

Article

Not peer-reviewed version

A Scaffolding Assembly Deficiency Detection System with Deep Learning and AR

[Ren-Jye Dzung](#)*, Chen-Wei Cheng, [Ching-Yu Cheng](#)

Posted Date: 29 December 2023

doi: 10.20944/preprints202312.2114.v1

Keywords: building scaffolding; safety detection; AI; deep learning; AR; HoloLens; BIM



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Article

A Scaffolding Assembly Deficiency Detection System with Deep Learning and AR

Ren-Jye Dzung ^{1,*}, Chen-Wei Cheng ² and Ching-Yu Cheng ³

¹ National Yang Ming Chiao Tung University; rjdzeng@nycu.edu.tw

² National Yang Ming Chiao Tung University; nokia3boss@gmail.com

³ School of Industrial Engineering, Purdue University; cheng640@purdue.edu

* Correspondence: rjdzeng@nycu.edu.tw;

Abstract: Scaffoldings play a critical role as temporary structures in supporting construction processes. Accidents at construction sites frequently stem from issues related to scaffoldings, including insufficient support caused by deviations from the construction design, insecure rod connections, or absence of cross bracing, which result in uneven loading and potential collapse, leading to casualties. In this study, we introduce a deep-learning-based, augmented reality (AR)-enabled system called scaffolding assembly deficiency detection system (SADDS) to assist field inspectors in identifying deficiencies in scaffolding assemblies. Inspectors have the flexibility to utilize SADDS through various devices, such as video cameras, mobile phones, or AR goggles (i.e., Microsoft HoloLens 2), for automated detection of deficiencies in scaffolding assemblies. The training test yielded satisfactory results, with a mean average precision of 0.89 and individual precision values of 0.96, 0.82, 0.90, and 0.89 for “qualified” frames and frames with the “missing cross-tie rod,” “missing lower-tie rod,” and “missing footboard” deficiencies, respectively. Field tests conducted at two construction sites demonstrated improved system performance compared to that in the training test. However, these field tests also revealed certain limitations of the SADDS system.

Keywords: building scaffolding; safety detection; AI; deep learning; AR; HoloLens; BIM

1. Introduction

Scaffolds are important as temporary structures for supporting workers, equipment, and materials during construction. The scaffolding assembly process is subject to strict safety regulations, including the inclusion of safe support systems, structural stability, and proper connection of connecting rods. Accidents on construction sites are often related to scaffolding, such as insufficient support owing to deviations from the construction design, insecure rod connections, or absence of cross bracing, which leads to uneven loading and collapse. The scaffolding assembly process requires inspectors to check for the presence of tie rods, cross-bracing rods, and base plates, in addition to ensuring that these components are properly secured. While some of these inspections can be performed visually, others may require inspectors to use touch, additional measuring tools, or other auxiliary instruments. In high-rise or large buildings, hundreds of structural scaffolding frames may be used, and it would be time-consuming for inspectors to check each frame even visually. Moreover, inspectors would very likely miss a few of the deficiencies in these frames.

In recent years, artificial intelligence (AI) has been widely used for image recognition at construction sites. In particular, deep learning models have significantly driven the uptake of AI for site monitoring and inspection. For example, Li et al. [1] applied a deep learning algorithm to detect concealed cracks from ground penetrating radar images. Fang et al. [2] used deep learning to detect construction equipment and workers on a construction in an attempt to create a safer work environment through real-time monitoring. Reja et al. [3] used deep learning to track construction progress by analyzing images and videos to enhance project management activities and decision-making. Shanti et al. [4] demonstrated the use of deep learning for recognizing safety violations at

construction sites. Wang et al. [5] applied deep learning to detect and quantify cracks on surfaces of concrete structures.

Since Microsoft released HoloLens (HL) and HoloLens2 (HL2) [6], mixed-reality (MR) headsets, they have been used in various industries, including the construction industry. The newer HL2 is equipped with a computational platform, which allows for on-device data processing and execution of AI applications. Moreover, it is equipped with multiple RGB and infrared cameras, which allow for spatial awareness and position tracking of the surrounding environment. Users can interact with the device manually, through eye-tracking, or by using voice commands to place AR (augmented reality) models. Meanwhile, in terms of project visualization, the construction industry has undergone a transformative shift in recent years owing to the widespread adoption of Building Information Modeling (BIM) [7]. BIM is a three-dimensional (3D) digital representation of the physical and functional aspects of a construction project, providing a comprehensive view of the project's lifecycle, including the construction phase.

Park et al. [8] conducted a comprehensive review of the academic applications of HL across diverse domains, encompassing medical and surgical aids, medical education and simulation, industrial engineering, as well as architecture and civil engineering. In a notable example, Pratt et al. [9] employed HL to assist medical surgeons in accurately and efficiently locating perforating vessels, leveraging information extracted from preoperative computed tomography angiography images. Additionally, Al-Maeeni et al. [10] utilized HL to guide machine users in executing tasks in the correct sequence, thereby optimizing retrofit time and cost when remodeling production machinery.

In the fields of architecture and civil engineering, HoloLens exhibits a real-time inside-out tracking capability, enabling precise visualization of virtual elements within the spatial environment. However, it necessitates a one-time localization of the augmented reality (AR) platform within the local coordinate frame of the building model to integrate indoor surroundings with the corresponding building model data. Consequently, research has delved into fundamental spatial mapping utilizing digital models and visualization techniques (e.g., [11,12]). In the realm of construction management, Mourtzis et al. [13] utilized HoloLens to visualize production scheduling and monitoring, while Moezzi et al. [14] concentrated on simultaneous localization and mapping (SLAM) for autonomous robot navigation, leveraging HoloLens to facilitate control over positioning, mapping, and trajectory tracking. This study specifically addresses deficiencies observed when employing AR in construction. Karaaslan et al. [15] developed an MR framework integrated with an HL headset to assist bridge inspectors by automatically analyzing defects, such as cracks, and providing real-time dimension information along with the condition state.

