INTERGRATING UAV DEVELOPMENT TECHNOLOGY WITH AUGMENTED REALITY TOWARD LANDSCAPE TELE-SIMULATION

LIANG YAN¹, TOMOHIRO FUKUDA² and NOBUYOSHI YABUKI³
¹,²,³ Division of Sustainable Energy and Environmental Engineering, Graduate School of Engineering, Osaka University, Japan
¹ u215231e@ecs.osaka-u.ac.jp ²,³ {fukuda|yabuki}@see.eng.osaka-u.ac.jp

Abstract. Augmented reality (AR) is an emerging landscape simulation technology being used in the construction industry to reduce losses in subsequent projects by reviewing the landscape before a building is completed. However, since AR projects virtual models into the real world through portable devices, the designer’s review perspective and the number of people able to participate in the review process is limited. Therefore, a system that combines AR and unmanned aerial vehicle (UAV) development with telecommunications technology was designed and prototyped to use the UAV camera as the source of the video stream of AR. This frees the designer’s review perspective through ground control and allows remote communication with off-site people, thus allowing more users site access and improving system usability. This paper details the construction of the integrated system, including the integrating of different development languages, environments, and mutual calls used, the AR and UAV development modules, the construction process of the telecommunication protocol, and mutual data interoperability.

Keywords. Landscape simulation; tele-simulation; Markerless Augmented Reality (AR); Unmanned Aerial Vehicle (UAV); telecommunication.

1. Introduction

Once construction begins, changes or corrections in the project design can be difficult and costly to implement; any flaw in the design process can cause irreparable losses. Augmented reality (AR) expects to be able to reduce these losses by previewing the target construction site using a virtual building model of the design. Wang et al. (2014) highlighted the need for a structured methodology fully integrating AR technology in building information modeling (BIM) and formulated methods for configuring BIM + AR prototypes. Similarly, Fukuda et al. (2014) argued that importing AR into visual landscape assessment would reduce the time and cost required to perform 3DCG modeling and proposed a feasible and effective handheld AR system for users to carry out landscape simulations.

Prior efforts have been made to integrate AR technology into other industries, including some practical efforts involving AR and unmanned aerial vehicles (UAVs). For example, Stefanie et al. (2014) used AR to render real-time virtual 3D building models, whereas Wang et al. (2015) used farm UAVs and a GPS-assisted AR system to provide farmers with a more lucrative and environmentally friendly cultivation program. Ji et al. (2017) used AR technology to improve upon the experience of UAV aircraft operators, who had previously only been subject to 2D displays. Huuskonen et al. (2018) used UAV to collect soil RGB images and used AR glasses to aid farmers collecting samples from the management area. Jacopo et al. (2017) constructed a Visuo-haptic AR system with additional visual tactile sensors to help operators remotely detect nuclear radiation areas. Wang et al. (2016) added a simple AR virtual tag to the video stream that the UAV returns in real time through the AR browser and GPS positioning information. Although the above researches use AR and UAV, it does not really combine them. They only use the UAV’s aircraft to collect data and then analyze it with AR. There is a time difference between these two stages. The use of UAVs in the construction industry has also become more common in recent years. Ham et al. (2016) simplified the process of collecting visual data by using drones to visually monitoring buildings. Siebert et al. (2014) tested the location estimating performance of UAVs’ in civil engineering, and identified the feasibility of UAV development in the construction industry.

Although implementation of AR in the construction industry has improved the architectural design process, many problems still exist. First, the AR and UAV have not always been implemented in real time. Secondly, since the AR image has been used to acquire real-world information through the camera of the device which user is using on-site and then synthesize it with the 3D model in real time, the user is only able to observe the AR image on the on-site device. This lack of portability by using big screen devices and small screen size of mobile devices such as cellphone or tablet limits the number of participants and the degree of participation, which will reduce the efficiency of the entire landscape simulation process. Additionally, as real-world information is acquired using the on-site device’s camera, the review angle of the handheld AR system is also limited, thereby breaking the integrity of the simulation perspective. The main purpose of this system is to improve AR system currently used in construction industry and to achieve landscape tele-simulation. The word ‘tele’ here refers not only to the distance from the participants off-site to the construction site completing by telecommunication protocol, but also to the the distance from camera used by the AR system to the handheld device by completing UAV+AR.

Therefore, an outdoor landscape simulating review system integrating AR and UAV technologies with telecommunication technology is proposed. The proposed system uses a real-time video stream returned by a camera aboard the UAV to generate AR images on multiple devices, which expands the perspective of the simulation review and increases the number of people that can be involved in the simulation process. A prototype of the proposed landscape tele-simulation using UAV and AR is designed to achieve two main functions: (1) outdoor landscape simulation system using markerless AR with multi-angles, which is
Currently limited by on-site device’s screen, and (2) the allowance of more users to participate in the landscape review process by transmitting the simulated scene to off-site devices through telecommunication technology in real time. Outdoor stability tests were performed for AR module and UAV developed function (Yan, et al. 2018).

