ang University, Seoul 06974, Korea

[3] Korea Electronics Technology Institute, Seongnam-si, Gyeonggi-do 13509, Korea

[4] School of Computer Art, College of Art and Technology, Chung-Ang University, Anseong-si, Gyeonggi-do 17546, Korea

1 Introduction

Augmented reality (AR) combines virtual and real worlds, allowing users to experience virtual objects in real space. AR is different from virtual reality, which allows users to immerse themselves in a new space by building an artificial world. AR complements the real world and helps users' understanding by implementing virtual interfaces based on real world. Hence, it has recently been used in various industries such as healthcare, education, gaming, and entertainment [1, 2].

As the usability of augmented reality technologies improves, changes in interface expression methods and interaction functions according to user input are required. Display, tracking, and video processing technologies are used in augmented reality. Among them, tracking technology is crucial to the performance and functionality of applications as it identifies the point-in-time position and maintains a specific state through recognition. Therefore, many studies have been conducted to achieve high completeness.

Tracking techniques in augmented reality are generally based on markers, as shown in Fig. 1. When various types of markers are recognized, virtual visual information(interface) is augmented based on their location [3, 4]. Traditional tracking techniques visualize information based on markers, resulting in restrictions on the working space. To overcome this limitation, it is necessary to recognize objects that exist inside AR based on space; this is how spatial markers are used. However, when using spatial markers, the control unit is not an object; thus, it is not possible to interact with or control the change in the working space. In other words, depending on the type of marker, the challenges are to overcome the constraints of space and gain control of each object. Therefore, this study proposes a system that enables tracking of cameras in a real-world space, recognition of objects in real time, and visualization of virtual visual information through location estimation.

To realize the proposed system, point cloud data are generated based on the color and depth image information obtained using the camera. The generated point cloud data are used to construct a space through tracking. Using SLAM [5], the locations of objects in space are estimated based on the point cloud data. We propose a method to eliminate the gaps that arise in the characteristic points. The method is also applicable in recognizing objects and estimating three-dimensional (3D) poses

Fig. 1 Augmented reality using marker (left is binary marker and right is image marker)

in the video stream of AR execution process. We propose a 3D pose estimation method that recognizes two-dimensional (2D) objects using deep learning and utilizes the recognized 2D positional information to minimize errors.

The contributions of our research are as follows. First, the space can be detected through real-world camera tracking, which can create a space map using an RGB-D camera without using special equipment. Second, objects can be recognized based on the generated space and the estimated camera positions. In this study, real time was ensured by applying multithread and fast deep learning algorithms. Third, the camera point of view was corrected by extending the recognized 2D object model to three dimensions. Finally, the 3D extension of object information was augmented via AR, allowing virtual visual information to be visualized according to the actual object location.

2 Related work

2.1 Object recognition and location estimation techniques

Studies that focus on object recognition use various techniques for estimating locations [6–8]. The method for recognizing objects from an input video or image is largely divided into computer vision-based and deep learning-based methods. Computer vision-based recognition detects features in an image and determines object. Early object recognition studies mainly used computer vision methods. Scale invariant feature transform(SIFT) [9], speeded up robust features(SURF) [10], and oriented FAST and rotated BRIEF(ORB) [11] are the commonly used algorithms. Feature detection algorithms have different underlying techniques but detection occurs in the geometric feature part of the recognition object. Because it is performed pixel-by-pixel, it can be applied to estimate location information while finding matching feature points.

In deep learning-based recognition, object recognition is performed through neural network learning. Early neural networks were used only for object recognition, followed by R-CNN [12] and spatial pooling in deep convolutional networks(SPPnet) [13], and single shot multibox detector(SSD) [14], which estimate object recognition and 2D locations. Because the location information of objects in real space has three dimensions, studies have been conducted to obtain vision-based 3D location information [15–17], or deep learning-based 3D location information [18, 19]. Previous studies have the disadvantages of not recognizing objects, or not having sufficient accuracy in positioning. Deep learning-based studies have yielded high performance in terms of accuracy and object recognition. In addition, they either require many datasets for learning or are difficult to operate in large spaces. Our study proposes a system that can operate in a large real space, estimate 3D locations, and recognize objects with high accuracy even with small datasets.

2.2 Implementing augmented reality

Augmented reality was first reported in 1992 [20] and has since been studied together with computer vision technology. Subsequent studies have mainly focused on using markers to achieve accurate positioning and tracking. Marker forms include binaries, images, objects, and spaces. Initially, the binary markers were the main form of markers used. This was because it was easy to construct multiple shapes and extract feature points using binary markers [21]. However, when placing markers in the real world, they are naturally connected to cover other objects without being seen. As image and object markers, objects that exist in the real world are used. Therefore, there are few cases of visual alienation or interference. However, feature extraction and throughput for marker recognition for image and object markers must be relatively large compared to binary markers. Consequently, they were not applied in early tracking technologies. A research showed that lighter systems with high processing speeds were possible in real time but resulted in high usage [22]. AR using binary, image, or object markers is not suitable for AR applications in large spaces because augmented reality is possible only within a limited space where the camera recognizes the marker.

In augmented reality, space can be used as a marker, for example, in GPS-based cases such as [23] and Pokemon GO games. However, it is difficult to expect a natural synthesis of virtual objects with real world because detailed location calculations or operations are not possible indoors. Therefore, the use of SLAM technology for augmented reality has been proposed [24]. If the space is used as a marker, it is impossible to estimate the recognition and location of objects in the real world. Recently, studies have been conducted to simultaneously recognize space and objects [25, 26]. Our system can construct real-world spaces and objects in three dimensions using AR technology and can be used as a space-based marker to implement AR.