Although the guidelines on the safety of scaffolding have been studied (e.g., [16]), only a few researchers have conducted digitalization-related research specifically on scaffolding. For example, Chan-woo Baek [17] focused on improving transparency, accountability, and traceability in construction projects by applying blockchain technology to support a secure, decentralized ledger for documenting and verifying scaffolding installation processes. Sakhakarmi et. al. [18] developed a machine-learning model to classify cases of scaffolding failure in buildings and predicted safety conditions based on strain datasets of scaffolding columns spanning multiple bays and stories. Similarly, Choa et. al.[19] developed an Arduino module to build an Internet of Things network for collecting the boundary conditions associated with the dynamic loading conditions of scaffolding structures. In addition, they used the finite element method to estimate the structural behavior of scaffolds in real time.

As Sakhakarmi et. al. [18] pointed out, although 65% of construction workers work on scaffolding structures and are often exposed to safety hazards, the existing method for monitoring scaffolding structures is inadequate. Despite regular safety inspections and safety planning, numerous fatal scaffolding-related accidents continue to occur at construction sites. The existing practices that rely on human inspection are not only ineffective but also unreliable owing to the dynamic nature of construction activities [19]. In this research work, we integrate a machine deep learning model with an AR model by using the HL2 as the main visual device to help superintendents

perform visual inspections in conformance with the regulations governing scaffolding for building facades during the construction phase.

According to the safety regulations for inspecting construction scaffolding in Taiwan [20], the inspection requirements related to construction sites are divided into three types, namely, visual inspection, measurement inspection, and strain monitoring. We use this classification to define the scope of this research. For example, one can visually inspect and determine compliance with Article 4, which stipulates that cross-tie rods and lower-tie rods should be installed on both sides of scaffolding. One must perform measurements to determine compliance with Article 9, which stipulates that the distance between two scaffolding wall poles should be less than 5.5 m in the vertical direction and 7.5 m in the horizontal direction. In a few special dynamic situations, stress and strain gauges must be installed.

In this work, we focus only on the automation of visual inspection, which does not require measurement or installation of stress and strain gauges and is easy to implement at construction sites. A few examples are as follows.

Article 4: "Cross-tie rods and lower-tie rods should be installed on both sides of the scaffolding," and "There should be appropriate guardrails on the stairs going up and down the scaffolding."

Article 6: "Brackets should be used to lay auxiliary pedals or long anti-fall nets between the scaffolding and the structure."

Article 12: "The scaffolding should have a foundation plate that is securely placed on the ground."

2. Materials and Methods

2.1. Problem Statement and Materials

We focus on automating the inspection of deficiencies in scaffolding assembly, that is, the process of checking for missing parts in the structure of a scaffolding frame. Table 1 presents an example of a qualified scaffolding assembly, in which the columns and beams of the scaffolding frame are complete with appropriate cross-tie and lower-tie rods, as well as a footboard. In addition, Table 1 presents examples of scaffolding assemblies with different types of deficiencies, for instance, "missing cross-tie rod," "missing lower-tie rod," and "missing footboard." The color and number under each type of deficiency represent the highlight color and number of frames collected for use in the training phase of the deep learning model, which will be described next in the implementation section.






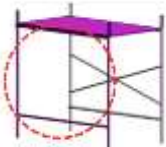
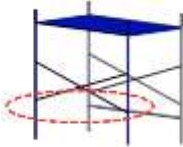

A total of 408 photos were acquired from 12 construction sites of different concrete residential building projects. As a precautionary measure for safety, these photos were captured from vantage points situated outside the exterior walls of the structures. In addition, the photos were taken at varying intervals throughout the day to mitigate potential biases. Each of the photos encompasses multiple scaffolding frames, featuring either compliant frames or deficiencies as delineated in Table 1.

Each photo comprises multiple scaffolding frames with qualified frames or deficiencies described in Table 1. The photos were further annotated by framing each of the image areas containing qualified frames or frames with different types of deficiencies as the target features and attaching labels (i.e., different integers and colors representing "missing cross-tie rod" or "missing lower-tie rod" and so on) (Figure 3). Table 1 shows the highlight colors and the numbers of labeled frames for each type of target feature in the original photos.

To augment the original dataset, we used Roboflow (specifically using the auto-orient and resize functions) [21], a web platform that helps developers to build and deploy computer vision models, to expand the initial photo collection to a total of 2226 images. Subsequently, these expanded photos were randomly partitioned into training, validation, and test datasets, constituting 80%, 10%, and 10% of the total, respectively (4,240 images for training, 530 for validation, and 530 for testing). The training and validation datasets are earmarked for the development and training phases of the

proposed system, while the test datasets are exclusively reserved for evaluating the system post-development.

Table 1. Comparison between qualified scaffold assembly and those with deficiencies.

Qualified	Missing cross-tie rod	Missing lower-tie rod	Missing footboard
			
			
green (763)	magenta (245)	purple (575)	red (643)

2.2 Scaffolding Assembly Deficiency Detection System

Figure 1 presents a conceptual model of the proposed scaffolding assembly deficiency detection system (SADDs), which helps field inspectors identify deficiencies in scaffolding assemblies. The building icons in the top center part of the image represent the real world, in which scaffoldings with predetermined QR-code markers attached are present. To capture the image streams of these scaffolding, one may use a video camera or mobile phone or wear AR goggles (e.g., HL2). SADDs takes the captured scaffolding images as the input and generates highlights of assembly deficiencies on the images as the output. The input images can be processed in two ways depending on the image capture device used. The dashed lines in Figure 1 represent the processing flow when a user captures images using a mobile phone or video camera, sends the video stream to a web server, and views the highlights (output) directly on the web.

