Smart Structures and Systems, *Vol. 31, No. 6 (2023) 573-583*
https://doi.org/10.12989/sss.2023.31.6.573

Development of a structural inspection system with marking damage information at onsite based on an augmented reality technique

Junyeon Chung, Kiyoung Kim and Hoon Sohn*

Civil and Environmental Engineering, Korea Advanced Institute of Science and Technology, 34141, Daehak-to, Yuseong-gu, Daejeon, Republic of Korea

(*Received March 27, 2023, Revised May 16, 2023, Accepted May 26, 2023*)

Abstract. Although unmanned aerial vehicles have been used to overcome the limited accessibility of human-based visual inspection, unresolved issues still remain. Onsite inspectors face difficulty finding previously detected damage locations and tracking their status onsite. For example, an inspector still marks the damage location on a target structure with chalk or drawings while comparing the current status of existing damages to their previous status, as documented onsite. In this study, an augmented-reality-based structural inspection system with onsite damage information marking was developed to enhance the convenience of inspectors. The developed system detects structural damage, creates a holographic marker with damage information on the actual physical damage, and displays the marker onsite via an augmented reality headset. Because inspectors can view a marker with damage information in real time on the display, they can easily identify where the previous damage has occurred and whether the size of the damage is increasing. The performance of the developed system was validated through a field test, demonstrating that the system can enhance convenience by accelerating the inspector's essential tasks such as detecting damages, measuring their size, manually recording their information, and locating previous damages.

Keywords: augmented reality; damage detection; damage management; mixed reality; structural inspection

1. Introduction

Managing the increasing number of civil infrastructure projects has become a pressing issue in several countries. In South Korea, for example, up to 270,000 structures require annual inspection, despite shrinking budgets and manpower (Kim and Cho 2020). In addition, according to a 2017 report by the American Society of Civil Engineering, 39% of all US bridges are over 50 years old, and the total bridge maintenance cost is estimated to be $123 billion (Kim and Cho 2020).

As structural safety concerns have increased, the need for effective management has emerged. Consequently, an integrated structural management system was proposed. The structural management system includes (1) periodic visual inspection, (2) structural condition assessment, (3) appropriate repair method proposals, and (4) residual life assessments (Dai *et al*. 2017). In particular, the correctness of periodic visual inspections should be ensured because all subsequent steps are based on the results of the inspection.

In the periodic visual inspection step, a human-oriented inspection method is most popular because of its usability and reliability (Falorca *et al*. 2021). An expert inspector visually examines structural damage such as corrosion, spalling, exposed rebar, efflorescence, and cracks, measures their sizes, and photographs them. The inspector then marks their locations on the drawing document and sometimes on the structure with chalk, so that they can easily find the location of the previously detected damage during subsequent inspections. However, a labor effectiveness problem has been indicated because these manual processes require considerable time, especially for finding previously detected damages, examining their status in subsequent inspections, and recording the damage status of documents. In addition, this method is inconsistent because all damage detection and size measurements are based on an inspector's subjective judgment.

Various inspection methods have been introduced to address the inefficiency and inconsistency of human-oriented visual inspection. Particular attention has been paid to unmanned aerial vehicles (UAVs) to overcome the limited accessibility of human-oriented visual inspections. Kim *et al*. (2018) used a UAV and deep learning to detect cracks in large civil structures such as bridges (Kim *et al*. 2018); however, the technique could not be applied indoors because a global positioning system (GPS) cannot work indoors. Ali *et al*. (2021) developed an autonomous UAV system to detect corrosion, cracks, and loosened bolts using a deep learning technique. This system can be applied indoors because it utilizes ultrasonic beacons instead of GPS (Ali *et al*. 2021). Although this technique demonstrated the possibility of indoor inspection with UAV, it was still difficult to apply it in a complex indoor area with many obstacles.

Recently, the effectiveness of augmented reality (AR) headsets in visual inspection has been explored to improve the convenience of inspectors. Karaaslan *et al*. (2019) used

*Corresponding author, Ph.D., Professor,
E-mail: hoonsohn@kaist.ac.kr

ISSN: 1738-1584 (Print), 1738-1991 (Online)

a deep-learning-based damage-detection technique to display the detection results on an onsite inspector's AR headset–(Karaaslan *et al*. 2019). Wang *et al*. (2021) also used an AR headset to automatically detect damage with box representation using a deep-learning technique and visualized the results (Wang *et al*. 2021). Mascarenas *et al*. (2020) studied the use of AR techniques to measure crack lengths on pavement area (Mascarenas *et al*. 2020). These studies described a novel possibility of using AR headsets in structural damage inspection by visualizing the damage detection results to onsite inspectors. However, these studies only focused on improving the inspector's convenience during a single inspection and did not investigate how the AR technique assists inspectors during subsequent inspections. Furthermore, Maharjan *et al*. (2020) enabled the real-time visualization of sensor data on a holographic panel, and Nguyen *et al*. (2021) developed a system to visualize holographic 3D BIM models with damage information (Maharjan *et al*. 2020, Nguyen *et al*. 2021). These studies might help the inspector intuitively understand the status of the structure, but did not improve the inspection efficiency or inspector convenience.

Several researchers have attempted to solve the limitations of human-oriented visual inspection, but there are still problems to be solved: 1) In subsequent inspection, onsite inspectors have difficulty finding the previously detected damages only by drawing documents indicating the damage locations, 2) onsite inspectors cannot check the previous damage status and track the damage history, and 3) a vision-based deep learning algorithm cannot detect structural damages onsite perfectly yet.