2. Proposed Method
2.1. SUMMARY
The technology used can be divided into three main areas:

- **Simultaneous Localization and Mapping (SLAM)-based markerless AR technology**: for real-time rendering of virtual building models in a real-world environment, used to generate landscape simulation images in real time in the game engine.
- **UAV developing technology**: to review designs from a number of manpower-inaccessible perspectives, used in mobile development platforms (iOS/Android) to connect UAV aircraft with the controller and receive the returned streaming data.
- **Telecommunication technology**: for real-time data transmission of AR images to off-site participants for better reviewing capabilities, used to transfer the on-site simulation to off-site participants via the communication protocol.

The system integrates these three technologies with 3D model prepared by BIM/CAD software into one platform (mobile development platform or C sharp based game engine platform) and outputs a package file usable on portable devices. The developed integration method varied depending on the final development platform; the integration methods proposed for a mobile development platform and a game engine are summarized in Figures 1 and 2, respectively.

In the method detailed in Figure 1, method one, the C++ project from game engine is imported to the mobile development platform, which UAV developing project is based on, to ensure that the original code structure in it remains to the greatest extent. In mobile development platform, different modules call each other in the project by adding new activities.

In the method detailed in Figure 2, method two, implementing the mobile developing project to the game engine, which is exactly the opposite of method one. We should turn a native Android application to a dynamic library at first (.aar), and import it into the game engine with C++ project already installed as a plug-in.

Unlike method one, method two requires to redesign the layout in UAV developing project from mobile development platform by GUI tools in game engine, resulting in an excessively long code, which increases the likelihood of errors to flashback when calling the code. Using the mobile development platform can also reduce the running time by obfuscating resources. Therefore method one was selected. An overview of the system setup, including two test methods for the final output program, are shown in Figure 3, and Figure 4 presents a conceptual diagram for the entire integrated system. Firstly, the 3D model will be created by
BIM/CAD software, then it will be imported into game engine to complete the AR part, and then it is exported to the Android development platform as an acceptable format library, finally it will integrated into the existing UAV developing project on it. After the mutual calling of the two development platforms is completed, the integration of the project and the library is performed. The integration of project and library mainly has three parts: the call between activities, the integration of layout and the replacement of video stream source. Final completed program can be tested by real device or emulator.

![Data flow based on Android developed platform.](image1)

![Data flow based on game engine.](image2)

2.2. UAV SETUP

Of the many functional UAV interfaces, the designed system mainly uses the remote control interface and camera. The streaming video is returned to the Android developing platform to synthesize the AR image in the next step. The remote controlling function of the aircraft required by the system and the real-time streaming data transmission can be realized by calling different interfaces in the UAV development module. After implementing these functions, the video stream originally used in the AR project is replaced with that obtained from UAVs’ camera.
Figure 3. Complete data flow in integrated system.

Figure 4. System schematic in actual use.
2.3. SLAM-BASED MARKERLESS AR

2.3.1. Making the 3D model in advance

The building information models must first be imported from BIM/CAD software to game engine to prepare for AR rendering in the next step.

2.3.2. AR tracking

AR technology can be roughly classified into two types by the location registration method: sensor-based AR that is registered by a sensor such as GPS, and image-based AR that recognizes feature points from image. Image-based AR is divided into two types whether uses special markers or not, which is named marker-based AR by using it, another one is marker-less AR.

SLAM technology allows for the generation of an environmental map while estimating the self-position without using the map at the same time. The SLAM algorithm here stands for visual SLAM which only uses the camera as a sensor. Widely used in robotics engineering, this tool improves the stability of the AR tracking algorithm. Therefore, the designed tele-simulation system uses a markerless AR algorithm based on SLAM technology. Here, a 3D construction model is imported into the game engine by other BIM/CAD modeling software before the AR program begins. Although SLAM-based marker-less AR increases the stability of the 3D model during the movement, it cannot calibrate the initial generation position of the 3D model, so it is set to generally appears in the center of the screen.

2.4. TELECOMMUNICATION TECHNOLOGY

Different digital devices share the landscape simulation review scene through the network communication protocol. Users can also exchange ideas through the device microphone.

3. System setup

System construction was divided into three parts: preparation of the 3D model, AR implementation within the game engine, and the integration of the functions on the Java developing platform.