The other way, represented by solid lines in Figure 1, depicts the processing flow when the user wears an HL2 to capture images and view the highlights on the HL2 itself. This process is more complex because it involves the integration of real-world images and digital 3D models in Unity. The process comprises three mandatory functional modules and one optional module. First, the image stream is sent to the recognition module, which uses a deep learning model to identify the types of deficiencies in the scaffolding assembly captured in the images and highlights each deficiency by using a differently colored frame. The highlighted image stream is then sent to the AR visualization module, which uses the QR-code markers placed in the real world to reposition and reorient the 3D digital model (i.e., export of BIM to Unity) accordingly. Finally, the HL2 visualization module is used to present the highlighted AR images on the headset display. The optional function is the annotation module, which records the highlights representing the deficiencies along with the corresponding elements of the 3D model when necessary.

The incorporation of recognition functions into AR using HL offers a distinct advantage due to the availability of development tools in the market. Utilizing these tools equips developers with Application Interface Protocols (API), facilitating the seamless overlay of a 3D digital model onto real-world images through pre-determined markers. This integration alleviates concerns about the intricate positioning and orientation of the model as the HL wearer moves. When inspectors wish to annotate deficiencies following the identification of highlighted issues, the API additionally enables the use of hand gestures to mark the 3D model effectively.

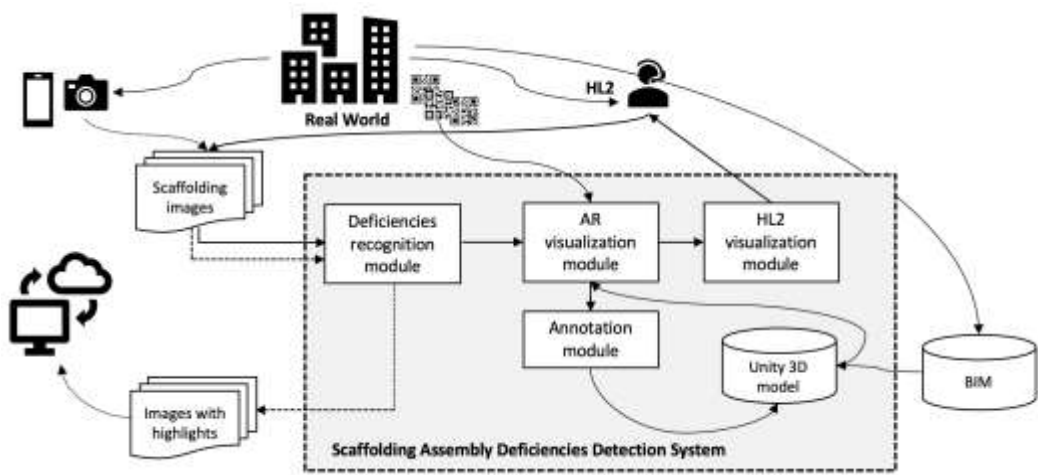


Figure 1. Conceptual model of Scaffolding Assembly Deficiencies Detection System.

2.3. Implementation Methods

This section describes the methods and the implementation of the main modules of the proposed model, including the deficiency recognition, AR visualization, and HL2 visualization modules.

2.3.1. Deficiency-Recognition Module

We use a deep machine learning model to automatically recognize assembly deficiencies in scaffoldings. Figure 2 depicts the process of establishing a pre-trained model for recognizing deficiencies in scaffolding assemblies. The boxes on the left represent the process of preparing the training data and selecting a suitable deep learning algorithm, including collecting photos of both qualified and unqualified scaffolds, labeling scaffolding frames accordingly, developing a machine learning model, and testing the feasibility of deploying the developed algorithm on the HL2 platform. After the algorithm is developed, a machine learning model is built using Python, and the collected photos are randomly divided into training, validation, and test sets. The total of 4,240 images and 530 images are used to train and validate the deep learning model.

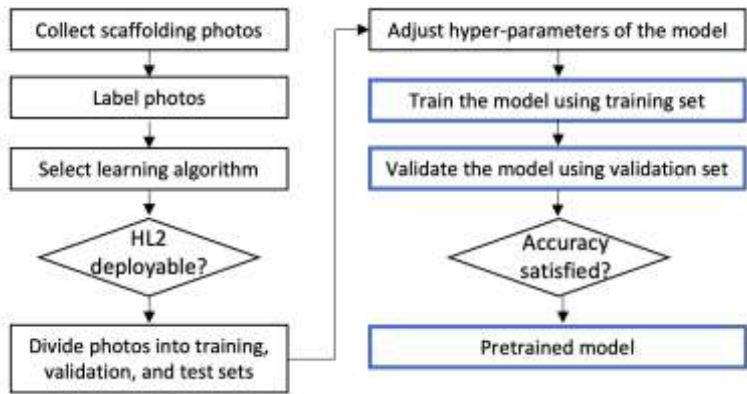


Figure 2. Pretraining process of deep learning model in deficiency-recognition module.