3 Proposed method

3.1 System overview

In this study, we propose a method for visualizing virtual information in a real-world location by recognizing objects and estimating their location in a 3D coordinate system while simultaneously tracking the camera. The operating stages of the system consist of image input, tracking, 2D object detection, 3D pose estimation, and visualization. The system receives color and depth image information through the camera and uses it to generate the point cloud data. The SLAM-based tracking system estimates the location of cameras in the real world in real time and configures the entered data in a map form. Before performing deep learning-based object detection, we built the datasets and performed neural network learning. The map created via the point cloud data in the last step, 2D spatial location obtained through object detection, and object

point cloud were input into the ICP algorithm to estimate the 3D position and posture of objects in the real world. The estimated location was augmented using virtual visual information through graphics technology. Figure 2 shows the conceptual flow of the system proposed in this study.

3.2 Date collection

A definition of space is needed to estimate the location of users in the real world and track them. There is a method to obtain satellite signals and locate the users using GPS sensors without defining the space. However, this method cannot be used indoors and errors occur below meters. Additional equipment, such as a dead recoding module, geographic information system, inertial estimation equipment, and cameras, was needed to compensate for this. For spatial definitions, the characteristics must be identifiable. The main target of the features is visual information, and the image is used as a medium. In this study, location estimation and tracking are performed indoors and outdoors using RGB-D cameras as input devices without complex equipment. When color and depth video are input through the RGB-D camera, a point cloud is created. We conducted the study assuming that objects were separated from each other in the scanned space. This was because when objects overlap, estimating the pose was problematic. This issue is discussed at the end. Figure 3 shows an input image from the camera and an example of a generated point cloud.

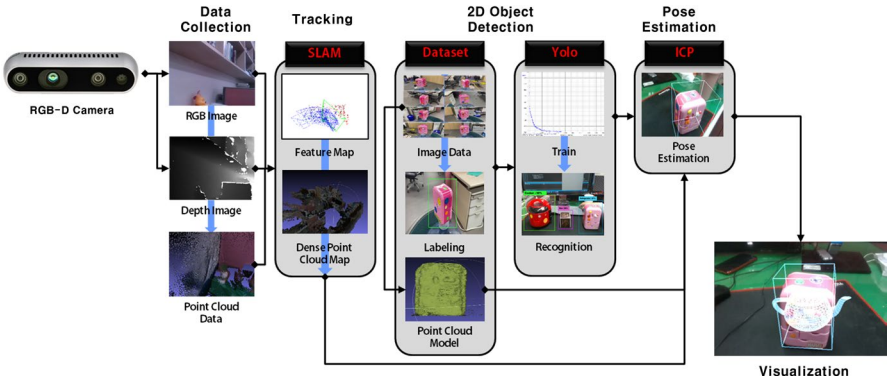


Fig. 2 System overview (First, we retrieve color and depth image data using the RGB-D camera. Second, a dense point cloud map is generated using the feature map and spare point cloud. Using the dense point cloud map, 2D objects are detected. The detection is performed by Yolo algorithm. Finally, we estimate the pose of an object for augmented reality visualization)

Fig. 3 Example of input data and point cloud (left: RGB image from the real world, middle: depth image from the real world, right: generated point cloud)

3.3 Tracking

A definition of space is needed to estimate the location of users in the real world and track them. The algorithms used to select suitable AR construction systems must satisfy the following conditions:

- Real-time operation in a light system environment
- Definition of spatial markers indoors and outdoors
- Prevent performance degradation as maps increase
- Fast return if location estimation and tracking are stopped

Tracking was carried out using ORB-SLAM [5], which satisfied the above conditions. Although there are various SLAM techniques [5, 27–29], we used ORB-SLAM in our system because it is the most satisfactory with respect to our system specifications. ORB-SLAM2 is based on an ORB feature extraction algorithm. It can establish a light system environment and enable real-time operation. Safe performance is ensured through the deletion of redundant points

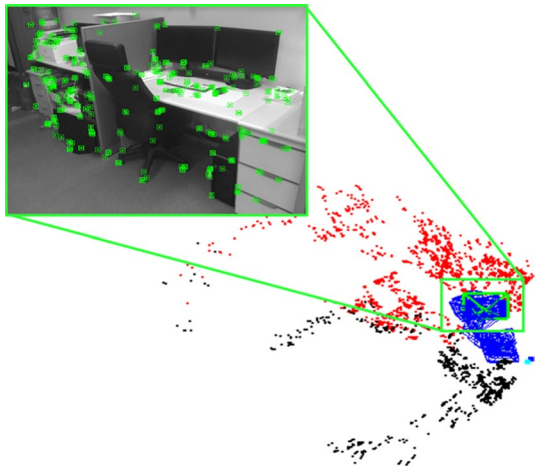


Fig. 4 Result of tracking based on RGB-D camera (This is extracted sparse point cloud using ORB-SLAM2)

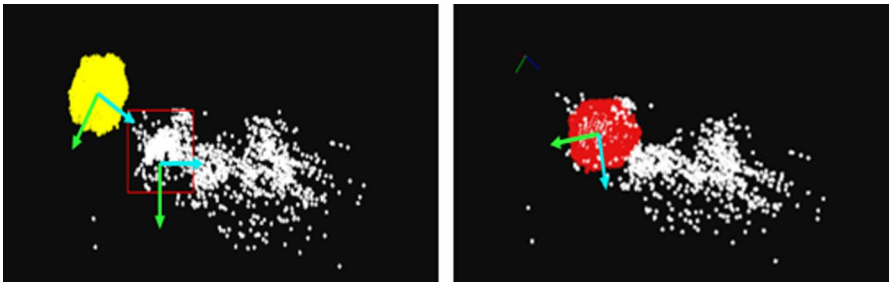


Fig. 5 Result of generating pose estimation with spare point cloud (left: before pose estimation, right: after pose estimation)

and loop closing in indoor and outdoor environments. Using the bag-of-words technique, it is also possible to quickly return to the positioning process at a tracking stop instant. Figure 4 shows the results of map creation and tracking based on the information input through the RGB-D camera.