In this study, a structural inspection system with marking damage information onsite based on an AR headset was proposed to improve the convenience of the onsite inspector in finding previously detected damage locations during subsequent inspections, efficiency in tracking damage history, and damage detection accuracy of the deep learning-based method. Through the AR headset display, the system enables onsite inspectors to detect and track structural damage by visualizing damage information in the form of a hologram. The overall operation procedure of the proposed system is as follows. The proposed system automatically detects damage and displays the detection result to the onsite inspector. When the inspector confirms that the result is correct, information regarding the detected damage is saved, and a holographic marker is created and displayed at the precise location of the detected damage. On his/her subsequent visit to the site, the inspector can readily identify the location of the previously detected damage based on holographic markers and review the damage information via a holographic panel. In addition, all damaged images are normalized to be viewed from the front at a consistent distance despite being captured from different angles and distances.

The contribution of this paper is as follows:

(1) The proposed system assists the onsite inspector in locating previously detected damage by displaying a holographic marker at the precise location of the damage during subsequent inspection.
(2) An inspector can track the damage history, including the previous and current status of the damage onsite via a holographic panel and readily check the current state of the damage based on the normalized damage images.
(3) Various structural damages (such as spalling, exposed rebar, and efflorescence) can be clearly detected and quantified through human interaction using an AR headset by reducing false alarms.

The remainder of this paper is organized as follows. Section 2 describes the hardware configuration and detailed operating principle of the proposed system. In Section 3, the validation of the performance of the proposed system through field tests is described. Finally, the conclusions and discussion are presented in Section 4.

2. Proposed system

As depicted in Fig. 1, the proposed system operates in two separate phases: initial and follow-up inspections. In the initial inspection phase, the target structure is inspected for the first time using the proposed system; no records of previous inspections are found. The follow-up inspection

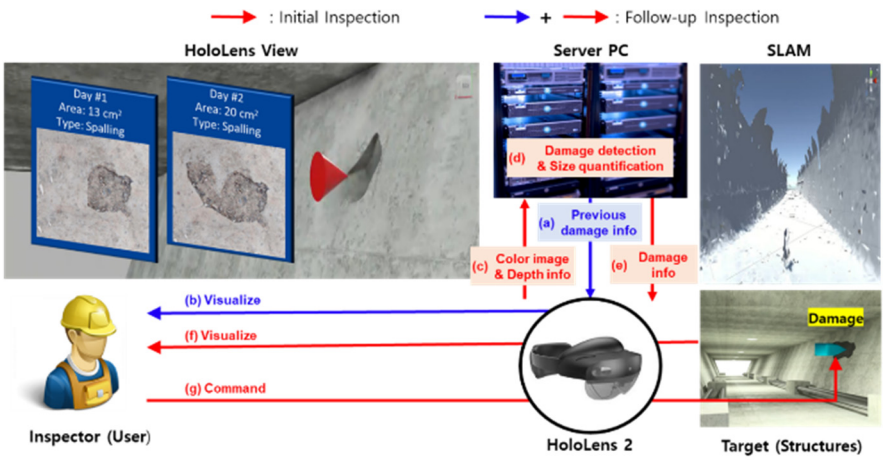


Fig. 1 Initial inspection & Follow-up inspection process

Fig. 2 Schematic of the proposed system

Table 1 Specifications of HoloLens 2

Camera resolution	Up to 2272 × 1278	Camera field of view	64.69° × 36.38°
Camera frame rate	Up to 30 fps	Memory	4 GB
Display resolution	1440 × 936	Display field of view	43° × 29°

phase is activated when the inspector has previously inspected the structure, and the damage inspection information generated during the previous inspection is saved in the proposed system. In Section 2, the details of the initial and follow-up inspections are explained, after a comprehensive description of the hardware of the proposed system.

2.1 System configuration

The proposed system consists of an AR headset and server PC, as shown in Fig. 2. Microsoft HoloLens 2, which is one of the most popular AR headsets, was used as the AR headset for the proposed system. HoloLens 2 is equipped with various sensors, such as an RGB camera, depth imager, and an inertial measurement unit (IMUs). The depth imager captures the three-dimensional distance between surroundings. The 3D distance measurements were combined with the acceleration and angular velocity from the IMU to continuously track the location of the headset, and a 3D map of its surroundings was generated using the HoloLens 2 simultaneous localization and mapping (SLAM) algorithm (Bahri *et al*. 2019). HoloLens 2 enables hands-free computing in onsite environments by supporting voice and gesture commands. The detailed specifications of HoloLens 2 are listed in Table 1.

The AR headset and server transmit data over a Wi-Fi network using TCP. As illustrated in Fig. 2, the AR headset transmits the captured color image and depth information to the server PC while computing its location using the SLAM algorithm. The server conducts damage detection and damage size quantification, and returns the results to the AR headset, including pixel-level segmented images and damage size. The proposed system saves and manages the following five information points for each detected damage:

- inspection date
- Azure spatial anchor ID
- damage type
- damage size
- damage image

The inspection date is a dd-mm-yyyy-formatted character string indicating the date on which the damage was detected. The Azure spatial anchor ID is the key to connecting to the Azure spatial anchor, which enables the permanent placement of a holographic marker at a precise damage location. The details of the azure spatial anchor are explained in Section 2.2.4. The damage type is one of the three most common damage types (i.e., spalling, exposed rebar, and efflorescence) and is the outcome of the deep-learning-based damage-detection algorithm in the proposed system. The damage size denotes the area of the detected damage and is measured in cm^2. The damage image is a captured color image of the detected damage, which is scaled and transformed as if taken from a specific distance in front.