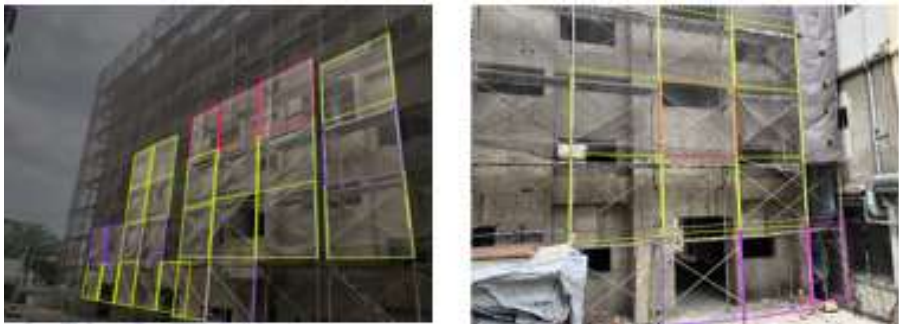


Figure 3. Examples of frame labeling in photos.

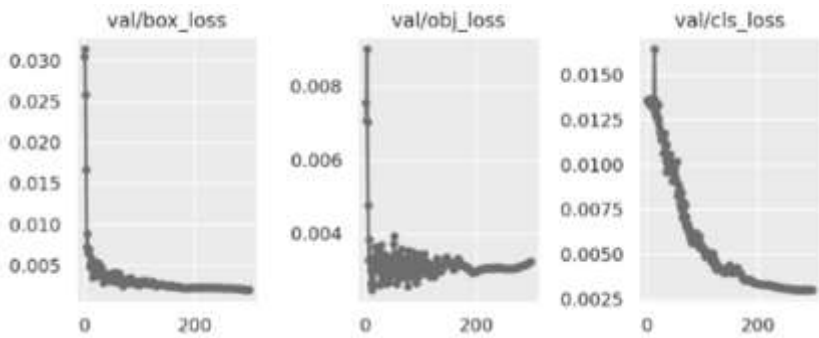


Figure 4. Validation losses.

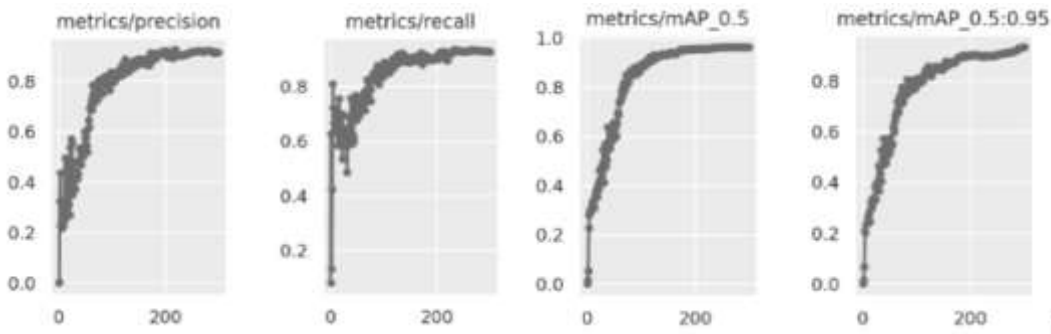


Figure 5. Precision, recall, and mAP in the validation phase.

The training process of deep learning includes setting up of the deep learning model with initial hyperparameters, model training, and model validation. The training process may require iterations with different hyperparameters until satisfactory accuracies are obtained. In Roboflow, MaskRCNN [22] was initially selected to train the developed model, but this trained model could not be installed on HL2. Subsequently, we used YOLO v5 [23] to train the developed model.

2.3.2. AR Visualization Module

Four steps were followed to use markers in the real world for integrating images of the real world with the 3D model in Unity and, subsequently, project the images to HL2.

Step 1: Set up markers

We used the Vuforia Engine [16] to integrate AR, Unity, and Microsoft’s Mixed-Reality Toolkit (MRTK) (for hand gesture recognition and other purposes). The Vuforia HoloLens 2 Sample, provided by Unity in the Standard Unity Asset Store [25], with a pre-configured Unity scene and project settings, was used during development as a starting point for a customized application. One can choose from various types of markers available on the Vuforia platform with different complexity

ratings. The more complicated the markers, the easier and more accurate they are from the recognition perspective, but more time is required to recognize them.

Step 2: Implement and initialize markers in Unity

To implement and initialize markers in Unity, the following highlights the required sub-steps.

1. Switch to the *Universal Windows Platform* setup page by choosing *Build Settings/Platform/Universal Windows Platform*.
2. Setup the initialization parameters by choosing *Project Settings/XR Plug-in Management/Windows*, and activate *Initialize XR on Startup*.
3. Specify the XR device by choosing *Project Settings/XR Plug-in Management/Windows*, and activate *OpenXR* and *Microsoft HoloLens feature group*.
4. Add the mobile control module by adding *Object Manipulator Script*.
5. Add the hand gesture recognition module by choosing *Project Settings/OpenXR/Interaction Profiles*, and selecting “*Eye Gaze Interaction Profile*,” “*Microsoft Hand Interaction Profile*,” and “*Microsoft Motion Controller Profile*” (Figure 6).
6. Activate the hand recognition module by choosing *Project Settings/OpenXR/OpenXR Feature Groups* and activating Microsoft HoloLens’ “*Hand Tracking*” and “*Motion Controller Model*.”

We used different color highlights to represent different types of deficiencies in scaffolding assemblies. Half-transparent color blocks were created based on the results of the recognition module and overlaid on the Unity model.

