

URBANSIGHT

Application of Digital Twin to Augmented Reality

TZU-HSIN HSIEH¹ and TAY-SHENG JENG²

¹*Department of Computer Science and Information Engineering, National Cheng Kung University*

^{1,2}*Department of Architecture, National Cheng Kung University*

¹*celinehsieh68@gmail.com, 0009-0007-2031-7928*

²*tsjeng@mail.ncku.edu.tw*

Abstract. BIM (Building Information Modelling) is currently used to construct building components, which does not provide good assistance for environmental sensing data and user feedback for management and maintenance. There are often problems with timely synchronization of building model information. In order to extend the BIM technology from the building design phase to the facility management phase, a 3D visualization platform is needed to optimize the efficiency of building management by introducing Digital Twin. This study therefore aims to combine Digital Twin developed through building information modelling with augmented reality (AR) headset to achieve automated control of building operation management for future urban governance. In this paper, we use Unity with Autodesk Forge API to build a BIM model with building information in the real site and display it in Microsoft HoloLens 2, which is a wearable device. First, Autodesk Forge API is used to build the monitoring platform of the field model. The data collected by the sensors are transferred to the IoT base station built by Raspberry pi to build the database. Then the digital twin dashboard interface on HoloLens 2 is displayed to control the sensors in the environment. This system breaks the 2D interface of the existing urban management interface and visualizes it in 3D in augmented reality (AR), creating a more intuitive and efficient way to manage smart cities.

Keywords. Digital Twin, Augmented Reality (AR), Internet of Things (IoT), Smart City, Management Platform

1. Introduction

In recent years, the application of Building Information Modelling (BIM) into design and engineering has become increasingly popular. With the aim of incorporating data in building life cycle into 3D computer-aided design (CAD) models of buildings, thereby reducing costs and preventing errors. However, BIM still falls short in updating

the current state of the building in real time, making it difficult to meet the need for real-time information transformation throughout the building life cycle, from design to construction, operation, and maintenance.

To solve this problem, Digital Twin has emerged as a solution. It creates a dynamic digital model of the physical environment, allowing users to visualize information in a virtual environment. This study integrates digital twins into augmented reality (AR) using head-mounted devices for real-time synchronization. The goal is to provide intuitive management platforms for building administrators and urban decision makers. Digital twins ensure up-to-date data availability throughout the building life cycle. The study analyses the feasibility and challenges of integrating digital twins and augmented reality (AR) in smart cities. Its objective is to explore the strategic development of digital twins in augmented reality (AR) for future smart city management.

2. Related Work

2.1. DIGITAL TWIN CONCEPT AND DEVELOPMENT

Building design and management tools have been developed for decades, starting with computer-aided design (CAD). The term "Building Information Modelling (BIM)" was first coined in the early 1990s (van Nederveen and Tolman, 1992) and has attracted widespread attention and application in the past. Each of these phases can be divided into overlapping information layers that require an effective information exchange strategy to achieve interoperability across all lifecycle phases (Deng et al., 2021).

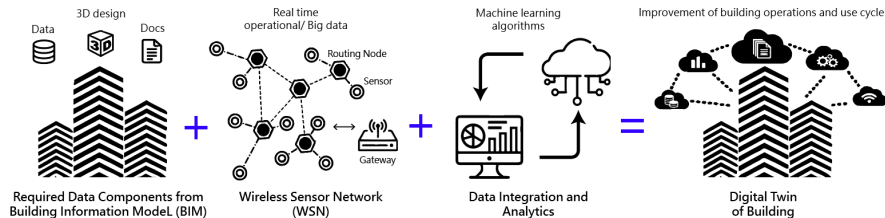


Figure 1. Basic components for creating a digital twin (revised from Khajavi et al., 2019)

However, BIM itself usually provides only static data of the built environment and cannot automatically update the real-time information in the model. With the emergence of the Internet of Things (IoT), defined as the interconnection of sensing devices capable of providing information exchange across different platforms (Gubbi et al., 2013), the integration of sensing data deployed in the environment and data provided by BIM models in real time has led to the concept of Digital Twin. Digital Twin aims to synchronize real-world and virtual platforms to control facility management, environmental monitoring, and other lifecycle processes in the built environment, connecting data to bring the virtual and real worlds together. Digital twins in buildings can use a variety of sensors and the Internet of Things (IoT) to create a real-time view (Figure 1). This dynamic view enables real-time analysis, informed decision making, building efficiency and comfort enhancement.