2.2 Initial inspection

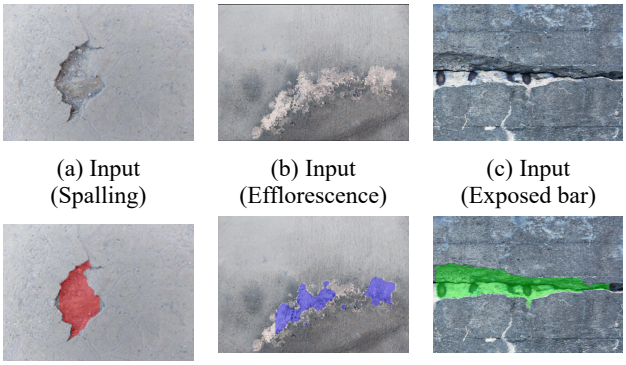
The procedure for the initial inspection phase is shown in Fig. 1. An inspector wearing the AR headset walks through the structure looking for damage in the same manner as in a conventional visual inspection. When the inspector identifies a potentially defective area and approaches the area closely (less than 1 m), the headset begins transmitting a captured color image of the area and the depth information of the surroundings to the server every second (Fig. 1(c)). The server then detects the damage at the pixel level using a deep learning algorithm and computes the area of the damage in metric units (Fig. 1(d)). After the computation, the server transmits the results to the headset (Fig. 1(e)), which displays the results to the inspector by superimposing the pixel-level segmented images on the actual physical damage (Fig. 1(f)). If the inspector determines that the visualized image is correct, a holographic marker indicating the damage location and a holographic panel, including the damage date, damage image, and damage size, are generated for the actual physical damage (Fig. 1(g)).

2.2.1 Damage detection scheme

In the proposed system, YOLACT is employed to detect structural damage because it is a one-stage pixel-level object-detection algorithm that allows real-time calculations with comparably high accuracy (Bolya *et al*. 2019). YOLACT operates in four steps. First, the feature maps of various sizes are generated using the ResNet-101 backbone network and a feature pyramid network. The feature maps are then transmitted to two parallel branches: the protonet and prediction head branch. In the Protonet branch, the largest feature map of the backbone network is used as the input to generate multiple prototype masks. The prediction head branch uses feature maps of all sizes generated by the backbone network and predicts the bounding boxes, labels, and mask coefficients; redundancy is removed through non-maximum suppression. The prototype masks and mask

Table 2 Dataset for YOLACT model

Damage type	Training & validation	Testing	Total
Spalling	369	52	421
Exposed bar	448	64	512
Efflorescence	333	47	380
Total	1,150	163	1,313



(a) Input (Spalling) (b) Input (Efflorescence) (c) Input (Exposed bar)

(d) Output (Spalling) (e) Output (Efflorescence) (f) Output (Exposed bar)

Fig. 3 Sample results of the YOLACT model.

coefficients are then linearly combined to produce the final mask for each target.

The dataset for the deep learning model comprised 1,313 damage images, as listed in Table 2. The images of damage were obtained from various sources, including industrial partners and other academic institutions (Zhang *et al*. 2021, 2020, Yang *et al*. 2018a, b, Li *et al*. 2019). To improve the prediction accuracy, data augmentation techniques such as random rotation, scaling, and Gaussian noise were applied to the training images. After training, the model exhibited a mean intersection over a union of 88.8% for the test dataset. Fig. 3 shows three examples of these results.

It is to be noted that the perfect detection of all damage to real structures using the trained YOLACT model is challenging because the number of training images in the dataset is insufficient to cover all construction environments. To overcome this limitation, an inspector can participate in the decision to use the YOLACT model in the proposed system. First, only images captured within 1 m of the target structure were transmitted to the server and input to the YOLACT model because most of the training images were captured from a close point. After the YOLACT model automatically detects the structural damage, the inspector makes a final judgment regarding the output. For example, when a model produces incorrect output, an inspector can ignore the output to prevent false alarms.

2.2.2 Damage size quantification

In the proposed system, the YOLACT model is used to acquire a pixel-level segmented result, and the area of detected damage is computed in pixel units. However, a damaged area in metric units is required to assess the current conditions and history of damage. The proposed

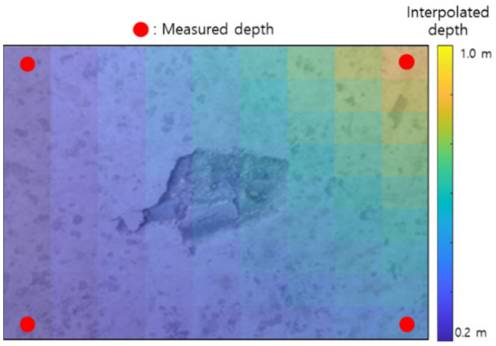


Fig. 4 Interpolation of depth measurements

system utilizes depth information to convert damaged areas into metric pixel units. Let $scale_{hor}$ and $scale_{ver}$ be the pixel width and height per unit depth, respectively. Subsequently, $scale_{hor}$ and $scale_{ver}$ can be computed using the following equations.