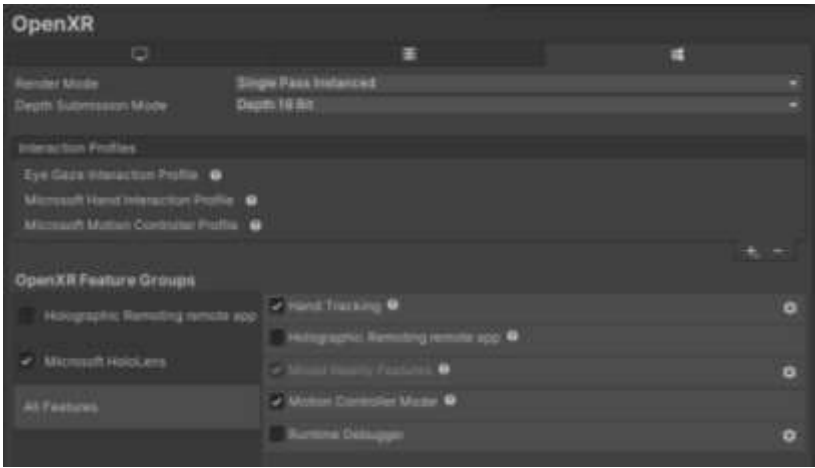


Figure 6. Adding hand gesture recognition module in Unity.

2.3.3. HL2 Visualization Module

Pre-configuration was required to use the Unity model on the HL2 platform. To this end, the developer mode in HL2 was activated, and the exact *WiFi IP* URL used when developing the Unity model in Microsoft Visual Studio was entered [26]. Moreover, at the construction site, QR-code markers were placed in accordance with the predetermined marker setup in the Unity model. By using these markers and the screen captured from the real world, the Unity model was re-positioned and re-oriented.

3. Results

Among the three main modules, i.e., deficiency recognition, AR visualization, and HL2 visualization modules, the deep learning model of the deficiency recognition module is the one that determines the recognition accuracy of SADDs.

During the training phase, the mean average precision (mAP) of the trained model was 0.951, precision was 88.3%, recall was 90.4%, and F1 score was 0.893 after 166 epochs. Table 2 lists the precision values of the model trained using Roboflow.

Table 2. Mean average precision values of the model trained using Roboflow.

	mAP	Qualified	Missing cross-tie rod	Missing lower-tie rod	Missing footboard
Validation	0.94	0.97	0.96	0.937	0.93
Test	0.88	0.95	0.80	0.90	0.88

Because we did not have access to the codes of the models trained in the Roboflow software environment, we recreated the YOLO v5 version of the model by using the pytorch package of Python, where we set batch = 16, epoch = 300, and learning_rate = 0.01. The expanded image dataset was used to train and test this model. Table 3 shows the mAP and the corresponding precisions of this self-built model for the “Qualified,” “Missing cross-tie rod,” “Missing lower-tie rod,” and “Missing footboard” classes. The test mAP was 0.89, with precision values of 0.96, 0.82, 0.90, and 0.89 for “Qualified,” “Missing cross-tie rod,” “Missing lower-tie rod,” and “Missing footboard” types of deficiencies, respectively. Table 4 and Figure 4 summarize and illustrate the losses of this model during the validation phase. Figure 5 depicts the convergence of precision, recall, and mAP in the validation phase. The precision data indicate that the results obtained using the trained model were satisfactory. The trained model was then used as the deficiency-recognition module in SADDs.

Table 3. Mean average precision values of the self-built trained model.

	mAP	Qualified	Missing cross-tie rod	Missing lower-tie rod	Missing footboard
Validation	0.96	0.98	0.979	0.90	0.96
Test	0.89	0.96	0.82	0.90	0.89

Table 4. Losses of self-built trained model.

	Box loss	Object loss	Class loss
Validation	0.0020	0.0032	0.0030
Test	0.0021	0.0041	0.0037

The visualization of the result of the deficiency recognition module depends of viewing devices. As described previously in the conceptual model of Figure 1, there are two ways of using SADDs, i.e., with or without HL2 AR goggle, to help a user to check scaffolding frames. When a user captures images using a mobile phone or video camera, and sends the video stream to a web server, the highlights are viewed directly on the web as shown in the left image of Figure 7.

When a user uses HL2 AR goggle, the AR visualization module synchronizes the real world and the digital Unity model based on the QR-code markers. The center image of Figure 7 shows the example of highlights projected in the Unity model. Subsequently, the HL2 visualization module projects the highlights on HL2 as shown in the right image of Figure 7.

The visualization of the outcomes from the deficiency recognition module is contingent upon the viewing devices employed. As expounded in the conceptual model depicted in Figure 1, the utilization of SADDs manifests in two modes: with or without the use of HL2 AR goggles, facilitating users in inspecting scaffolding frames. When users capture images through a mobile phone or video camera and transmit the video stream to a web server, the resulting highlights are directly perceivable on the web interface, as depicted in the left panel of Figure 7.

Conversely, when users opt for the HL2 AR goggles, the AR visualization module synchronizes the real-world environment and the digital Unity model based on QR-code markers. The central panel of Figure 7 provides an example of highlights projected onto the Unity model. Subsequently, the HL2 visualization module projects these highlights onto the AR goggles, as exemplified in the right panel of Figure 7.