Because the distribution of the point cloud is very sparse, the resulting map in Fig. 4 does not cause any problems in tracking. However, the map does not work when estimating the pose of an object later. This can be seen in Fig. 5. The object coordinates on the left side of Fig. 5 differ by 45° from the map coordinates. After estimating the pose, it is evident that the positional movement has been performed normally, as shown on the right side of Fig. 5, but the rotation shows a difference of approximately 90°.

To solve this problem, we used the point library to create a dense point cloud. Figure 6a shows the results of a single frame. This information generates maps via SLAM. At this point, it is necessary to locate and synthesize each frame. In general, point matching algorithms, such as ICP, are used to synthesize different point cloud data. However, there are difficulties in real-time operation owing to the need for a long performance time. The optimization of duplicate points is also required. Therefore, this study uses ORB-SLAM to omit the processing of the frame position and redundant points. The synthesis is performed through feature maps and the dense point cloud matching method, and can be completed in real time. Figure 6b shows the process and result of creating a dense point cloud map.

3.4 2D object detection

Based on ORB-SLAM, feature points were extracted from the images, and maps were generated. Because space was defined, tracking and camera positioning were possible. However, it was impossible to recognize objects and estimate their locations in three dimensions. This was because objects were included in space, but the computer system could understand only a geometric form of space. Therefore, additional systems are needed to define the characteristics of the objects and estimate their recognition and location. This study detects

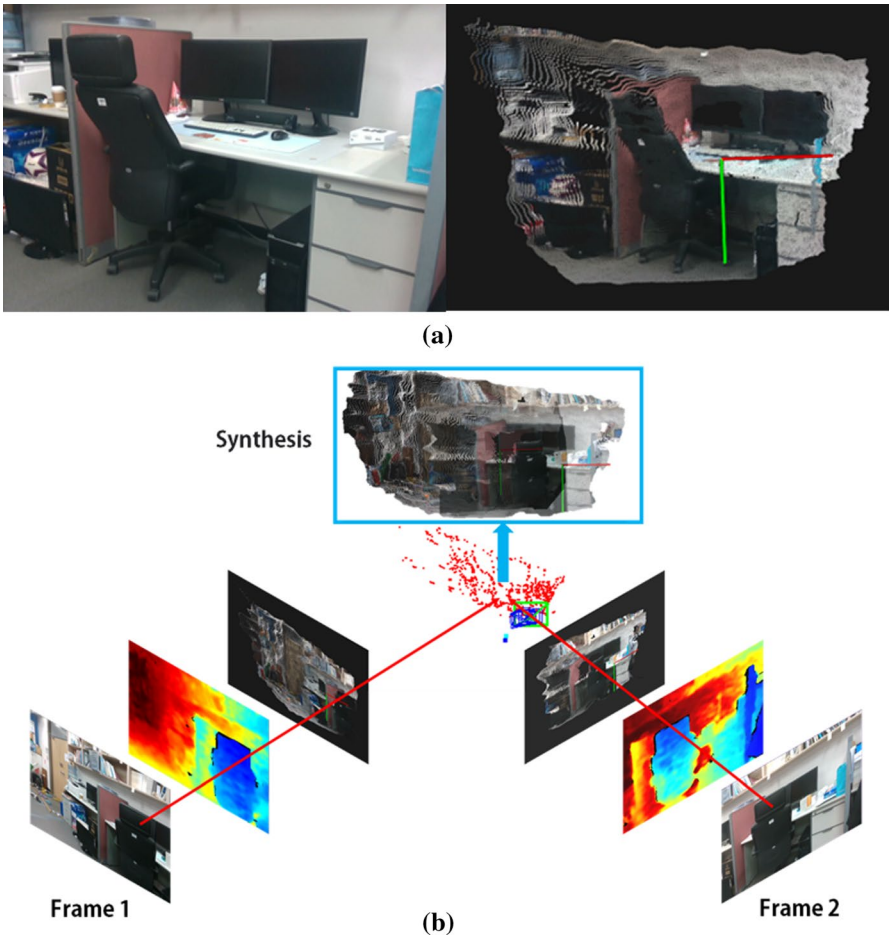


Fig. 6 Process and result of dense point cloud (**a** result of dense point cloud in single frame, **b** process of generating dense point cloud)

Table 1 Compare throughput and mean average precision (mAP) values by neural network

Network model	mAP	Time (ms)
SSD	28.0	61
Yolov3	28.2	22
Faster R-CNN	29.9	85

objects (perception and location estimation) based on deep learning. The data required for neural network learning are constructed through semi-auto labeling and data inflation. The detection results are passed to the tracking system to visualize the class of objects and results of the 3D position estimation are passed to the user. Of the various neural networks that can recognize objects,

we used the Yolov3 [30] model because its recognition is sufficiently accurate, lightweight, and fast in terms of system usage. As Table 1 shows, its object recognition accuracy object is not significantly different from that of other neural networks, yet the execution is very fast.

To reduce the dataset construction time, the initial data are acquired through frame segmentation in the image, and several datasets are constructed via data inflation through growing, color change, rotation, synthesis, and partial deletion. For each generated image, a data label is assigned. The composition of the data label consists of x-and y-coordinates, width, and height. The input of these values must be provided by the user. Therefore, to improve data labeling, we developed a semi-auto labeling system based on SURF feature point matching. Figure 7 shows the structure of the semi-auto labeling system.

The bounding box entered is automatically transformed into a labeling coordinate configuration. Subsequently, the process of automatically determining the bounding box is repeated until the end by matching the generated source image with the next frame image. Learning was conducted using datasets built through data inflation and semi-auto labeling on the Yolo neural network. There are three classes of objects learned with a batch size of 64 and the number of learnings of 2700. Figure 8 shows the accuracy results of detecting objects through a learned neural network.