3.1. 3D MODEL PREPARATION

To achieve the goal of landscape tele-simulation, the system required accurate and verifiable 3D construction models as the basis for AR image formation. The model design tool SketchUp was used to accomplish this. The 3D model is prepared in SketchUp and then exported it for game engine to use as AR model. However, limitations of the game engine used, Unity, require that 3ds Max must be used as an intermediary to convert the file from SketchUp to Unity, as shown in Figure 5.
3.2. AR IMPLEMENTATION IN GAME ENGINE

The 3D construction model is then imported into the Unity game engine to implement the markerless AR phase of integrating 3D model with the real world in real time. The system uses the SLAM-based markerless AR algorithm named KUDAN SLAM AR to achieve the required outdoor review function. The process for implementing a single AR module is shown in Figure 6.

3.3. INTEGRATION ON THE JAVA DEVELOPING PLATFORM

The AR project is then exported from Unity as a Java project because the library responsible for live streaming video playback from UAV should be implementable on the iOS/Android development platform. The AR project is then integrated as a module into the Android development platform named Android Studio as a part of DJI mobile SDK, and also the telecommunication protocol part will be added after whole system completed. Finally, Android Studio exports the executable program by connecting the aircraft and its controller, allowing the creation of the AR image using the streaming video captured by UAV rather than using the devices’ camera. The development process of the integrated system is shown in Figure 7.

To enable the AR project written in C++ to be successfully called by the Android development platform, it needs to be completed with an additional connection conversion interface. This relationship is detailed in Figure 8, where Java calls C/C++ with a .so library and C sharp and Java can call each other via Native Development Kit (NDK) (Figure 8).

NDK is a toolkit provided by the Android development platform that allows other developing languages to be used on it. A Java Native Interface (JNI) is the interface that provides the call according to NDK.

The calling logic inside the DJI activity of the code is presented in Figure 9. Here, the DJI modules, AR modules from Unity and telecommunication protocol
module in the system are managed by similar but independent activities which can jump to each other.

Figure 7. Process flow in integration phase.

After adding modules from the AR project and the telecommunication protocol, the one activity mode of DJI project is transferred to a multiple activities mode that can jump between each activity inside the project.

To ensure that the last exported executable program has only one exit, the system uses the DJI mobile SDK as the main program; the rest of the embedded projects are used as the dynamic link library called by the main program.

Finally, to allow the embedded AR library to reference the video stream from the UAV instead of the camera originally used on the device, it is necessary not only to intercommunicate the data calling between two development platforms, but also need to learn more deeply about the Android development itself, in order to achieve the purpose of replacing the video stream source in future research.

4. Implementation
4.1. DEVELOPMENT BACKGROUND LEARNING

In order to find out the specific method of building the system, we learned a lot about the AR, the game engine and the Android development. In order to ensure the feasibility of the system, we considered SLAM with AR and finally chose KUDAN AR to use.
In terms of system development, we learned the development ideas and development methods of Java language, also the correspondence between various files and so on. The most time-consuming part is to find data inter working and calling methods between different modules used in the system and also the reason why to choose Android developing platform as final platform.

![Process flow inside DJI developing part.](image)

4.2. SYSTEM INTEGRATION

In order to achieve the goal of landscape tele-simulation, Unity and Android Studio must first be associated, and then the live streaming data by UAV can be used to generate AR images in real time by replacing the streaming video source from device’s camera. The integrated system is ready to run after overcoming many different flashback errors. The only problem is that the live streaming data has not been replaced by the data from UAV’s camera, and we will solve this problem in the future.

5. Conclusion and future work

In an effort to build an outdoor landscape simulating review system, this paper designed and developed a integrated system, analyzed the system from the aspects of development feasibility and call logic, and tested the features of each developed module. AR and UAV development technologies were integrated with telecommunication technology on an Android developing platform. The UAV developing project operates as the main program, whereas the other functions are
dynamic link libraries called by the main program. In the proposed integrated system, the AR image is synthesized using a SLAM-based markerless AR algorithm with a real-time video stream returned from the UAV that can be shown on portable digital devices. Utilizing the camera of the UAV expands the visibility of the field and the use of telecommunication technology allows more people to participate in the review process, which will make the whole landscape simulation process more efficiency than before.

The algorithms used were also verified and compared, as was the feasibility of the system. Future work will involve the completion of all system development processes including importing the telecommunication function module and testing tele-simulation function.

Acknowledgement

This research was partly supported by JSPS KAKENHI Grant Number JP16K00707.

References


Ham, Y., Han, K.K., Lin, J.J. and Golparvar-Fard, M.a.n.i.: 2016, Visual monitoring of civil infrastructure systems via camera-equipped Unmanned Aerial Vehicles (UAVs): a review of related works, *Visualization in Engineering 4.1*.