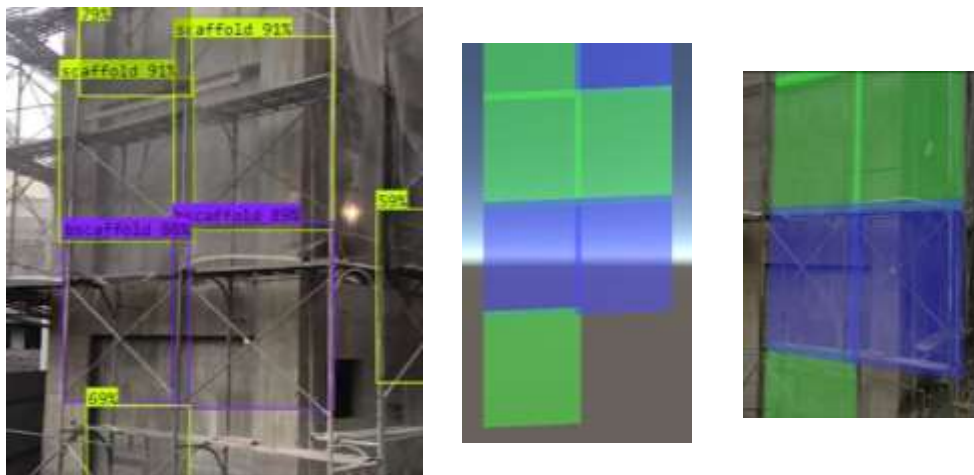


Figure 7. Comparison of color highlights by YOLO (left), Unity model (center), and HL2 (right).

4. Field Test and Discussion

To field test the SADDs, we deployed it at two other construction sites, namely, 7- and 14-story concrete residential buildings, in Hsinchu City, Taiwan. One of the authors wore the HL2, walked slowly, and recorded the front and rear facades of these under-construction buildings from the exterior on the ground. The weather was cloudy and occasionally drizzly, which did not significantly affect image quality.

Figures 8 and 9 present examples of the recognition results obtained at these two test sites. Automated detection of the target deficiencies worked successfully, and most of the scaffolding frames were found to be qualified (green label), twelve had “missing lower-tie rods” (purple), two had “missing cross-tie rods” (magenta), and one had a “missing footboard” (red). The following lessons were learned from the field tests.

- The camera shooting angle should be as orthogonal to the target wall face as possible. Nonetheless, the recognition module successfully recognized the deficiencies in some frames sooner or later as the wearer approached those frames. However, because the attached alert frames are always orthogonal squares, the highlights may cause humans to misread the wrong frame. This problem can be avoided so long as the camera shooting angle is orthogonal to the target wall face.
- When shooting at an oblique angle with respect to the target wall face, far away frames may not be recognized by the module owing to self-occlusion. This problem is understandable because even humans cannot evaluate those frames in the same situation, and those frames will be evaluated correctly once the camera moves toward them in the absence of occlusions.
- Considering the practical use case, to enhance work efficiency and inspector’s safety, the tests were performed in front of scaffolds on the ground without actually climbing on the scaffolding boards to efficiently capture multiple frames at a glance. So long as the shooting angle is near orthogonal to the target wall face, an image with 20–50 frames did not seem to be a problem to SADDs. In this way, the use of SADDs is more efficient than inspection with the human eye. Nevertheless, in double-frame scaffold systems, most of the inner frames will not be recognized by the system owing to self-occlusion by the outer frames. Although one may stand on a scaffolding board to shoot the inner frames without occlusion, the number of frames covered in an image would be very limited, and the frames would need to be checked one by one. In such a case, direct inspection with human eyes would be more convenient.
- Before the field test, we were concerned about the system’s ability to recognize “missing cross-tie rod,” which had the least precision (i.e., 0.82 compared to, for example, 0.90 for “missing lower-tie rod”) among the three types of target deficiencies. However, this did not seem to be a problem during the field test. A possible explanation is that in the training test, precision values were calculated per image, and each misidentification was counted. However, during actual field use, images were run as a stream, and when the HL2 wearer was moving, SADDs had many chances to successfully identify deficiencies and, eventually, alert the wearer.

- The scaffolds at both test sites were enclosed by safety nets (e.g., anti-fall or dustproof nets), which did not affect the recognition accuracy of SADDs so long as human eyes could see through the net. Indeed, in the presence of safety nets, it was more difficult for humans to recognize unqualified assemblies from a distance than it was for SADDs.

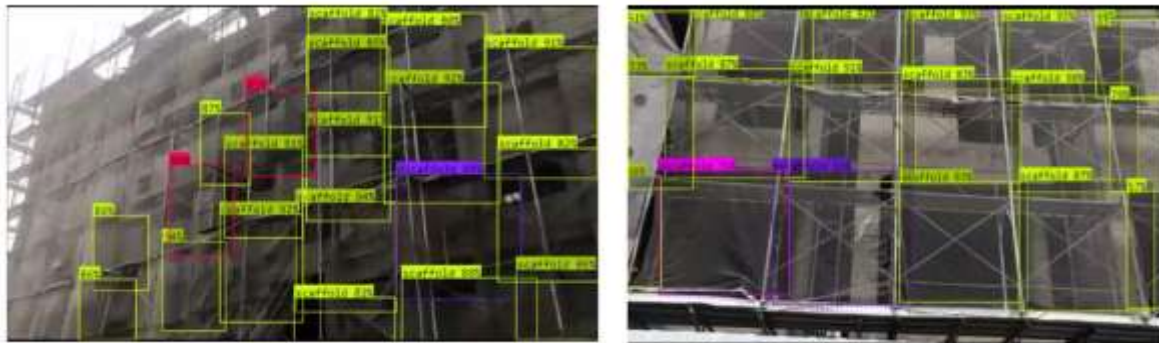


Figure 8. Examples of images obtained in field test involving 7-story building



Figure 9. Examples of images obtained in field test involving 14-story building.

The auto-generated highlights on HL2 provide its wearer real-time warnings related to the frames of unqualified assemblies, and the wearer can suggest remedial measures right away. To record such warnings, it is best to annotate the highlight on the corresponding elements on the 3D Unity model, and if necessary, export it back to the .ifc BIM format so that it can be read using BIM-compatible software, such as Revit. Figure 10 shows the recorded highlights of unqualified scaffolding assemblies on the corresponding elements of the Revit model.