$$scale_{hor} = \frac{2\tan(0.5FoV_{hor})}{w_{frame}}, \quad (1)$$

$$scale_{ver} = \frac{2\tan(0.5FoV_{ver})}{h_{frame}}, \quad (2)$$

where FoV_{ver} and FoV_{hor} are the vertical and horizontal fields of view and w_{frame} and h_{frame} are the horizontal and vertical resolutions of the HoloLens camera, respectively. Next, the depth information of the four corner pixels of the image was acquired using a depth imager equipped in the HoloLens. The depth information of the other pixels was computed using bilinear interpolation, as shown in Fig. 4, under the assumption that the damage was located on a planar surface. After interpolation, i^{th} pixel width and height in the metric unit, denoted as w^i_{pixel} and h^i_{pixel}, can be computed by multiplying $scale_{hor}$ and $scale_{ver}$ by the depth of the i^{th}pixel, as follows

$$w^i_{pixel} = scale_{hor} \times depth^i_{pixel} \quad (3)$$

$$h^i_{pixel} = scale_{hor} \times depth^i_{pixel} \quad (4)$$

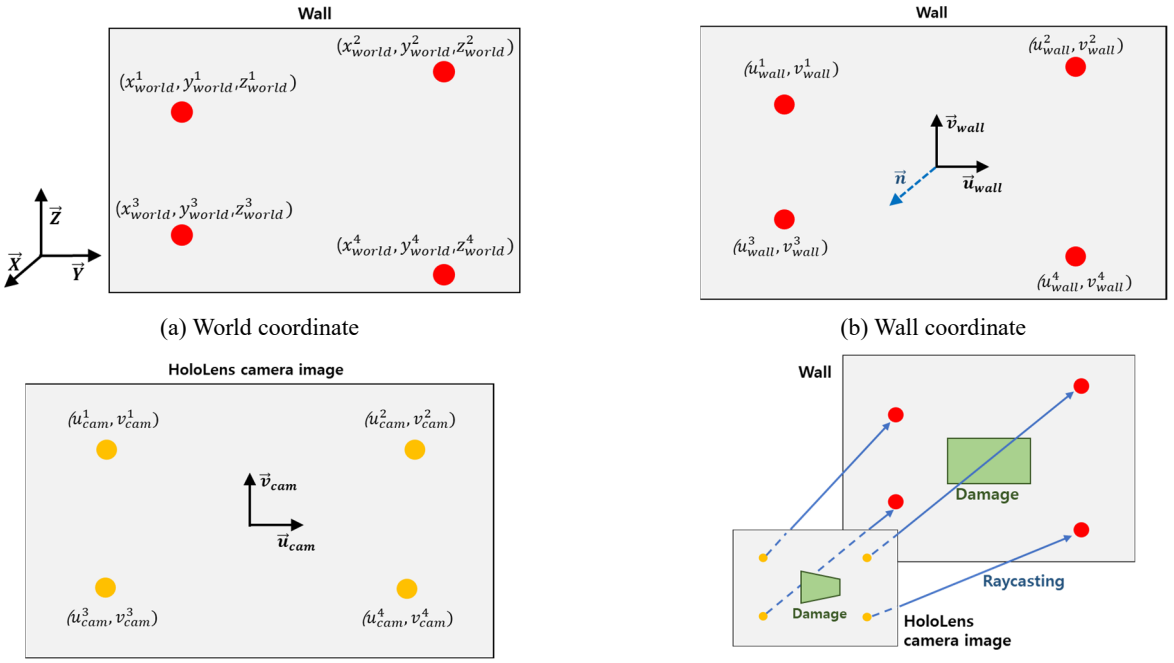
where $depth^i_{pixel}$ denotes the depth of the i^{th} pixel. The area of damage was calculated by summing all areas of the pixels that were classified as damaged as follows

$$\text{Damage area} = \sum_{i=1}^{n} (w^i_{pixel} \times h^i_{pixel}) \quad (5)$$

where n is the number of pixels classified as damage.

2.2.3 View normalization

In a conventional visual inspection and management system, the detected damage is captured as image files and managed by the system. However, the images are captured from various viewpoints and distances, which can lead to confusion because the damage captured from a close



(a) World coordinate

(b) Wall coordinate

(c) HoloLens camera coordinate

(d) Raycasting pixels of HoloLens camera to wall

Fig. 5 Description of view normalization

viewpoint appears to be larger than that captured from a far viewpoint. Therefore, it is convenient to convert all the images into normalized images. These normalized images are scaled and transformed as if they are taken from a specific distance in the front.

In the view normalization process, three coordinate systems are defined: world, wall, and HoloLens camera coordinates. The world coordinate system depicted in Fig. 5(a) is a 3D coordinate system that shares its origin with the HoloLens mapping coordinates. The point coordinates can be expressed as $(x^i_{world}, y^i_{world}, z^i_{world})$, where i is the point index. The wall frame coordinates (u^i_{wall}, v^i_{wall}), as shown in Fig. 5(b), are 2D coordinates defined with two base vectors $\vec{v}_{wall}$ and $\vec{u}_{wall}$ as follows

$$\vec{v}_{wall} = \vec{z}_{world} - \frac{\vec{z}_{world} \cdot \vec{n}}{\|\vec{n}\|^2}\vec{n} \tag{6}$$

$$\vec{u}_{wall} = \vec{v}_{wall} \times \vec{n} \tag{7}$$

where $\vec{n}$ is the normal vector of the wall plane. Subsequently, $\vec{v}_{wall}$, as expressed in Eq. (6), is a vector projected onto $\vec{z}_{world}$ of the world coordinates, and $\vec{u}_{wall}$ is expressed as a cross product of $\vec{v}_{wall}$ and $\vec{n}$. The HoloLens camera coordinates (u^i_{cam}, v^i_{cam}) are 2D, the origin of which is the principal point of the HoloLens color camera, as illustrated in Fig. 5(c).