2.2. AUGMENTED REALITY APPLICATIONS

Digital twins replicate the physical world in a virtual environment, while augmented reality (AR) merges virtual elements with the real-world using display screens like cell phones or glasses. Augmented reality (AR) enhances the user's perception of the real environment by incorporating computer-generated displays, sounds, text, and effects (Dunleavy and Dede, 2014). It provides information that may not be directly perceivable in the physical world. In Figure 2, four rapidly evolving technologies are summarized that correspond to the four stages required to develop augmented reality applications (Loijens et al., 2017). AR integrates the virtual and real worlds through technologies such as cloud computing, location tracking, mobile devices, and user interfaces. Augmented reality also finds applications in the AEC industry, where it is used for tasks like construction site monitoring. For instance, Zaher et al. developed a system that combines BIM and AR on smartphones to monitor construction progress, leveraging the portability and visualization capabilities of mobile devices in construction projects (Zaher et al., 2018).

Technologies of developing AR applications			
Cloud Computing Environment	Localization Technologies	Portable and Mobile Devices	Natural User Interface
<ul style="list-style-type: none"> • BIM • Storage Services • Database • Internet • Edge Computing 	<ul style="list-style-type: none"> • GPS • SLAM • IPS (Wi-Fi, RFID...) • VPS (camera input) 	<ul style="list-style-type: none"> • Smartphones, Tablets • Wearable AR Devices • Handheld AR Devices 	<ul style="list-style-type: none"> • Gesture Recognition • Voice Recognition • Eye Tracking • Motion Capture

Figure 2. Technologies of developing AR applications

2.3. RELEVANCE

Digital Twin solves the problem of updating model information and status in real time for BIM. However, the lack of intuitive and interactive user interface has been described in some literature as difficult to understand and use. On the other hand, Augmented Reality (AR), while not creating a virtual world in sync with the real world like Digital Twin, enhances the real world by visualizing data and information. For example, to provide information to assist in construction, or maintenance management, allowing for a smoother flow through all phases of the building lifecycle. Therefore, this study combines digital twins with augmented reality. Digital twins enhance augmented reality systems by accurately linking to remote sensors, helping to pinpoint the location of users and nearby objects. Augmented reality, in turn, adds an interactive and user-friendly layer to Digital Twin, making it easier for users to comprehend and utilize the information effectively for better decision-making and maintenance.

3. UrbanSight

3.1. SYSTEM DESIGN

To validate the digital twin application for building management, this study focuses on the system's development architecture and experimental construction method. As depicted in Figure 3, the system comprises two primary parts: the back-end (web platform) and the front-end (Unity platform). The front-end uses the collected data from the back-end to build a managed user interface, which is deployed in the Microsoft HoloLens 2 headset for managers to monitor the site more intuitively.