3.5 3D pose estimation

In this study, the proposed method estimates the position of the camera and detects the objects simultaneously. Estimating an object using the results discussed in Sects. 3 and 4 causes the problem of augmenting virtual information inconsistently with the object, as shown in Fig. 9. This is because the location estimation information is

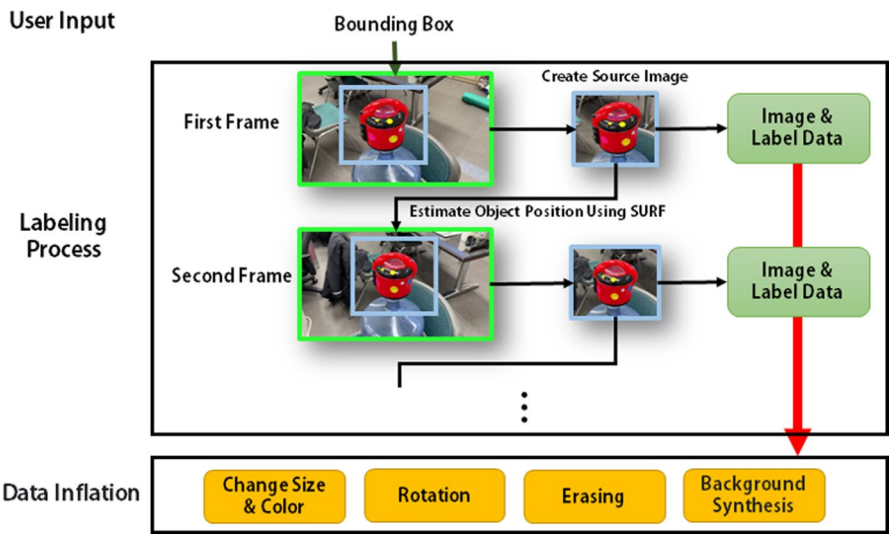


Fig. 7 System of semi-auto labeling

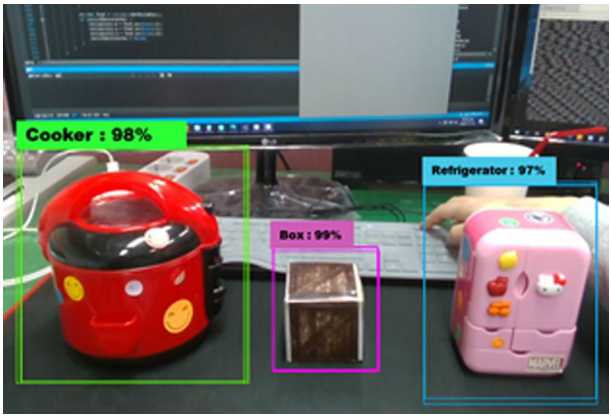


Fig. 8 Result of object recognition (cooker: 98%, Box: 99%, Refrigerator: 97%)

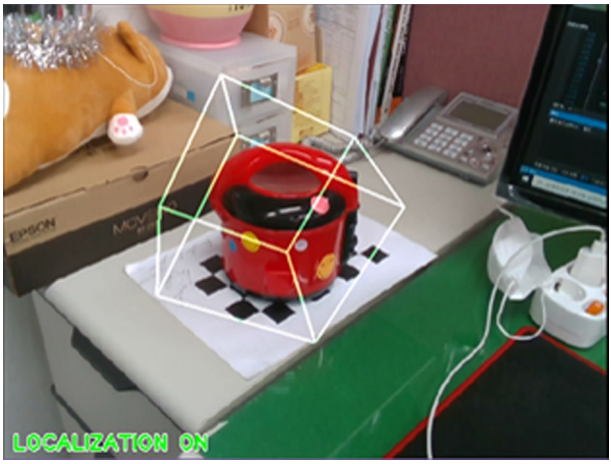


Fig. 9 Result of pose estimation and augmentation using 2D information

two dimensional, that is, it is stereoscopic and cannot be expressed according to the direction of rotation. Therefore, the ICP algorithm [31] was used to calibrate the 3D pose.

The ICP algorithm outputs the difference in position and angle between the two models in a matrix through the repeated matching between close points. Because of repeated matching between points, the data entered is in point cloud format. It uses a point cloud of objects created through a dense point cloud map and model reconstruction. Figure 10 shows the results of the point cloud creation with objects reconstructed using the Meshroom program.

Because ICP includes both position and rotation information, it can proceed immediately without two-dimensional information. However, using ICP alone can

