Diminished reality using semantic segmentation and generative adversarial network for landscape assessment: Evaluation of image inpainting according to colour vision

Takuya Kikuchi, Tomohiro Fukuda*, and Nobuyoshi Yabuki
Division of Sustainable Energy and Environmental Engineering, Graduate School of Engineering, Osaka University 2-1, Yamadaoka, Suita, Osaka 565-0871, Japan.
E-mail: fukuda.tomohiro.see.eng@osaka-u.ac.jp (T.F.), takuya.kikuchi@it.see.eng.osaka-u.ac.jp (T.K.), yabuki@see.eng.osaka-u.ac.jp (N.Y.)
*Corresponding author. Tel: +81-6-6879-7661; Fax: +81-6-6879-7661

Abstract
The objective of this research is to develop a method to detect and virtually remove representations of existing buildings from a video stream in real-time for the purpose of visualising a future scenario without these buildings. This is done by using semantic segmentation, which eliminates the need to create 3D models of the buildings and the surrounding scenery, and a generative adversarial network (GAN), a deep learning method for generating images. Real-time communication between devices enables users to utilize only portable devices equipped with a camera to visualise the future landscape onsite. As verification of the proposed method's usefulness, we evaluated the complementation accuracy of the GAN and real-time performance of the entire method. The results indicated that the process is completed accurately when the area to be complemented is less than 15% of the view and that the process runs at 5.71 fps. The proposed method enables users to understand intuitively the future landscape and contributes to reducing the time and cost for building consensus.

Graphical Abstract

Keywords: Landscape assessment, diminished reality (DR), generative adversarial network (GAN), semantic segmentation, web real-time communication (WebRTC)
Abbreviations: AR, augmented reality; CNN, convolutional neural network; DR, diminished reality; GAN, generative adversarial network; GPU, graphics processing unit; GSV, Google Street View; IB-DR, inpainting-based diminished reality; IoU, intersection over union; JIS, Japan Industrial Standard; OB-DR, observation-based diminished reality; PC, personal computer; SfM, structure from motion; VR, virtual reality; WebRTC, web real-time communication

Highlights

・Developed DR system using semantic segmentation, a GAN, and real-time communication.

・Automatically visualises post-demolition landscape without prior preparation.

・High-accuracy virtual removal when the object occupies less than 15% of the screen.

・The landscape after virtual demolition can be visualised at 5.71 fps on average.

1. Introduction

In many cities today, the number of old buildings—the building stock—is increasing. To make way for redevelopment projects in existing urban areas, existing structures (hereinafter referred to as “objects to be removed”) are demolished and new structures are built or the land is reused as public spaces, such as sidewalks and plazas. The number of redevelopment projects is expected to increase in the future as the number of old buildings grows. In the redevelopment process, it is necessary to discuss and build consensus on the redevelopment plan among the stakeholders. In addition to clients and designers with expertise, neighbours and citizens who do not have expertise will participate. Differences in the understanding of the plan among stakeholders are expected to occur. Thus, visualisation methods that help foster an intuitive understanding of the planned future landscape are needed in order to facilitate smooth consensus-building (Bishop & Lange, 2005; Sheppard, 1989). Therefore, in the planning stages of redevelopment projects, methods are needed to virtually visualise future landscapes before structures are demolished.

Visualisation methods of the future landscape include the photomontage method (Rød & van der Meer, 2009), in which the future landscape is synthesized into a photograph of the current landscape, and augmented reality (AR) (Caudell & Mizell, 1992), which superimposes a 3D model created in a 3D virtual space onto the real space (Kido et al., 2021). However, the photomontage and miniature modelling methods have two problems: they cannot be applied immediately to accommodate opinions during a discussion, and they lack a sense of reality because the participants cannot examine the plan from the perspective of the actual space. AR-based methods superimpose a virtual model on the real space, so fewer 3D models need to be created in advance compared with virtual reality (VR) (Zeltzer, 1992). However, AR cannot virtually remove objects that exist in reality, so it is not possible to represent the landscape after the removal of buildings.