The basic principle of the view normalization process is illustrated in Fig. 5(d). First, the 3D points $(x^i_{world}, y^i_{world}, z^i_{world})$ are acquired with respect to the world coordinates by ray-casting the four predefined corner pixels (u^i_{cam}, v^i_{cam}) of the HoloLens camera image to the wall. Here, ray casting refers to finding a point where a virtual light ray is cast on its path from the focal point of a camera through each pixel in the camera sensor to determine what is visible along the ray in a 3D scene. Next, the 3D points $(x^i_{world}, y^i_{world}, z^i_{world})$ are transformed into wall coordinates (u^i_{wall}, v^i_{wall}). Subsequently, a 3 × 3 homography matrix **H** between the four pixels (u^i_{cam}, v^i_{cam}) of the HoloLens camera frame and the four points (u^i_{wall}, v^i_{wall}) of the wall coordinates is computed using the following expression

$$\begin{bmatrix} u^k_{cam} \\ v^k_{cam} \\ 1 \end{bmatrix} = \mathbf{H} \begin{bmatrix} u^k_{wall} \\ v^k_{wall} \\ 1 \end{bmatrix} \tag{8}$$

The captured image was then converted to be viewed from the front at a consistent distance by multiplying the computed homography matrix with the captured image and saving it as a damaged image.

2.2.4 Holographic marker localization

A holographic marker is an indicator in the form of a hologram that indicates the location of the detected damage. Accurate holographic marker localization enables the position of the marker to remain unchanged while the inspector is moving and when the inspector returns to the same area. This localization was accomplished using SLAM techniques and the Azure spatial anchor system (Ong 2021).

When a holographic marker was created, its 3D coordinates were computed using a custom SLAM algorithm developed by Microsoft (Izadi *et al.* 2011, Nießner *et al.* 2013, Dai *et al.* 2017). The 3D coordinates of the marker based on the SLAM algorithm represent the location of the detected damage; however, maintaining the position of the holographic marker over time using only the SLAM algorithm is challenging because of the accumulated

drift error in the SLAM algorithm and the change in the coordinate system generated when the system restarts.

To address these issues, the proposed system utilizes an azure spatial anchor system. Conventionally, spatial anchors are used in AR applications to stabilize or maintain holograms in physical spaces or surfaces. When the holographic marker was created using the spatial anchor system, the AR headset collected and stored the visual information surrounding the marker. When revisiting the same location, the AR headset collects the surrounding visual information and compares it with the previously stored data. If the information matches, the system determines that the inspector comes to the same place, and the marker is automatically recreated exactly where it was previously created and maintains its position accurately. Because the system depends on surrounding visual information, the holographic marker can maintain its position even though the coordinate system changes when the system restarts.

2.3 Follow-up inspection

A follow-up inspection was executed when the inspector revisited the structure for a subsequent examination after the initial inspection. The overall scheme of the follow-up inspection is shown in Fig. 1. When the inspector revisits the structure after the initial inspection and all the initial damage information is loaded, a holographic marker and panel are created at the previous damage location (Figs. 6(a) and (b)). The inspector can examine the damage in the same manner as the initial inspection to evaluate the current state of the previously detected damage or identify newly generated damage (Figs. 6(c)-(g)).

In this process, the proposed system automatically determines whether the detected damage is new or has been detected previously. When damage is newly detected, a holographic marker with damage information is created in the same manner as in the initial inspection. However, when damage has been previously detected, the proposed system updates its inspection date, damage size, and damage image. In addition, the captured image of the damage is processed using a novel view alignment method, which is discussed in Section 2.3.3 in detail, for perfect alignment with the damaged image at the initial inspection.

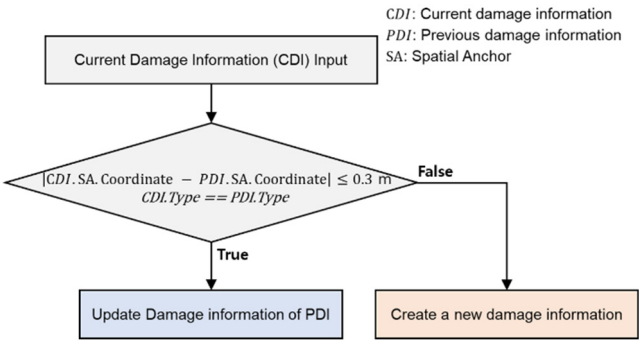


Fig. 6 Damage information update scheme

2.3.1 Loading previous damage information

Damage information, including the inspection date, damage images, types, sizes, and azimuth spatial anchor ID, is saved during the initial inspection. When the inspector loads the previous damage information, a holographic marker and panel displaying the damage information are created and placed at the previously detected damage location based on the Azure spatial anchor system. Although the coordinate system at the follow-up inspection differed from that at the initial inspection, the holographic marker is located at the previously detected damage location because the Azure spatial anchor system relies on visual surrounding information rather than coordinates. The inspector can then view the holographic marker and damage information through the AR headset, which enables him to promptly locate the previous damage and check and track the damage information onsite.

2.3.2 Damage information update

Damage location and type are crucial factors in determining whether damage has been previously detected. The damage location is represented by the coordinates of the azimuth spatial anchor. When the proposed system detects damage, the location and type of damage are compared with previous damage information. Because the mapping accuracy of HoloLens 2 is limited to approximately 0.3 m, the proposed system determines that if the distance between the damages is 0.3 m or less, they are in the same location. Consequently, as illustrated in Fig. 6, if the damage locations and types are identical, the damage is considered identical or an expansion of the previous damage. In this case, only the inspection date, damage size, and damage image were saved, in addition to the previously detected damage information. In contrast, if the damage locations or damage types differ, the damage is considered a new damage, and all the detected damage information is saved.

2.3.3 View alignment

Regardless of whether the detected damage is new or has been previously detected, all pictures of the damage are processed with view normalization to manage them with a specific angle and distance and saved as damage images.

When the previous and current damages are captured in different views, the damage images are matched through view normalization. However, there are a few discrepancies owing to errors in the depth measurement. Therefore, an additional process, called view alignment, is executed to perfectly match the current damaged image with the initially detected damaged image.