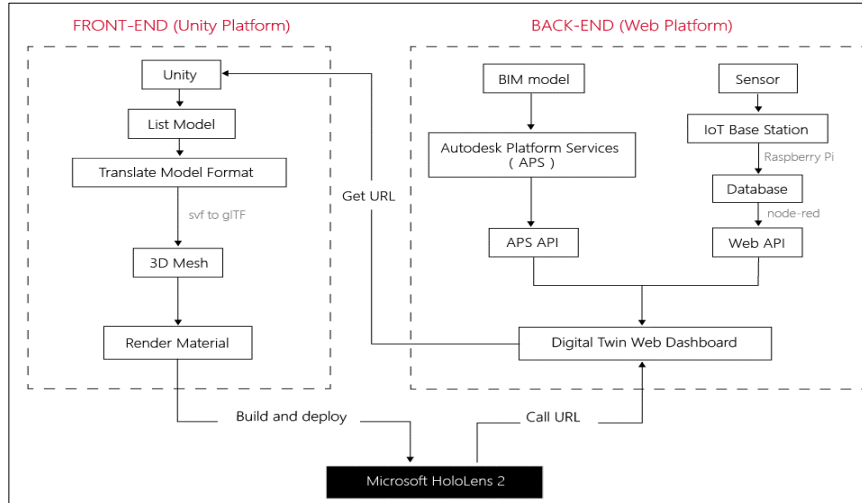


Figure 3. System Framework

3.1.1. Development Environment and Tools

Autodesk Platform Services (formerly Autodesk Forge) provides APIs to convert model files into a more usable format for web and mobile viewing and analysis. It is a cloud-based platform for developers to create and manage applications for the architecture, engineering, construction, and manufacturing industries, empowering innovation for cloud applications and system integration, and enhancing the efficiency of information transformation. Through Autodesk Platform Services, this study can access 2D and 3D design, engineering, and construction data uploaded to Autodesk Platform, build a database of sensor data, and build an IoT base station by Raspberry pi Linux system to combine BIM modelling and sensor data. The data is transferred to the real-time Digital Twin Web Dashboard, which provides an initial interface integration with the front-end (Unity platform).

For the Augmented Reality (AR) user management interface, this study uses the free personal version of the Unity Engine game development engine, version 2021.3.11f1, for prototyping, and all development scripts are built in C# programming language and use the Microsoft Mixed Reality Toolkit to help automate the setup of the mixed reality project. MRTK (Mixed Reality Toolkit) is a developer's mixed reality toolkit for Microsoft HoloLens that helps accelerate cross-platform mixed reality development in Unity by providing cross-platform input systems and building blocks for spatial interactions and UI, allowing users to quickly build interfaces through in-editor emulation, allowing them to view changes immediately. MRTK provides an extensible architecture that allows developers to exchange core components and supports a variety of devices in addition to Microsoft HoloLens 2.

3.1.2. System and Users

The system planning of UbranSight is divided into a back-end (web platform) web

page builder and a front-end (Unity platform) interface display. Since this project uses two different platforms, the system operation relationship diagram is drawn in Figure 4. Microsoft HoloLens 2 will return the corresponding model and sensor data for display on the user interface. On the other hand, users (administrators) of the web platform (Digital Twin Web Dashboard) can constantly update the new BIM models through the Autodesk Platform Services cloud service to store the models, and the Digital Twin Web Dashboard provides users with a maintenance update viewing platform. The Microsoft HoloLens 2 and Digital Twin Web Dashboard are connected via the network, and when Microsoft HoloLens 2 calls the Digital Twin Web Dashboard's url, the Digital Twin Web Dashboard returns the corresponding model with the transmission of the model. Dashboard returns the matching model and sensor data.

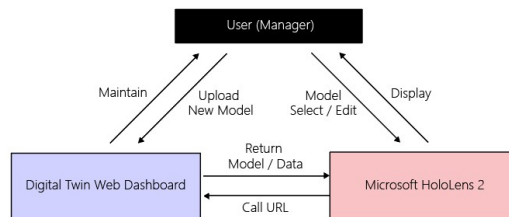


Figure 4. System Operation Relationships

In this study, the system implementation is divided into back-end (web platform) and front-end (Unity platform). The following section describes the back-end (web platform) that deals with the data collection and interfacing of models and sensors, and the front-end (Unity platform) that uses the collected data to construct a user interface.

3.2. BACK-END SETUP (WEB PLATFORM)

To experiment the practicality and effectiveness of this system, the research office of National Cheng Kung University was used as a test site for digital twin model construction and sensor deployment connection. The back-end setup (Web Platform) is planned to be divided into two parts as follows.

1. Digital twin modelling: Implementation of BIM model building and cloud publishing corresponding to the real space.
2. IoT sensor connection: Realize the deployment of IoT sensing and database construction in the real space to establish real-time data connection to the cloud BIM.

Finally, the two parts will be integrated and attached to the Digital Twin Web Dashboard to provide a maintenance platform for users to update their models.

3.2.1. Digital Twin Modelling

This study uses digital twin technology developed by Building Information Modeling (BIM) to create a platform for model change and maintenance for administrators to control the actual state of a spatial area. The technology enables real-time 3D visualization of sensor data in the most intuitive way. The webpage displays the historical environmental parameters of the space and its environment, such as air parameters, crowd status, etc., to assist administrators or users in decision making. The experiment starts with using Autodesk Revit to build the BIM of the experimental field. The basic elements of Revit's building information model can be broadly divided into (1) Model Elements, (2) Datum Elements, and (3) View-Specific Elements.



Figure 5. Digital Twin Modelling

Model Elements in a digital model include fixed objects and furniture like walls, roofs, windows, and chairs. Datum Elements encompass structural guidelines such as grids and floors. View-Specific Elements are used for annotations and dimensional marks. Attributes of each component, like usage status and environmental conditions (temperature, humidity), are monitored and adjusted using sensor data.