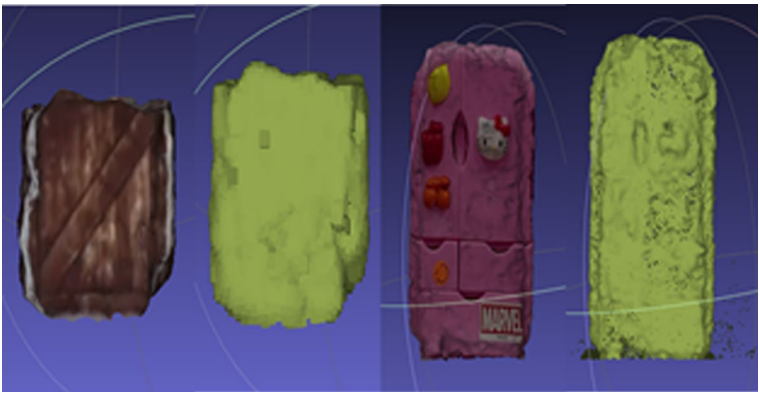


Fig. 10 Result of generating point cloud of objects

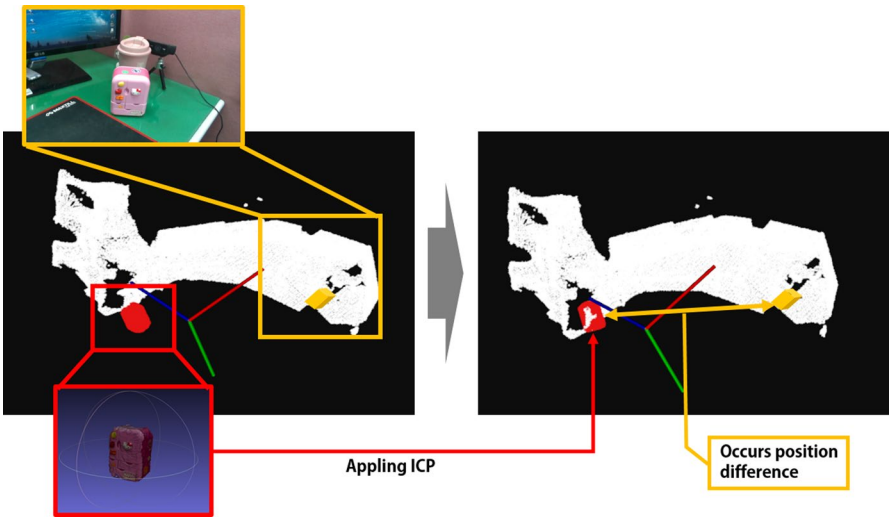


Fig. 11 Problems with ICP algorithm application after omitting 2D object detection steps

cause errors in estimating an object's pose. This is because ICP is performed with a point cloud of maps and objects rather than matching the same data. Figure 11 illustrates the problems that arise when the two-dimensional object detection step is omitted while estimating object pose.

The image on the left in Fig. 11 illustrates the performance of the ICP algorithm. The real world is represented by the point cloud map data, and the actual object location is shown in yellow. The point cloud data in red represent the reconstructed objects. The right side of Fig. 11 shows the matching results. Note that there is a difference in position as the algorithm is performed with adjacent points. Figure 12 shows the results of ICP and region limiting in point cloud map data through

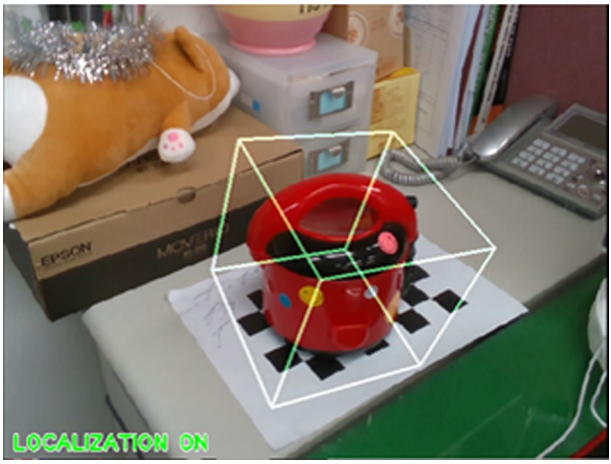


Fig. 12 Results of 3D pose estimation

two-dimensional location detection. Comparing these results with Fig. 9, we can see that the virtual information is augmented to match the real object.

4 Experimental result

This study proposes a system that enables tracking of cameras inside a real-world space and visualizes virtual visual information through recognition and location estimation of objects. The system was implemented in the environment described in Table 2.

To fulfill real-time requirements, tracking, two-dimensional object detection, and three-dimensional pose estimation steps were handled in different threads. Figure 13 briefly illustrates the thread-motion structure. The three threads shown in Fig. 13 operate simultaneously. The tracking thread was maintained, while the overall system was not stationary. Two-dimensional object detection and three-dimensional pose estimation can result in state changes upon user request.

Table 2 System environment

Contents	Environment
CPU	Intel Core i7-9700
RAM	32GB
GPU	Geforce RTX 2060 super
Language	C++
Map generation	ORB-SLAM
Object Recognition	Yolov3
Pose Calibration	Go-ICP

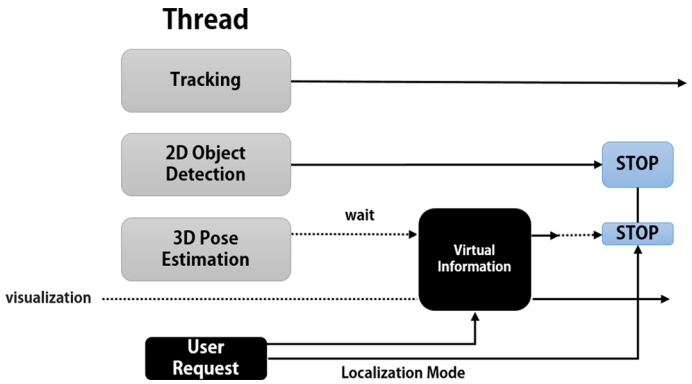


Fig. 13 Structure of thread operation

We conducted camera tracking, object recognition, and positioning in a 3 x 3 m2 area. Based on a spatial map composed of 500,000 dense point clouds, the object was found, and its location was estimated to be augmented.

Figure 14 shows the results of the synthesis of the real world and virtual visual information. The recognition results of each object are displayed with different colors of virtual visual information. Each object can be recognized and virtual visual information in different colors can be synthesized into the real world. When the three-dimensional bounding box was checked for each result, we observed that it accurately surrounded the object according to its rotation and position. The direction and position of the teapot indicate that accurate tracking is performed even when the camera moves. The result is that mapping is performed, and tracking of virtual visual information works normally even in localization mode. In addition, virtual visual information can be augmented on objects and tracking can be checked, as shown in Fig. 15, which operates normally even in a large area.