For view alignment, the damaged image from the initial inspection is used as a reference. With SURF (speeded-up robust features), the features are extracted from both the reference and current images, and the transformation matrix between the two images is computed [22]. The current picture is then converted by multiplying with the computed transformation matrix. Then, the two images perfectly overlap without discrepancies, and the inspector can easily and intuitively compare the damage status and track the damage growth.

Fig. 7 Inspection with HoloLens 2

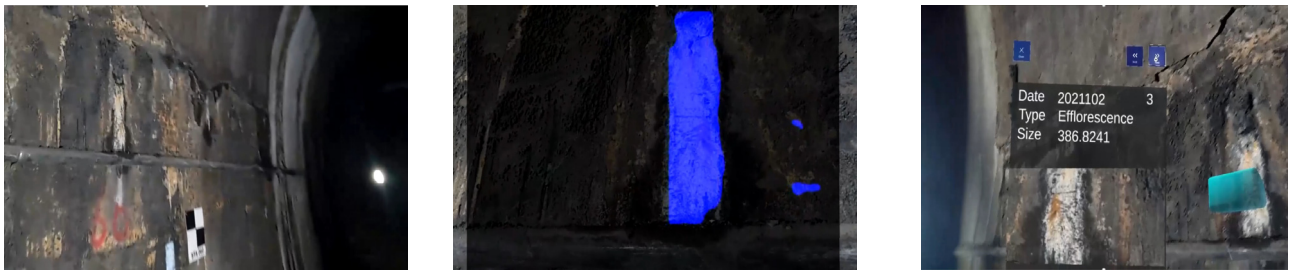
3. Experiments

The performance of the proposed system is verified through field tests in a closed tunnel in Daejeon, South Korea. The length of the tunnel is 440 m, and it has an efflorescence as large as 400 cm^2 in the middle. This verification aimed to evaluate whether the proposed system operates as intended in terms of damage detection, damage quantification, and damage localization. In addition, the efficiency of the proposed system was evaluated by analyzing its ability to reduce the inspection time relative to conventional human-oriented visual inspection. The proposed system was tested under two scenarios: initial and follow-up inspections. In the initial inspection scenario, inspectors first entered the tunnel for inspection using the proposed system. During the follow-up inspection, the inspectors revisited the tunnel for a second examination. Six participants aged 28–47 years participated in the user study to compare the inspection time of the proposed system with the conventional human-oriented visual inspection time. Half of the participants were requested to attempt the initial and follow-up inspections using the proposed system, as shown in Fig. 7, and the inspection time was measured. On the other hand, the others were requested to try the both initial and follow-up inspection with the conventional inspection and the inspection time was measured.

3.1 Field test: Initial inspection

An inspector wearing HoloLens 2 walked through the tunnel to look for damage. In this instance, the inspector cannot see any visual information through HoloLens 2, as shown in Fig. 8(a). When the inspector moved closer to the area where the damage appeared, the headset transmitted the color images and depth information to the server. The server then performed damage detection using the YOLACT model, which generated detection results in the form of a pixel-level segmented image and damage area in metric units and returned the damage detection results to the headset. After receiving the damage-detection results, HoloLens 2 displays the pixel-level segmented image to the inspector, as shown in Fig. 8(b). Here, efflorescence is visualized in blue, and the segmented image overlaps with the real physical damage. The inspector air-tapped to create a holographic marker after determining that the visualized result was correct. A holographic marker and panel were then created, and the damage information was saved and visualized through its display, as shown in Fig. 8(c).

In this test, the damage type was correctly identified as efflorescence, and the computed damage size was 386 cm^2,



(a) Walk toward damage (b) Inspect the damage (c) Visualize damage information

Fig. 8 Initial inspection results

(a) Original captured image (b) View normalization

Fig. 9 View normalization result

Table 3 Comparison of inspection average time at the initial inspection

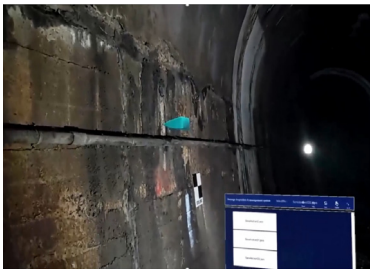
Tasks	Proposed system (seconds)	Conventional human-oriented inspection (seconds)
Finding damage with approach	296	302
Size measurement	1	31
Photographing a damage image	1	9
Total	298	342

which represented a 3.5% error compared to the ground truth. In addition, the damaged image was processed with view normalization to be viewed from the front at a fixed distance, as shown in Fig. 9. Furthermore, the inspection time was measured by subdividing the entire inspection into finding damage with approach, damage size measurement, and taking a damage image. The results were compared to the conventional human-oriented inspection, as detailed in Table 3. During the initial inspection, both methods required approximately 300 s to find and approach the damage; however, the proposed system reduced the time taken to measure the damage size and photograph the damage from 31 to 1 s and 9 to 1 s, respectively.

3.2 Field test: Follow-up inspection

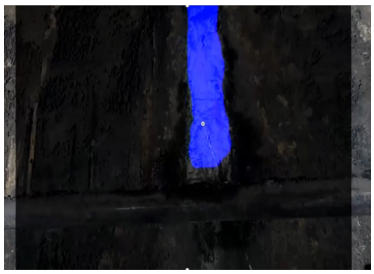
In this scenario, the inspector revisited the tunnel for a second inspection. Note that the inspector could not observe anything at this instance and was unaware of the location of the previously detected damage, as shown in Fig. 10(a). When the inspector loads the previous inspection information, a holographic marker is created at the exact location of the previously detected damage, allowing the inspector to easily locate the previous damage, as shown in Fig. 10(b). In addition, the inspector can check the previous damage information onsite, as shown in Fig. 10(c). When the inspector moved closer to inspect the damage again and examined whether it had worsened, the headset visualized its current detection result based on the YOLACT model, as shown in Fig. 10(d). Because the result was correct, the inspector was air-tapped to save the current damage information. The inspector could then check the current damage information on the display, as shown in Fig. 10(e).

