Augmented reality landscape simulation system with occlusion handling that allows free movement by a small drone

Improved hardware dependency

○ Naoki Kikuchi*1, Tomohiro Fukuda*2 and Nobuyoshi Yabuki*3

*1 Graduate Student, Div. of Sustainable Energy and Environmental Engineering, Graduate School of Engineering, Osaka University
*2 Assoc. Prof., Div. of Sustainable Energy and Environmental Engineering, Graduate School of Engineering, Osaka University, Ph.D.
*3 Prof., Div. of Sustainable Energy and Environmental Engineering, Graduate School of Engineering, Osaka University, Ph.D.

Keywords: Landscape Simulation;Virtual camera; Mirroring; Small Drone; Web-based AR; Occlusion.

1. Introduction

In the architectural and urban fields, consensus building is important among stakeholders to preserve good landscapes. In advance, Augmented reality (AR) can be used to study the future landscape in advance in harmony with the design target and the surrounding environment in advance. In conventional AR landscape simulations, a user holds the AR device directly in their hand or wears a head-mounted display (HMD) with the web camera on it. In this way, the viewpoints of the AR rendering is limited by the user's range of action. It is hard to study the landscape from a wider range of viewpoints, such as from the air.

To solve this problem, Wen et al.[1)] simulated an AR landscape from a free viewpoint using a drone to change the position and angle of the AR camera to view a three-dimensional scene. To involve more people in the review process, Yang et al.[2)] proposed a landscape simulation system integrating a drone and AR with remote communication technology. However, these systems had problems with poor usability.

Small drones are defined as drones of less than 200 g. They are classified as model aircraft under the Civil Aeronautics Act. They can fly in densely populated areas without a permit from the Ministry of Land, Infrastructure, and Transport. They are low cost and easy to use.

Web-based AR (WebAR), which renders AR on a browser, is being developed[5)]. WebAR does not require a specific platform and application to download. It is also easy for developers to implement AR. This feature will help users who are not experts in the field to understand it[3)].

In our previous study[4)], an AR landscape simulation system was proposed using small drones and WebAR with free-viewpoints. By using a small drone and WebAR, we developed a prototype of an AR landscape simulation. A user could study the relationship between a design target and the surrounding environment from the air.

However, our previous system can not hide virtual objects behind the real object to solve an occlusion problem. Therefore, the objective of this study is to develop an AR landscape simulation system with occlusion handling on WebAR from a wider range of viewpoints by a small drone. In this paper, we propose a method to realize an occlusion representation using instance segmentation. We evaluate the view, frame rate, and latency of the proposed AR landscape simulations with occlusion through an experiment.

2. Proposed System

The proposed system is an AR landscape simulation that tackles an occlusion problem and superimposes a virtual object on real-time video captured by a small drone. Figure 1 shows the conceptual diagram of the proposed system. Figure 2 shows a flowchart of the proposed system.

2.1. CAPTURING LIVE VIDEO FROM A SMALL DRONE

A user operates a small drone to reach the locations for the landscape simulation. Then, it captures live video by the drone camera from a wider range of viewpoints in real-time.

The captured video is displayed on the controller screen. The small drone and the controller communicate with each other via extended Wi-Fi. The above process is performed on a tablet and smartphone.

2.2. TRANSFERING VIDEO

Fewer hardware and software restrictions provide greater usability. We propose a method of transferring video using both mirroring and virtual cameras a machine independent system.

Mirroring is the act of displaying the same screen on multiple devices. It enables the video transfer between hardware.

A virtual camera enables desktop-captured video to be output by applications that use a web camera without connecting a physical web camera to the PC. It enables video transfer between software in the devices.

The video displayed on the screen of the controller in section 2.1 is transmitted to the PC by mirroring. The transferred video

日本建築学会情報システム技術委員会
第43回情報･システム･利用･技術シンポジウム論文集，152-155，2020年12月，オンライン
Proceedings of the 43rd Symposium on Computer Technology of Information,
Systems and Applications, AIJ, 152-155, Dec., 2020, Online