Figure 16 (end of paper) illustrates a comparison of the commercialized SDKs. Fig. 16a and b depicts marker-based augmented reality, where tracking stops when a marker is invisible. Our research shows that virtual visual information is maintained owing to the possibility of spatial tracking. Figure 16c and d illustrates the comparison between the results of constructing a map based on SLAM and enhancing virtual visual information on two different systems. As objects are added to space, traditional SDKs (MAXST SLAMs) are required to construct new maps to perform object recognition and position virtual visual information input by users. Because the input is performed directly on the development engine, the accurate placement is difficult. Our results show that the system performs object recognition and location estimation even when an object is added from the same map data. In addition, when creating maps with MAXST, there were difficulties in normal tracking behavior owing to the lack of environmental factors with characteristics.

The proposed system can synthesize virtual visual information at the exact object location in real time and perform normal tracking according to the camera location in the real world. However, owing to high dependence on visual information, it is

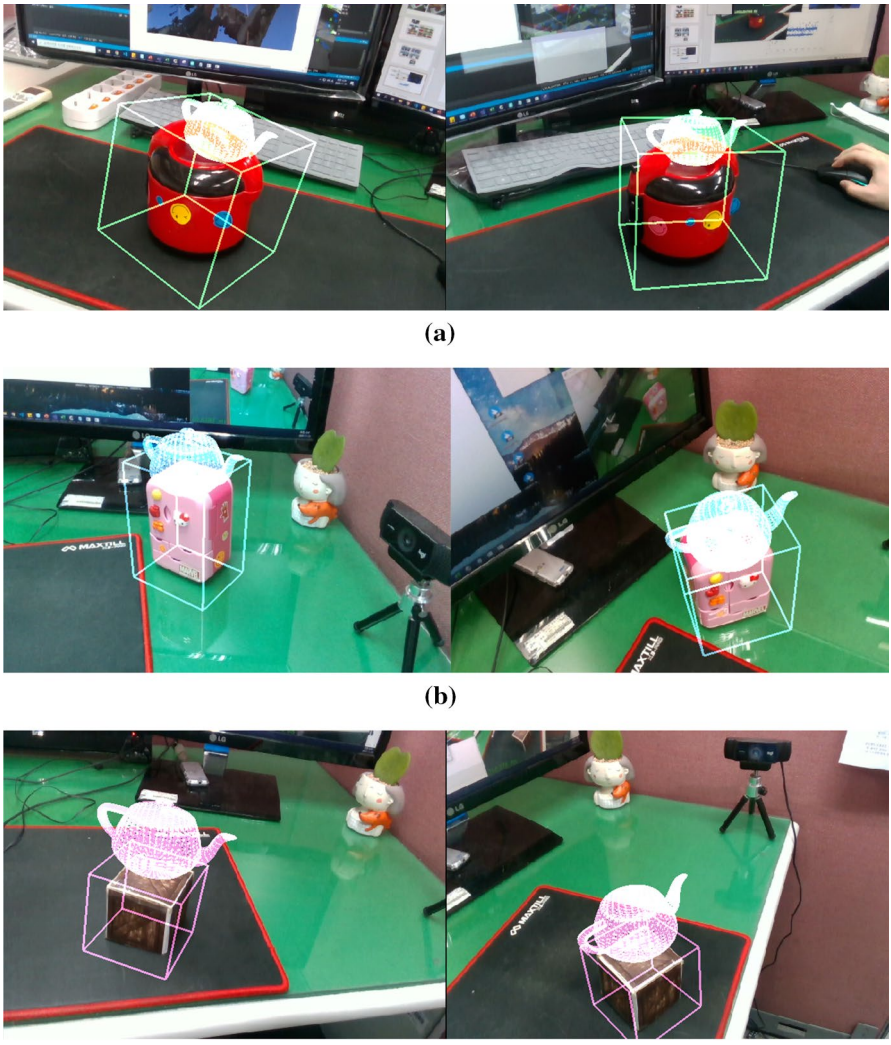


Fig. 14 Result of augmented visual information based on various objects (**a** cooker, **b** refrigerator, **c** box. As the 3D bounding box can be confirmed for each result, it accurately surrounds the object according to its rotation and position. Note that the direction and position of the kettle are accurately tracked and augmented even when the camera moves. The system operates normally even in localization mode)

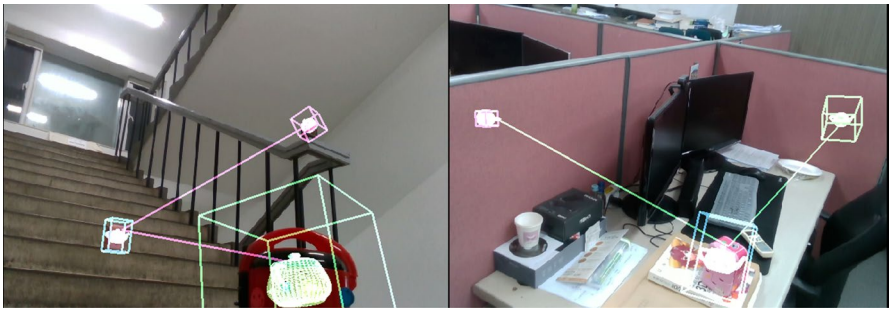


Fig. 15 Result of tracking in a large area (Note that the enhancement is maintained in a large room rather than a limited space. If an object is already augmented and tracked, tracking will continue even if the middle is interrupted by partitions or stairs)

difficult to operate properly with many objects or occlusions. In particular, ICPs that operate on point-cloud data are heavily affected. The ICP is responsible for correcting the virtual visual information to be synthesized with respect to the object location. Therefore, it is difficult to synthesize objects in the correct position if they are obscured or attached to other objects. Figure 17 shows the results of location estimation failure due to occlusion.