(a) Re-visit the identical place



(b) Load the previous damage information

(c) Visualize previous damage information



(d) Re-inspect the damage

(e) Visualize damage information

(f) Holographic marker from far distance

Fig. 10 Follow-up inspection results

(a) Original captured image

(b) View normalization

Fig. 11 View alignment result

(a) At distance of 1 m (b) At distance of 5 m (c) At distance of 10 m

(d) At distance of 20 m (e) At distance of 40 m (f) At distance of 50 m

Fig. 12 Holographic marker localization according to the distance

Table 4 Comparison of inspection average time at the follow-up inspection

Tasks	Proposed system (seconds)	Conventional human-oriented inspection (seconds)
Finding damage with approach	174	273
Size measurement	1	25
Photographing a damage image	1	11
Total	176	309

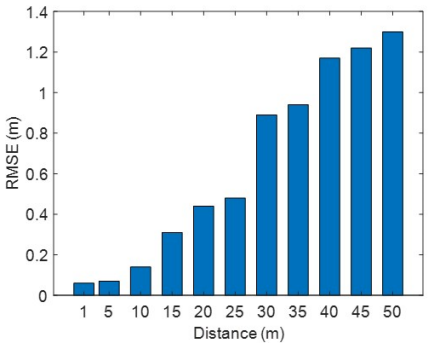


Fig. 13 Holographic marker localization error

In addition, the inspector can visually identify the holographic marker at a distance of up to 50 m, as shown in Fig. 10(f).

In the follow-up inspection, the detected damage was determined to be identical to the previously detected damage based on the damage information update algorithm because the damage type was the same as the previous damage and the distance between them was less than 0.3 m. The damage size was computed to be 430 cm^2 with a 7.5% error compared with the ground truth. In addition, the images were processed using view alignment to match the previously damaged image. Although Figs. 9(a) and 11(a) are captured from various viewpoints, Figs. 9(b) and 11(b) match well after view alignment. This result facilitates the inspector's understanding of the extent to which the damage has grown over time. With respect to inspection time, the proposed method showed improvements in all parts compared with conventional human-oriented inspection, as detailed in Table 4. In finding previous damages with approach, the proposed system required 174 s, demonstrating 36% improvement. Most of the inspection time was spent walking to the damage location, not finding the damage, because the system visualized where the damage was in the form of a holographic marker. However, conventional inspection took 273 s to find the location of the previously detected damage and approach it because the inspectors had difficulty in finding the damage location only with 2D drawings indicating the same. In terms of size measurement and photographing the damaged image, it showed a similar performance as the initial inspection. In addition, the location accuracy of the holographic marker was analyzed based on the distance from the inspector to the marker, as shown in Figs. 12 and 13. When the distance is 1 m, the holographic marker was 0.06 m away from the actual damage location. As the distance increased to 50 m, the error also increased to 1.3 m due to drift error of SLAM algorithm. Even though it appears to have a large error at a far distance, the accuracy was sufficient for navigating to the location of the previous damage because the marker moved to a more accurate location as the inspector approached it.

4. Conclusions

In this study, a structural inspection system for marking damage information onsite based on augmented reality technology was proposed to improve the convenience of onsite inspectors during consecutive inspections in the field, which has seldom been studied in previous research. The performance of the proposed system is validated using field tests. The improvement in inspection time and the possibility of damage history tracking demonstrate that the proposed system can effectively assist inspectors.

Currently, the proposed system assumes that a gap of more than 0.3 m exists between damages. To enhance the spatial resolution of the system, we plan to improve the mapping accuracy of the AR headset. In addition, the system needs improvement in damage detection in order to apply the system to real construction site. Firstly, the system focuses on detecting spalling, exposed rebar, and efflorescence currently. In order to accommodate more cases of structural damage, we plan to expand the categories to include more types of damage. Secondly, the inspector cannot correct the result in the case of a false negative indication of damage, though the inspector simply can ignore the outcome by not generating the holographic marker in the case of a false positive. This problem can be solved by introducing a click-based interactive segmentation model instead of the YOLACT model. The click-based interactive segmentation enables pixel-level segmentation and image editing with simple user clicks (Sofiiuk *et al*. 2020). By adopting the click-based interactive segmentation into the proposed system, the system can more accurately identify the damage by interacting with the inspector.

Declaration of competing interest

The authors declare that they have no competing financial interests or personal relationships that may have influenced the work reported in this study.

Acknowledgments

This work was supported by National Research Foundation of Korea (NRF) grants funded by the Korean government (MSIT) (grant numbers 2022R1C1C2008186). This study was supported by a grant (2021-MOIS32-041) from the Ministry of Interior and Safety (MOIS, Korea).

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2019-0-00075, Artificial Intelligence Graduate School Program (KAIST)).

References

Ali, R., Kang, D., Suh, G. and Cha, Y.J. (2021), "Real-time multiple damage mapping using autonomous UAV and deep faster region-based neural networks for GPS-denied structures", *Autom. Constr.*, **130**(103831). https://doi.org/10.1016/j.autcon.2021.103831

Bahri, H., Krcmarik, D., Moezzi, R. and Kočí, J. (2019), "Efficient use of mixed reality for BIM system using Microsoft HoloLens", *IFAC-Papers Online*, **52**(27) 235-239. https://doi.org/10.1016/j.ifacol.2019.12.762

Bolya, D., Zhou, C., Xiao, F. and Lee, Y.J. (2019), "Yolact: Real-time instance segmentation", *Proceedings of the IEEE/CVF International Conference on Computer Vision* (*ICCV*), pp. 9157-9166.