The model information is then uploaded and stored through Autodesk Platform Services (APS), utilizing four open API functions: Authentication, OAuth, Data Management API, Model Derivative API, and Viewer API. The overall flow of Autodesk Platform Services (APS) platform modelling is shown in Figure 6. Developers create an account and an application on Autodesk Platform Services, receiving a client ID and password. OAuth, a standard for authorization, allows third-party app access to resources on the platform via an Access Token. The Data Management API, mainly used for HTTP Get requests, handles data access and management, including folder and object management in the BIM 360 database.

The Model Derivative API supports over 60 industry model formats and is used for file conversion, including 2D and 3D models, into web-viewable formats like SVF/SVF2. This API also captures model data and assists in listing files for rendering in the viewer SDK. Autodesk Forge Tools, a plugin for Visual Studio Code, is used in this study for efficient interaction with these APIs, enabling easy upload, download, viewing, and deletion of models.

Lastly, the Viewer API allows web-based viewing of models. This JavaScript library, based on three.js, facilitates browsing,

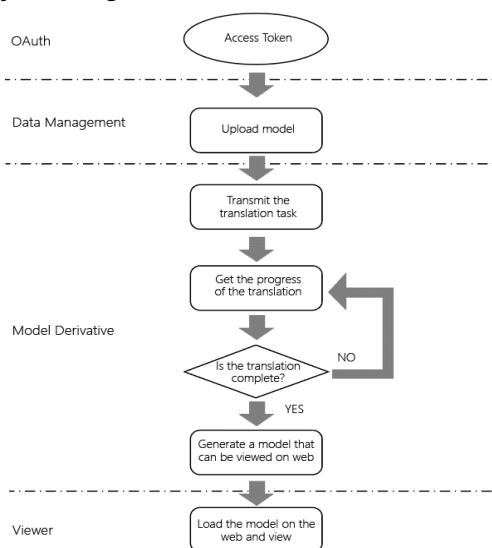


Figure 6. Autodesk Platform Services Modelling Process

viewing, and collaboration on 2D and 3D models within a webpage. In this study, it is used to set up the application interface and add functions such as profiling view, explosion view, measurement, and accessing component attributes, enhancing the application's capabilities.

3.2.2. IoT Sensor Connection

The sensors in this study on the Internet of Things use ADI's Smart Air Detector (SD) and Eagle Eye sensors. The SD is used for real-time sensing of gas composition (temperature, humidity, CO2, PM2.5, and other impact factors) and reporting data for back-end data collection and analysis. Eagle Eye is also used to detect the distribution of pedestrian flow

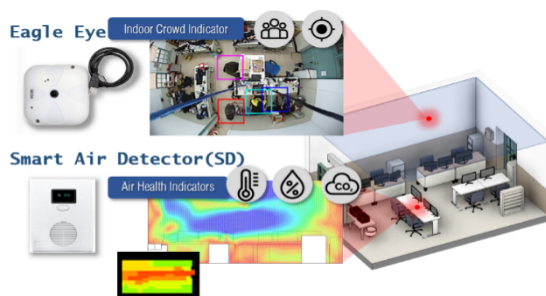


Figure 7. Deployment of sensors

in aisles and areas of the building to understand the usage of the building. In terms of sensor components, Eagle Eye was deployed in the ceiling of the research office. Based on the results of the BIM-assisted calculations, the smart air detector SD was placed in the most suitable position to obtain air health indicators and indoor occupant flow indicators for the site.

This project integrates two environmental indicators using sensor data, which is then transmitted to an IoT base station built with Raspberry Pi, creating a database. Control of the sensor is managed through a Digital Twin Web Dashboard, allowing adjustments based on detected improvements in environmental quality. For the database, we use Node-RED and MongoDB to construct a NoSQL database, as illustrated in Figure 8. MongoDB, a NoSQL database, stores data in files rather than in a relational format, offering flexibility to suit the specific needs of the building space environment. It utilizes JSON/BSON for data storage and access. Leveraging the JOIN function and the Mongoose library, a popular NPM module, MongoDB enables the development of complex relationships and efficient database designs, free from the constraints of traditional queries. This adaptability makes MongoDB the chosen platform for building our project's database.