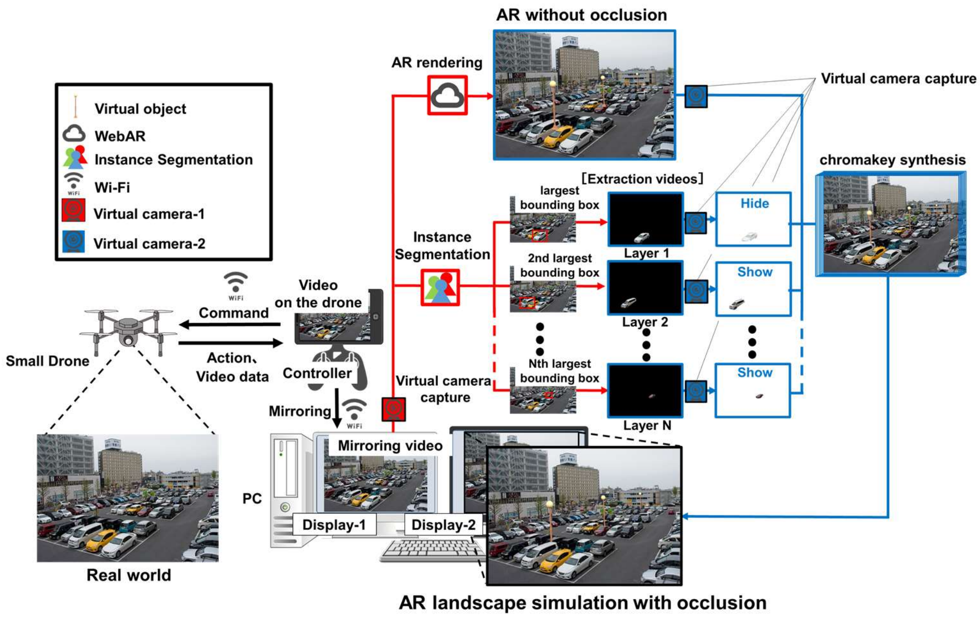


Figure 1. Conceptual diagram of the proposed system

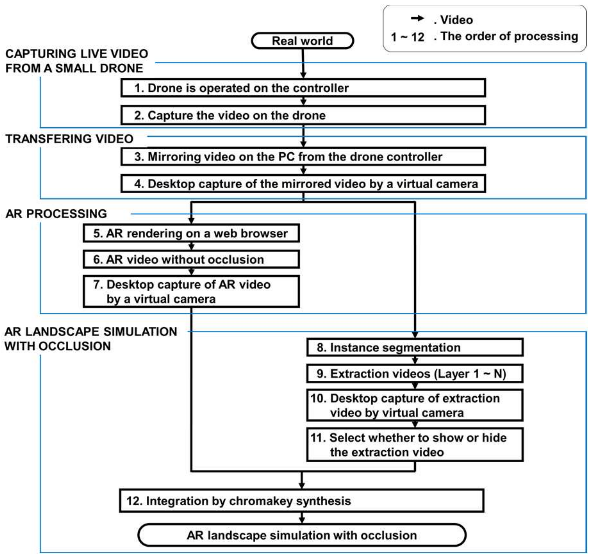


Figure 2. Flowchart of the proposed system

is displayed on the PC screen. The communication between the controller and the PC by mirroring is used Wi-Fi in the same LAN environment. The mirrored video displayed on the PC screen is captured on the desktop by a virtual camera.

The video is transferred to the AR environment and occlusion processes via mirroring and virtual camera processes. These processes reduce the burden of configuration and improve usability by the process of mirroring and virtual cameras.

2.3. AR PROCESSING

The AR platform is selected WebAR corresponding to a cross-platform to improve usability.

Mirrored video is captured by a virtual camera on the desktop in section 2.2. WebAR renders the captured video in AR on a web browser. The AR without occlusion is displayed on the PC screen. The displayed AR video is desktop-captured by the virtual camera in the same way as the mirrored video.

2.4. AR LANDSCAPE SIMULATION WITH OCCLUSION

To solve the occlusion problem, the extraction of real foreground objects is necessary. To extract the foreground objects, we employ instance segmentation that enables pixel-by-pixel object detection while distinguishing between objects of the same class.

In section 2.2, instance segmentation technique is used to extract foreground objects from the mirroring video captured by the virtual camera. The extraction videos of the recognized objects are displayed in order from the object with the area of the bounding box. The number of extraction videos is manually optional. The extraction videos are desktop captured by the virtual camera in the same way as the AR without occlusion in section 2.3. The captured extraction videos are manually selected to be shown or hidden according to the scene. The extraction videos are transmitted through the black background by a chromakey synthesis. The transferred video is superimposed on the AR video by AR processing.

The above process makes it possible to study the designed landscape on the PC screen in real-time, with dynamic occlusion handling between the real and virtual objects.

3. Experiment and Results

3.1. EXPERIMENTAL CONDITION

A prototype was built based on the proposed method. The hardware and software configuration are shown in Table 1 and Table 2.

3.2. EXPERIMENTAL ENVIRONMENT

In this section, we evaluate AR with occlusion between real and virtual objects in an outdoor experiment. We measured the processing speed and latency to check the performance. The system of the previous study was also built[4], and we compared the frame rate and latency with the proposed system prototype. A marker-based method is used as the geometric registration for AR.