Dai, A., Nießner, M., Zollhöfer, M., Izadi, S. and Theobalt, C. (2017), "BundleFusion: real-time globally consistent 3D reconstruction using on-the-fly surface reintegration", *ACM Trans. Graph.*, **36**(3), 1-18. https://doi.org/10.1145/3054739

Falorca, J.F., Miraldes, J.P. and Lanzinha, J.C.G. (2021), "New trends in visual inspection of buildings and structures: study for the use of drones", *Open Eng.*, **11**(1), 734-743. https://doi.org/10.1515/eng-2021-0071

Izadi, S., Kim, D., Hilliges, O., Molyneaux, D., Newcombe, R., Kohli, P., Shotton, J., Hodges, S., Freeman, D., Davison, A. and Fitzgibbon, A. (2011), "KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera", *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, Santa Barbara, CA, USA.

Karaaslan, E., Bagci, U. and Catbas, F.N. (2019), "Artificial intelligence assisted infrastructure assessment using mixed reality systems", *Transp. Res. Rec.*, **2673**(12). https://doi.org/10.1177/0361198119839988

Kim, B. and Cho, S. (2020), "Automated multiple concrete damage detection using instance segmentation deep learning model", *J. Appl. Sci.* **10**(22), 1-17. https://doi.org/10.3390/app10228008

Kim, I.H., Jeon, H., Baek, S.C., Hong, W.H. and Jung, H.J. (2018), "Application of crack identification techniques for an aging concrete bridge inspection using an un- manned aerial vehicle", *Sensors*, **18**(6), 1-14. https://doi.org/10.3390/s18061881

Li, S., Zhao, X. and Zhou, G. (2019), "Automatic pixel-level multiple damage detection of concrete structure using fully convolutional network", *Comput.Aided Civ. Infrastruct. Eng.*, **34**(7), 616-634. https://doi.org/10.1111/mice.12433

Maharjan, D., Agüero, M., Mascarenas, D., Fierro, R. and Moreu, F. (2020), "Enabling human–infrastructure interfaces for inspection using augmented reality", *Struct. Health Monitor.*, **20**(4), 147592172097701. https://doi.org/10.1177/1475921720977017

Mascarenas, D.D., Ballor, J.P., McClain, O.L., Mellor, M.A., Shen, C.Y., Bleck, B., Morales, J., Yeong, L.M.R., Narushof, B., Shelton, P. and Martinez, E. (2020), "Augmented reality for next generation infrastructure inspections", *Struct. Health Monitor.*, **20**(4), 147592172095384. https://doi.org/10.1177/1475921720953846

Nguyen, D.C., Nguyen, T.Q., Jin, R., Jeon, C.H. and Shim, C.S. (2021), "BIM-based mixed-reality application for bridge inspection and maintenance", *Constr. Innov.*, **22**(3), 487-503. https://doi.org/10.1108/ci-04-2021-0069

Nießner, M., Zollhöfer, M., Izadi, S. and Stamminger, M. (2013), "Real-time 3D reconstruction at scale using voxel hashing", *ACM Trans. Graph*, **32**(6), 1-11. https://doi.org/10.1145/2508363.2508374

Ong, S. (2021), *Beginning Windows Mixed Reality Programming*, (1st ed.), Apress, Berkeley, CA, USA. https://doi.org/10.1007/978-1-4842-2769-5

Puž, G., Radić, J. and Tenžera, D. (2012), "Visual inspection in evaluation of bridge condition", *Gradevinar*, **64**(9), 717-726. https://doi.org/10.14256/jce.718.2012

Rublee, E., Rabaud, V., Konolige, K. and Bradski, G. (2011), "ORB: an efficient alternative to SIFT or SURF", *Proceedings of 2011 International Conference on Computer Vision*,

Barcelona, Spain, pp. 2564-2571. https://doi.org/10.1109/ICCV.2011.6126544

Sofiiuk, K., Petrov, I., Barinova, O. and Konushin, A. (2020), "f-brs: Rethinking backpropagating refinement for interactive segmentation", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), pp. 8623-8632.

Wang, S., Zargar, S.A. and Yuan, F.G. (2021), "Augmented reality for enhanced visual inspection through knowledge-based deep learning", *Struct*. *Health Monit*., **20**(1), 426-442. https://doi.org/10.1177/1475921720976986

Yang, X., Li, H., Yu, Y., Luo, X., Huang, T. and Yang, X. (2018a), "Automatic pixel-level crack detection and measurement using fully convolutional network", *Comput*. *Aided Civ*. *Infrastruct*. *Eng*., **33**(12), 1090-1109. https://doi.org/10.1111/mice.12412

Yang, L., Li, B., Li, W., Jiang, B. and Xiao, J. (2018b), "Semantic metric 3D reconstruction for concrete inspection", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, Salt-Lake City, UT, USA, pp. 1624-1632. https://doi.org/10.1109/CVPRW.2018.00204

Zhang, C., Chang, C.C. and Jamshidi, M. (2020), "Concrete bridge surface damage detection using a single-stage detector", *Comput*.*-Aided Civ*. *Infrastruct*. *Eng*., **35**(4), 389-409. https://doi.org/10.1111/mice.12500

Zhang, C., Chang, C.C. and Jamshidi, M. (2021), "Simultaneous pixel-level concrete defect detection and grouping using a fully convolutional model", *Struct*. *Health Monitor*., **20**(4), 2199-2215. https://doi.org/10.1177/1475921720985437

HJ