Professionals at the test sites appreciated the real-time highlighting of unqualified scaffolding frames. This function helped inspectors to avoid missing any potential deficiencies in scaffolding assemblies even when they only glanced at such assemblies. However, they were not convinced about the need to record the highlights of unqualified frames on the 3D model. First, scaffolds are temporary structures that may be adapted frequently as the construction proceeds. Recording only a snapshot of such a dynamic temporary structure did not make sense to them. Second, A/E or construction contractors seldom implement scaffolding elements in BIM. Creating a scaffolding structure in a 3D model simply for annotating the deficiencies in the former did not seem a worthwhile endeavor to them. Note that we created the scaffolding elements manually simply for the purpose of this study.

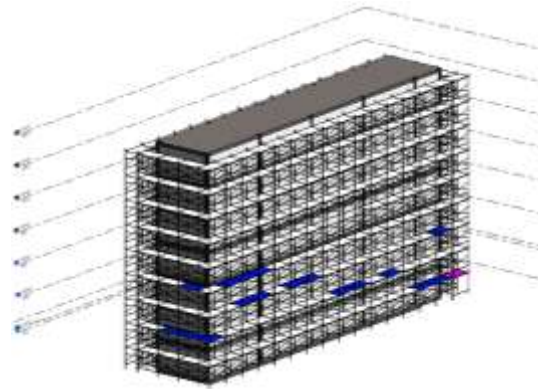


Figure 10. Recorded highlights on the corresponding elements in BIM.

Finally, there is a multitude of deficiencies in construction scaffolding. When training deep learning models, it is essential to compile a diverse set of cases that represent various deficiency patterns. This article specifically addresses the types of deficiencies outlined in Table 1. For instance, to train the model to recognize deficiencies such as missing cross-tie rods, lower-tie rods, or footboards, engineers can capture such types of deficiencies in the scaffolding's front view from the side, and train the model simultaneously. The simultaneous training of these three deficiency types is feasible because they share the same appropriate camera-shooting angles and could potentially coexist within the same captured image.

Conversely, identifying deficiencies in tie rods and fall protection nets necessitates engineers to position themselves either on the scaffolding or in the gap between the scaffolding and the building for optimal shooting. To spot deficiencies in the scaffolding's base plate, engineers should focus on the scaffold's bottom and, if necessary, capture images as close and vertically as possible toward the base plate. When identifying metal fastener deficiencies, engineers may need to concentrate on the joints between components, capturing a range of both qualified and unqualified patterns. We recommend training the model to recognize these deficiency types separately since the appropriate camera-shooting angles for each are distinct, and they rarely coexist in a single image.

5. Conclusions

Scaffolds are important as temporary structures that support construction processes. Accidents at construction sites are often related to scaffolds, such as insufficient support owing to deviations from the construction design, insecure rod connections, or absence of cross bracing, which lead to uneven loading and collapse, thus resulting in casualties. Herein, we proposed a deep-learning-based, AR-enabled system called SADDs to help field inspectors identify deficiencies in scaffolding assemblies. An inspector may employ a video camera, mobile phone, or wear AR goggles (e.g., HL2) when using SADDs for automated detection of deficiencies in scaffolding assemblies. The test mAP during training was 0.89, and the precision values for the "qualified," "missing cross-tie rod," "missing lower-tie rod," and "missing footboard" cases were 0.96, 0.82, 0.90, and 0.89, respectively. The subsequent field tests conducted at two construction sites yielded satisfactory performance in practical use cases. However, the field tests highlighted a few limitations of the system. For example, the system cannot be used to identify the deficiencies in the inner wall of a double-walled scaffolding structure from the outside owing to occlusions. In addition, the detection of deficiencies in assemblies is merely the beginning of scaffolding inspection. Many unqualified scenarios that can be visually identified have not been covered in this work (e.g., deformed frames, and metal fasteners). These scenarios must be addressed in future work. In addition, the detection of these scenarios may require video-shooting of the scaffolding frames from a close distance, which leads to the following efficiency issue: if an experienced inspector is willing to perform frame-by-frame inspection, why would they need to use SADDs? More recognition abilities will be developed and the related issues will be explored in future research.