Figure 3 shows the two experiment sites (Site A and B) in the same LAN environment. An AR operator is assigned to site B. A small drone operator is assigned to site A, and an AR operator is assigned to site B. At Site A, the AR marker (W900 x H900 mm) was placed vertically. Figure 4 (a) shows the drone operation of the Mavic mini to capture live video from various positions and angles, then transferred to Site B by mirroring. Figure 4 (b) shows the AR operator at Site B set the occlusion manually from the video transferred from Site A. In Site B, the AR with occlusion between real and virtual objects outputs on display so that we can study the future landscape correctly.

3.3. RESULTS

Figure 5 shows a screenshot of the AR displayed on the PC screen. The AR marker was often undetectable when the drone was moving; then, the video had to be captured without moving the drone in the air. With occlusion handly, the AR output correct when the drone operator did not move the drone.

Figure 6 shows the corrects output of AR with occlusion. We confirmed the position's accuracy of the virtual objects to be an error of 1~2 m from the correct position in Figure 7.

Table 1. The hardware of the proposed system

Name	Hardware	Feature
Mavic mini	Small Drone	Weight: 199 g Size: L245 x W290 x H55 mm FOV : 83°
DESKTOP-LVJBKOT	Desktop PC	OS: Windows 10 CPU: Intel® Core™ i7-8700k GPU: NVIDIA GeForce GTX 1080Ti
iPhone 11	Controller	Communication between the drone and the controller via extended Wi-Fi

Table 2. The software of the proposed system

Name	Software	Feature
AR.js & A-Frame	WebAR	Marker-based AR
Mask-R_CNN[5]	Instance segmentation	Data set: Common Objects in Contexts (COCO)
OBS Studio	Virtual camera	Multiple virtual cameras per software (up to 5)
Reflector3	Mirroring	Comunication between controller and Desktop PC Under the same wireless LAN

The measured frame rates and latency of the proposed system and the previous system are shown in Table 3.

Both values were calculated by the averages of five measurements. We confirmed that both the frame rate and the latency of the prototype have been improved.

4. Discussion

By extracting foreground objects, we have developed a system that enables AR landscape simulation with occlusion between real and virtual objects. In addition, the frame rate was partially improved compared to the previous system.

In addition, the AR marker was not detected during the small drone movement. It was caused by the disruption of videos due to movement. Several problems were found in the proposed system. First, the occlusion handling between virtual and real objects is inconvenient.

Table 3. Comparison of the proposed system and previous system

Method	Frame rate [fps]	Latency [s]
Proposed system	Background: 60 Foreground: 2.08	0.316
Previous system	25	0.354

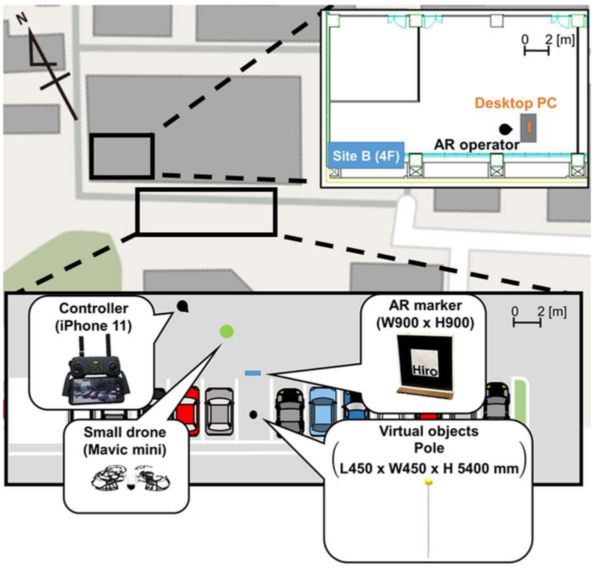
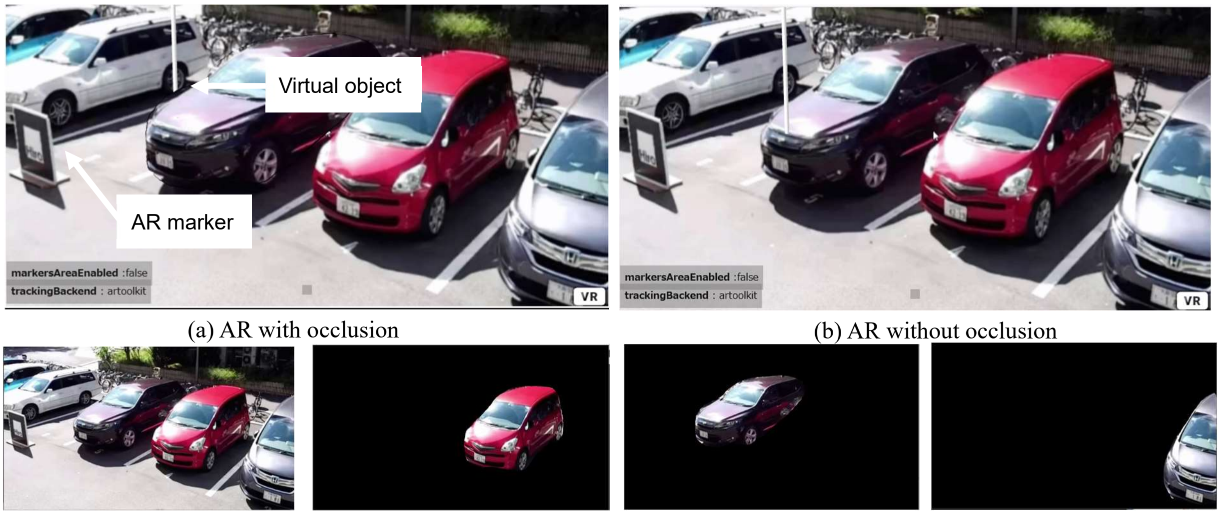