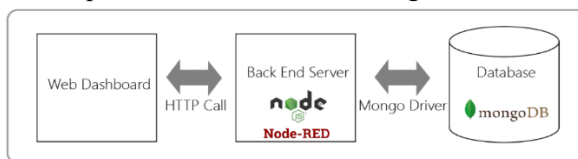


Figure 8. Database setup and connection flow

3.3. FRONT-END SETUP (UNITY PLATFORM)

After the completion of the back-end setup (Web Platform) in the previous section, this section proceeds to the front-end setup (Unity Platform) to import the models and data integrated in the Digital Twin Web Dashboard into the augmented reality (Microsoft

HoloLens 2). The model files are converted to a format that can be rendered by Microsoft HoloLens 2. To achieve a more intuitive, replaceable, and editable model user management interface.

3.3.1. Interface Integration

For the Augmented Reality (AR) user management interface, this study was developed using the Unity Engine. The C# programming language was written to allow Unity to communicate with the Node.js server to load models on the web platform (Digital Twin Web Dashboard). The glTFast-

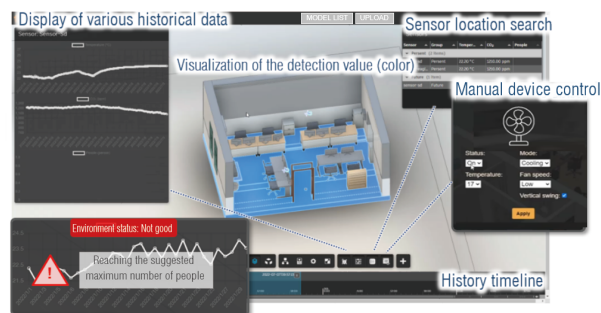


Figure 9. Interface connect to Digital Twin Web Dashboard

generated glb files are loaded and previewed to convert the model format on different platforms. The model type and rendered material are then created through 3D Mesh, and the rendered glTF model material is converted to a standard material that can be rendered by Microsoft HoloLens 2.

For model placement, the study creates a pointer script in Unity that interacts with the Node.js server. This script can identify objects and their locations by decoding the object ID in the color channel. Key variables include the collider (interaction point of the ray), distance (from ray start to collider), triangle index (specific triangle in the mesh), and point (coordinates of ray and collider intersection, defined as Vector3 objects). Vector3 origin specifies the ray's origin point, and a Draw Ray function is used to render the Ray in the virtual world.

3.3.2. Augmented Reality Components

This study uses the Microsoft Mixed Reality Toolkit (MRTK) to set up a mixed reality application in Unity to be compatible with the Microsoft HoloLens 2 headset. MRTK provides components and functionality to accelerate the development of cross-platform mixed reality applications. It was built in Unity using Universal Windows Platform and then deployed to HoloLens 2 using Microsoft Visual Studio using Wi-Fi or USB. In this case, the Wi-Fi deployment project was chosen to connect to Microsoft HoloLens 2 by entering the IP address of the device for matching and remote connection.

4. Experimental Results & Analysis

4.1. DATA CONNECTIVITY

In this study, we use a Raspberry Pi as a cost-effective IoT base station for connecting sensor data, and MongoDB for its scalable data storage capabilities. The database effectively handles model building data and sensor data, but the Raspberry Pi's limited processing power and memory make it unsuitable for larger IoT systems needing

extensive data processing. Currently, the system manages two weeks of historical data smoothly. However, scaling this model to a smart city level may overwhelm the Raspberry Pi's capacity, especially for processing large-scale urban data like traffic and weather.

The study also utilizes a Node-RED database, which is user-friendly but can become slow and resource-heavy with large data sets or complex operations, particularly on smaller devices like the Raspberry Pi. Expanding the experiment and increasing the BIM model size could make Node-RED's functions more complex and challenging to manage, as they depend on external libraries.

4.2. USER INTERFACE

Figure 10 presents the HoloLens 2 user interface. The left image shows the startup screen where the user selects our app to enter the augmented reality management interface. The middle image shows the initial screen of the app, still loading the model from the Digital Twin Web Dashboard, with a real environment, a virtual menu, and two virtual cubes for orientation. The right image displays the BIM model fully loaded, allowing the user to switch between models using the virtual menu.