5 Conclusion

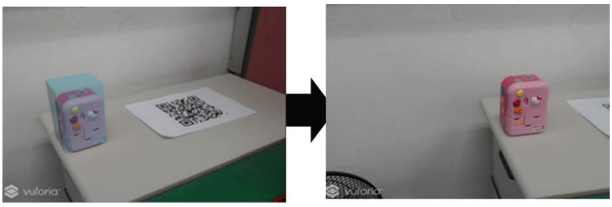
In this study, we propose a system that enables camera tracking in the real world and visualizes virtual visual information through object recognition estimation. Our system could augment a space through camera tracking, two-dimensional object recognition, and three-dimensional pose estimation based on RGB-D camera information, while recognizing three-dimensional positional information and objects. A SLAM-based camera tracking technology was used because the two coordinates need to be shared for the interaction between reality and virtual space. The minimum unit of control in the space is the object. However, it is impossible to recognize and estimate the location of objects if a spatial map is created based on SLAM. This issue was resolved using deep learning-based object recognition. Finally, the ICP algorithm augments the virtual visual information to match the position and rotation direction of the real object.

The contributions of our study are as follows: First, a given space can be scanned using simple camera equipment without special equipment. Usually, to reconstruct a

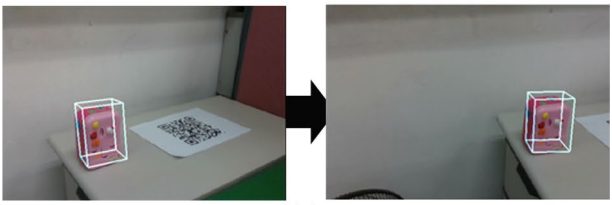
Fig. 16 Comparison with other sdk (**a**, **b** comparison of object augmentation according to camera conversion—the Vuforia SDK does not become an augmented-blue box if the camera rotates and the marker is not recognized. Our work, on the other hand, maintains an augmented-white box because it is spatial marker-based and therefore augmented on object-based basis without markers. **c**, **d** comparison of augmentation based on map configuration If an object is added to the same space, a new map must be constructed and repositioned, but our results do not need to be) ▶

three-dimensional space, special equipment or specialized tools must be used. However, in our study, a three-dimensional space was configured by photographing video images using a simple RGB-D camera. Second, object recognition can be performed using the created point cloud. Based on the point cloud data collected from the video image, not only space but also objects existing in it can be reconstructed. Moreover, objects can be recognized in a short time. Third, it is possible to determine the 3D poses of the localized objects and detect the state in which they are placed. This helps to augment other virtual objects.

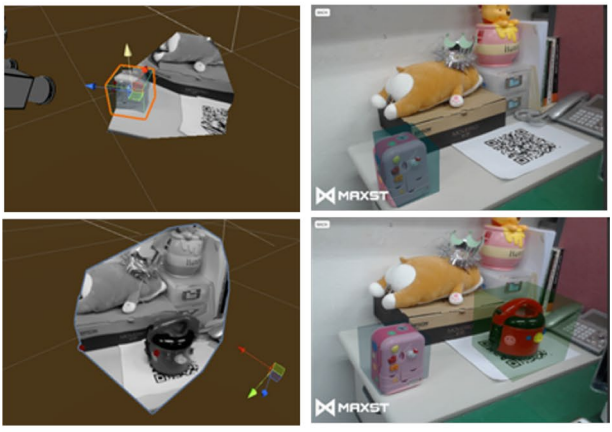
The proposed system can become the underlying technology that enables real-time interaction between the real world and objects present in virtual space. It is expected that AR will be available through convergence with Internet of Things (IoT) if problems arising from location estimation or occlusion of dynamic objects, which exist as complementary points to our research, are resolved. AR technology enables users to control and manage objects through virtual visual information in the real world. It is expected to provide users with a sense of immersion and realism as an operational method through an intuitive interface. Sensory-related benefits and advances in tracking technology can increase the likelihood of application in other fields, such as artificial intelligence, healthcare, education, gaming, military, and entertainment. As a simple example, it is



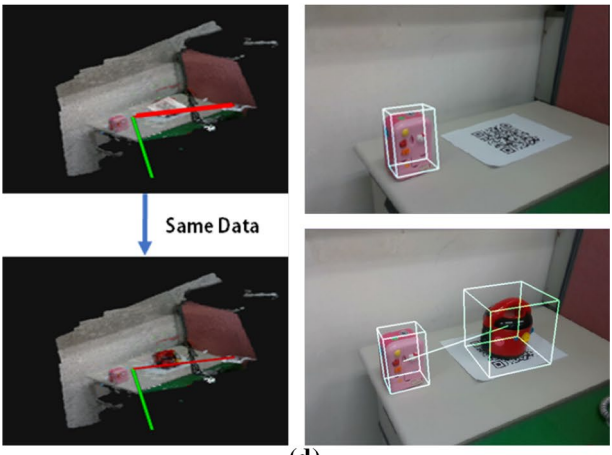
(a)



(b)



(c)



(d)

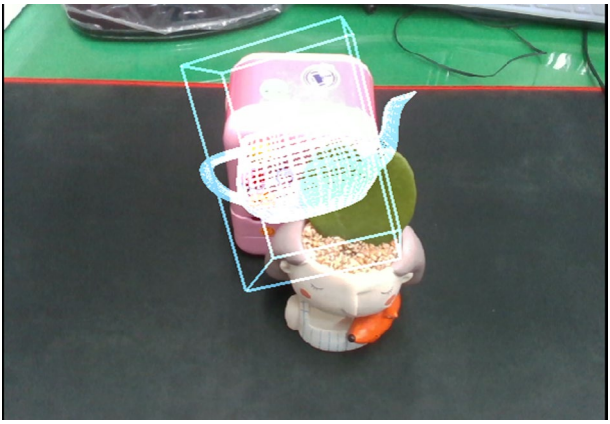


Fig. 17 Failure to estimate location due to occlusion (This is an error that occurs in point-to-point matching of ICP. When two objects are overlapped, attempting to match the object in front yields an unexpected enhancement for improved clarity)

expected that datasets that are necessary when learning 3D objects in artificial intelligence can be established using AR tracking technology.