Figure 3. Situation on site A and site B

(a) Drone operator (b) AR operator

Figure 4. Use the system



(a) AR with occlusion (b) AR without occlusion

(c) Real world (e) Extraction videos (layer 1 ~ 3)

Figure 5. Outputs on the PC display at Site A

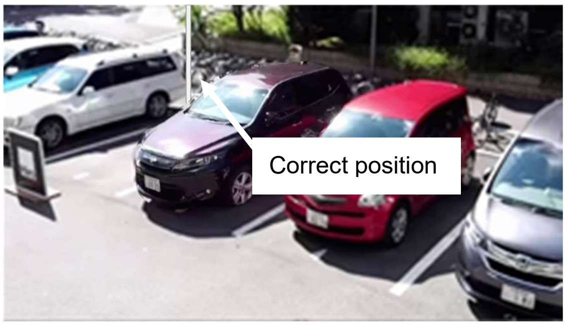


Figure 6. The correct position of the virtual object (sketch)

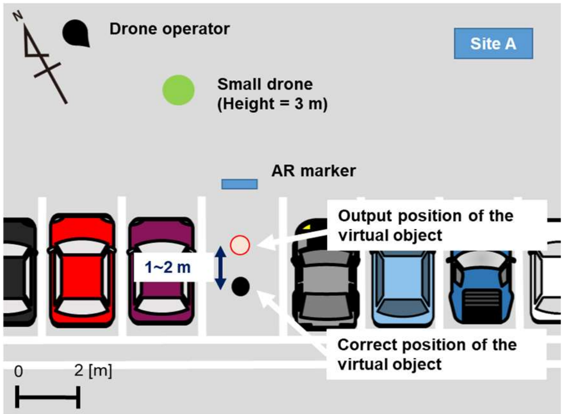


Figure 7. Error in the position of the virtual object

In the proposed system, the foreground objects were extracted manually. To make the system more flexible, foreground objects need to be extracted automatically. In addition, the frame rate of the foreground objects is low at 2.08 fps. The algorithm needs to be optimized to perform an AR landscape simulation in real-time.

Second, there is a limitation of mobility of the system. In the proposed system, the two sites must be in the same LAN environment. This restricts the location of the two sites. We need to construct a system that is not limited by the same LAN environment.

5. Conclusion and Future works

To simulate a realistic landscape after construction, we developed an AR landscape simulation method that partially solves the occlusion problem.

In the future, the representation of occlusion between virtual and real objects by automatic foreground objects is necessary and allows free movement by a small drone.

Acknowledgments

This research has been partly supported by JSPS KAKENHI Grant Number JP19K12681.

References

1) Wen, M., C., Kang, S., C. (2014). Augmented Reality and Unmanned Aerial Vehicle Assist in Construction Management, COMPUTING IN CIVIL AND BUILDING ENGINEERING ©ASCE 2014, pp.1570-1577.

2) Yan, L., Fukuda, T., Yabuki, N. (2019). Integrating UAV Development Technology with Augmented Reality Toward Landscape Tele-simulation, Proceedings of the 24th International Conference on Computer-Aided Architectural Design Research in Asia (CAADRIA 2019), Vol.1, pp.423-432.

3) Qiao, X., Ren, P., Dustdar, S., Liu, L., Ma, H., Chen, J. (2019). WebAR: A Promising Future for Mobile Augmented Reality State of the Art, Challenges, and Insights, in Proceedings of the IEEE, vol. 107, no.4, pp. 651-666, April, doi: 10.1109/JPROC.2019.2895105.

4) Kikuchi, N., Fukuda, T., and Yabuki, N. (2020). A Landscape Simulation System with Free-Viewpoint Using a Small Drone and Web-based Augmented Realit. AIJ KANTOU 2020, Summaries of Technical Papers of Annual Meeting, Information Systems Technology, pp.223-224.

5) K. He, G. Gkioxari, P. Dollar, and R. Girshick. (2017). Mask r-cnn. arXiv:1703.06870,