Funding: This research was funded by National Science and Technology Council, Taiwan, grant number 108-2221-E-009 -018 -MY3, 112-2918-I-A49 -004, and 111-2221-E-A49 -040 -MY3.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Li, S.; Gu, X.; Xu, X.; Xu, D.; Zhang, T.; Liu, Z.; Dong, Q. Detection of concealed cracks from ground penetrating radar images based on deep learning algorithm. *Construction and Building Materials* **2021**, Vol. 273, pp.121949. <https://doi.org/10.1016/j.conbuildmat.2020.121949>.
2. Fang, W.; Ding, L.; Zhong, B.; Love, P.E.; Luo, H. Automated detection of workers and heavy equipment on construction sites: a convolutional neural network approach. *Advanced Engr. Informatics* **2018**, Vol. 37, p.139-149. <https://doi.org/10.1016/j.aei.2018.05.003>.
3. Reja, V.K.; Varghese, K.; Ha, Q.P. Computer vision-based construction progress monitoring. *Automation in Construction* **2022**, Vol. (138), pp.104245. <https://doi.org/10.1016/j.autcon.2022.104245>.
4. Shanti, M.Z.; Cho, C.S.; Byon, Y.J.; Yeun, C.Y. A novel implementation of an AI-based Smart construction safety inspection protocol in the UAE. *IEEE Access* **2021**, Vol. 9, pp. 166603-166616, doi: 10.1109/ACCESS.2021.3135662.
5. Park, S.E.; Eem, S.H.; Jeon, H. Concrete crack detection and quantification using deep learning and structured light. *Construction and Building Materials* **2020**, Vol. 252, p. 119096, <https://doi.org/10.1016/j.conbuildmat.2020.119096>.
6. Microsoft (2023). "HoloLens 2 release notes," <https://learn.microsoft.com/en-us/hololens/hololens-release-notes#about-hololens-releases>, last updated on 10/11/2023.
7. Leite, F.; Cho, Y.; Behzadan, A.H.; Lee, S.H.; Choe, S.; Fang, Y.; Akhavian, R.; Hwang, S. Visualization, information modeling, and simulation: grand challenges in the construction industry. *J. of Computing in Civil Engr.* **2016**, 30(6), [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000604](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000604).
8. Park, S.; Bokijonov, S.; Choi, Y. Review of Microsoft HoloLens applications over the past five years. *Applied Sciences* **2020**, Vol. 11(16), p. 7259, <https://doi.org/10.3390/app11167259>.
9. Pratt, P., Ives, M., Lawton, G. et al. Through the HoloLens looking glass: augmented reality for extremity reconstruction surgery using 3D vascular models with perforating vessels. *Eur Radiol Exp* **2**, 2 (2018). <https://doi.org/10.1186/s41747-017-0033-2>
10. Al-Maeni, Sara S. H., Kuhnhen C., Engel B.; Schiller M. Smart retrofitting of machine tools in the context of industry 4.0. *Procedia CIRP* **2019**, Vol. 88, pp. 369-374, <https://doi.org/10.1016/j.procir.2020.05.064>.
11. Hübner, P.; Clintworth, K.; Liu, Q.; Weinmann, M.; Wursthorn, S. Evaluation of HoloLens tracking and depth sensing for indoor mapping applications. *Sensors* **2020**, Vol.20(4), pp.1021. <https://doi.org/10.3390/s20041021>.
12. Wu, M.; Dai, S.-L.; Yang, C. Mixed reality enhanced user interactive path planning for omnidirectional mobile robot. *Appl. Sci.* **2020**, Vol.10, pp.1135. <https://doi.org/10.3390/app10031135>.
13. Mourtzis, D., Siatras, V., and Zogopoulos, V. Augmented reality visualization of production scheduling and monitoring. *Procedia CIRP* **2020**, Vol.88, 151-156. <https://doi.org/10.1016/j.procir.2020.05.027>.
14. Moezzi, R.; Krcmarik, D.; Hlava, J.; Cýrus, J. Hybrid SLAM modeling of autonomous robot with augmented reality device. *Mater. Today Proc.* **2020**, Vol.32, pp.103–107. <https://doi.org/10.1016/j.matpr.2020.03.036>.
15. Karaaslan, E., Bagci, U., & Catbas, F. N. (2019). Artificial intelligence assisted infrastructure assessment using mixed reality systems. *Transportation Research Record* **2019**, Vol.2673(12), pp.413-424. <https://doi.org/10.1177/0361198119839988>.
16. Sanni-Anibire, M.O.; Salami, B.A.; Muili, N. A framework for the safe use of bamboo scaffolding in the Nigerian construction industry. *Safety Science* **2022**, Vol. 151, p. 105725. <https://doi.org/10.1016/j.ssci.2022.105725>.
17. Baek, C.W.; Lee, D.Y.; Park, C. S. Blockchain based Framework for Verifying the Adequacy of Scaffolding Installation. In Proc. of the 37th ISARC (Int. Symposium on Automation and Robotics in Construction), Vol. 37, pp. 425-432, 2020, Kitakyushu, Japan, IAARC Publications.
18. Sakhakarmi, S.; Park, J.W.; Cho, C. Enhanced machine learning classification accuracy for scaffolding safety using increased features. *J. of Construction Engr. and Mgmt* **2019**, Vol. 145(2): 04018133 pp.1-9. [https://doi.org/10.1061/\(ASCE\)CO.1943-7862.0001601](https://doi.org/10.1061/(ASCE)CO.1943-7862.0001601).
19. Choa, C.; Sakhakarmi, S.; Kim, K.; Park, J.W. Scaffolding Modeling for Real-time Monitoring Using a Strain Sensing Approach. In Proc. of 35th ISARC (Int. Symposium on Automation and Robotics in Construction), Berlin, Germany, 2018, pp. 48-55. <https://doi.org/10.22260/ISARC2018/0007>.
20. Ministry of Labor of Taiwan, *Safety Regulations for Inspecting Construction Scaffolding*. <https://laws.mol.gov.tw/FLAW/FLAWDAT01.aspx?id=FL083843> (in Chinese), 2018.
21. Roboflow, Inc. (2023), roboflow official site, <https://roboflow.com/>.
22. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proc. of 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017, pp. 2980-2988, doi: 10.1109/ICCV.2017.322

23. Jocher, G.; Stoken, A.; Borovec, J.; NanoCode012; ChristopherSTAN; Changyu, L.; Laughing, tkianai, yxNONG, Hogan, A., et al. (2021). "Ultralytics/yolov5: v4.0 – nn.SiLU() activations, weights & biases logging, PyTorch hub integration," DOI: 10.5281/zenodo.4418161
24. PTC, "Vuforia Engine Developer's Portal" (2023), '<https://developer.vuforia.com/>
25. Unity (2023), "Vuforia Hololens 2 Sample", <https://assetstore.unity.com/packages/templates/packs/vuforia-hololens-2-sample-101553>
26. Microsoft Inc. (2023) 'GitHub Copilot and Visual Studio 2022', <https://visualstudio.microsoft.com/zh-hant/>

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.