Acknowledgements This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. IITP2020000103001, Development of 5G-based 3D spatial scanning device technology for virtual space composition) and supported by the Chung-Ang University Research Scholarship Grants in 2021.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Azuma RT (1997) A survey of augmented reality. Presence Teleoper Virt Environ 6(4):355–385
2. Coltekin A, Lochhead I, Madden M, Christophe S, Devaux A, Pettit C, Kubicek P (2020) Extended reality in spatial sciences: a review of research challenges and future directions. ISPRS Int J Geo-Inform 9(7):439
3. Keisuke T, Itaru K, Yuichi O (2007) Nested marker for augmented reality. In: 2007 IEEE Virtual Reality Conference. https://doi.org/10.1109/VR.2007.352495
4. Anuroop K, Karan K, Chetan G (2015) Marker based augmented reality. Adv Comput Sci Inform Technol 2(5):441–445
5. Mur-Artal R, Tardos JD (2017) Orb-slam2: an open-source slam system for monocular, stereo, and rgb-d cameras. IEEE Trans Robot 33(5):1255–1262
6. Ze Y, Liwie W (2019) Learning relationships for multi-view 3D object recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 7505–7514

7. Ning W, Yuanyuan W, Meng J (2020) Review on deep learning techniques for marine object recognition: architectures and algorithms. Control Eng Pract 104458:104458. https://doi.org/10.1016/j.conengprac.2020.104458
8. Jingru T, Cahngbao W, Buyu L, Quanguan L, Wanli O, Changqing Y, Junjie Y (2020) Equalization loss for long-tailed object recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11662–11671
9. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int j Comput Vis 60(2):91–110
10. Bay H, Tuytelaars T, Van Gool L (2006) Surf: speeded up robust features. In: European Conference on Computer Vision. Springer, Berlin, Heidelberg, pp 404–417
11. Rublee E, Rabaud V, Konolige K, Bradski G (2011) ORB: an efficient alternative to SIFT or SURF. In: 2011 International Conference on Computer Vision, pp 2564–2571
12. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 580–587
13. He K, Zhang X, Ren S, Sun J (2015) Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans Patt Anal Mach Intell 37(9):1904–1916
14. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, Berg AC (2016) Ssd: single shot multibox detector. In: European Conference on Computer Vision, pp 21–37
15. Rothganger F, Lazebnik S, Schmid C, Ponce J (2003) 3D object modeling and recognition using affine-invariant patches and multi-view spatial constraints. In: 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol 2, pp 272–277
16. Ozyesil O, Voroninski V, Basri R, Singer A (2017) A survey of structure from motion. arXiv preprint arXiv:1701.08493
17. Xiao J, Russell B, Torralba A (2012) Localizing 3D cuboids in single-view images. Adv Neural Inform Process Syst, pp 746–754
18. Pavlakos G, Zhou X, Chan A, Derpanis K.G, Daniilidis K (2017) 6-dof object pose from semantic keypoints. In: 2017 IEEE International Conference on Robotics and Automation (ICRA), 2011–2018
19. Peng S, Liu Y, Huang Q, Zhou X, Bao H (2019) Pvnet: pixel-wise voting network for 6dof pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 4561–4570
20. Thomas PC, David WM (1992) Augmented reality: an application of heads-up display technology to manual manufacturing processes. In: Hawaii International Conference on System Sciences, pp 659–669
21. Kalkusch M, Lidy T, Knapp N, Reitmayr G, Kaufmann H, Schmalstieg D (2002) Structured visual markers for indoor pathfinding. In: The First IEEE international workshop agumented reality toolkit, pp 1–8
22. Wagner D, Reitmayr G, Mulloni A, Drummond T, Schmalstieg D (2008) Pose tracking from natural features on mobile phones. In: 2008 7th IEEE/ACM international symposium on mixed and augmented reality, pp 125–134
23. Heok AD , Fong SW, Goh KH, Yang X, Liu W, Farzbiz F (2003) Human Pacman: a sensing-based mobile entertainment system with ubiquitous computing and tangible interaction. In: Proceedings of the 2nd workshop on network and system support for games, pp 106–117
24. Munoz-Montoya F, Juan MC, Mendez-Lopez M, Fidalgo C (2018) Augmented reality based on SLAM to assess spatial short-term memory. IEEE Access 7:2453–2466
25. Runz M, Buffier M, Agapito L (2018) Maskfusion: real-time recognition, tracking and reconstruction of multiple moving objects. In: 2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp 10–20
26. Seo S.H, Kang D.W, Park S.O (2018) Real-time adaptable and coherent rendering for outdoor augmented reality. EURASIP J Image Video Process 118
27. Linyan C, Chaowei M (2019) SOF-SLAM: a semantic visual SLAM for dynamic environments. IEEE Access 7:166528–166539
28. Chao Y, Zuxin L, Xin-Jun L, Fugui X, Yi Y, Qi W, Qiao F (2018) DS-SLAM: a semantic visual SLAM towards dynamic environments. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems. https://doi.org/10.1109/IROS.2018.8593691
29. Shinya S, Mikiya S, Ken S (2019) Openvslam: a versatile visual slam framework. In: Proceedings of the 27th ACM International Conference on Multimedia, pp 2292–2295

30. Redmon J, Farhadi A (2018) Yolov3: an incremental improvement. arXiv preprint arXiv:1804.02767
31. Yang J, Li H, Campbell D, Jia Y (2015) Go-ICP: a globally optimal solution to 3D ICP point-set registration. IEEE Trans Patt Anal Mach intelligence 38(11):2241–2254

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.