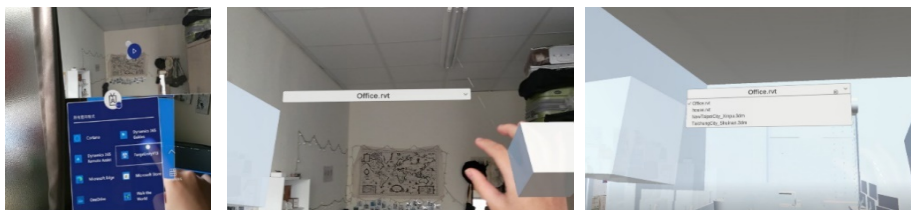


Figure 10. HoloLens 2 user interface
(Left: HoloLens 2 home screen/ Middle: Application Start/ Right: Display BIM model and menu)



Figure 11. Model Object Display (Left: PC device in unity / Right: HoloLens 2)

After testing with HoloLens 2, we discovered that the Unity-designed virtual menu doesn't work well with the HoloLens 2's augmented reality interface. Users have to use a virtual line from their fingertip instead of directly touching the menu. To improve this, the HoloLens 2's user interface should be updated with the Mixed Reality Toolkit (MRTK) modules for a smoother and easier-to-use admin interface.

In our application, model objects are dynamically loaded into the AR environment to keep the user experience updated. However, we've faced challenges with the HoloLens 2 device in accurately rendering textures and colors of these model objects. Sometimes, the objects appear white even with the Mixed Reality Toolkit/Standard shader. This issue could be due to the device's limited processing time, affecting its ability to render during loading. Further investigation and optimization are needed to

enhance the visual quality and user experience in our mixed reality application.

5. Conclusion

This study creates a management system combining digital twins with AR headsets, creating an integrated visual management platform. It focuses on using digital twins for effective building lifecycle management and maintenance, facilitated by an AR-enhanced interface for easy use by administrators. The Digital Twin Web Dashboard, developed with Node.js, is integrated into Microsoft HoloLens 2 using the Unity Engine. The system features a back-end (web platform) and a front-end (Unity platform), enabling administrators to intuitively monitor and control the interface.

To enable real-time data transmission, sensors are set up in the set field space. This sensor data is stored in a database and linked to a cloud-based Building Information Modeling (BIM) system for live display. For the Augmented Reality, the Microsoft Mixed Reality Toolkit was used to help automate the setup of the mixed reality project to be compatible with the Microsoft HoloLens 2 headset. By integrating Digital Twin technology with AR, the system provides up-to-date information on urban buildings and infrastructure. The system integrates Internet of Things (IoT) and Building Information Modeling (BIM) data in an easy-to-use Augmented Reality (AR) interface, enhancing smart city management and decision-making.

Acknowledgement

This work was supported by the Taiwan Ministry of Science and Technology, grant No. MOST 111-2221-E-006 -049 -MY2. We sincerely thank Pei-Chi Tsai, Chien-Kai Kuo, Yen-Ching Chang, and Sheng-Heng Liao for their contributions to this project.

References

- van Nederveen, G. A., & Tolman, F. P. (1992). Modelling multiple views on buildings. *Automation in Construction*, 1(3), 215-224.
- Deng, M., Menassa, C. C., & Kamat, V. R. (2021). From BIM to digital twins: A systematic review of the evolution of intelligent building representations in the AEC-FM industry. *Journal of Information Technology in Construction*, 26.
- Khajavi, S. H., Motlagh, N. H., Jaribion, A., Werner, L. C., & Holmström, J. (2019). Digital twin: vision, benefits, boundaries, and creation for buildings. *IEEE access*, 7, 147406-147419.
- Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future generation computer systems*, 29(7), 1645-1660.
- Dunleavy, M., & Dede, C. (2014). Augmented reality teaching and learning. *Handbook of research on educational communications and technology*, 735-745.
- Loijens, L. W., Brohm, D., & Domurath, N. (2017). What is augmented reality? In *Augmented reality for food marketers and consumers* (p. 356). Wageningen Academic Publishers.
- Zaher, M., Greenwood, D., & Marzouk, M. (2018). Mobile augmented reality applications for construction projects. *Construction Innovation*, 18(2), 152-166.
- Chi, H. L., Kang, S. C., & Wang, X. (2013). Research trends and opportunities of augmented reality applications in architecture, engineering, and construction. *Automation in Construction*, 33, 116-122.