To solve these problems, the diminished reality (DR) (Mann, 1994) method was proposed to represent the modified landscape. DR is a technology for making existing objects virtually invisible by superimposing a landscape in the background. In the DR-based landscape visualisation method, the future landscape is displayed
by virtual demolition of the target structures. Inoue et al. (2018) proposed a DR method using structure-from-motion (SfM) (Ullman, 1979) technology to represent the future landscape, and Kido et al. (2020) proposed a DR method using SfM technology and object detection. These methods enabled the virtual removal of objects slated for demolition and removal and the representation of future landscapes. However, these methods have the problem that 3D models of the object to be removed and the surrounding landscape need to be created in advance. In addition, Zhang et al. (2021) proposed a method combining semantic segmentation (Long et al., 2015) and a generative adversarial network (GAN) (Goodfellow et al., 2014) to represent the landscape after removing the object to be removed without any prior preparation. However, there are no reports of this method being applied to buildings, although it has been used for pedestrians, bicycles, and vegetation.

The objective of this research is to develop a method for detecting and virtually removing representations of existing buildings from a video stream in real-time in order to enable visualisation of a future scenario without these buildings. This is done by using semantic segmentation, which eliminates the need to create 3D models of the buildings and the surrounding scenery, and a GAN. Real-time communication between devices enables users to utilize only portable devices equipped with a camera to visualise the future landscape onsite. This method makes it possible to visualise the future landscape (after the demolition and removal of buildings) with only mobile devices taken to the viewpoint location, without requiring time for advance preparation. Therefore, compared with previous methods, it is easier to visualise the future landscape after the demolition and removal of structures. As a result, in redevelopment including the demolition of buildings, the proposed method is expected to contribute to reducing the cost of consensus-building by making it possible to share the future vision intuitively. In addition, the proposed method is expected to address previously unmet needs in the engineering field (Hartmann & Trappey, 2020).

The previous version of this report published in eCAADe2021 (Kikuchi et al., 2021) showed that it is possible to remove buildings by combining semantic segmentation and GAN. However, field verification at multiple locations and evaluation of completion accuracy according to human senses were left as future tasks. Therefore, in this paper, another verification target was added and the accuracy was verified again, and the completion accuracy was evaluated according to human senses. For the latter purpose, \( \Delta E^* \) (Backhaus et al., 1998), which evaluates colour differences in the CIELAB colour space (International Organization for Standardization, 2008) and is considered to approximate human vision, was used as an evaluation index. This makes it possible to clarify the conditions under which the proposed method can be applied and to evaluate the degree to which completion accuracy varies depending on the conditions, according to human senses.

This paper consists of six sections. Following this section presenting the background and objectives of the study, Section 2 summarizes previous studies related to this research and clarifies the contribution of this research. The proposed method is explained in Section 3, and verification methods and results are described in Section 4 to demonstrate the usefulness of the proposed method. The results are discussed in Section 5, and Section 6 provides conclusions.

2. Literature Review

In this section, we summarize the relevant previous research and conclude with the contribution of this research.
2.1. DR for landscape study

Many AR/VR studies have been conducted to enable various stakeholders to understand redevelopment plans. Lin et al. (2018) proposed a VR system to discuss designs with non-experts. This system is a semi-immersive VR system using building information modelling. Although a large screen is required to project the VR, it can be used to help hospital staff fully understand and examine both the external and internal design of new hospital buildings. Abualdenien and Borrmann (2020) proposed a method that used VR to represent the design, including ambiguous parts that have not yet been finalized. Using this method, they reduced the time needed to understand the design. Han et al. (2021) proposed an occlusion detection method to visualise hidden objects in VR. By comparing the point cloud obtained by virtual scanning of an object with the cloud obtained by transforming the object, the front-back relationship between parts can be determined semi-automatically, although the accuracy of the determination depends on the position of the virtual scan. Mutis and Ambekar (2020) also proposed a method to improve the immersion of walkthrough AR. Although the accuracy decreases as the distance from the point where the simulation is started increases, this method can be used to manage the progress of construction by comparing the current state with the superimposed completed state. In addition, Pejic et al. (2017) developed a mobile application that uses VR and AR to represent the future landscape of a residential area. This enables each user to view free from the position he or she wants to see, making it easier to understand the structure and other aspects. A project was also conducted to develop systems for human-computer interaction, visualisation and intelligent modelling related to the architectural field using a common VR platform (Schnabel et al., 2021).

As shown above, AR/VR is a useful technology for building consensus among stakeholders because it can represent the future and help them understand redevelopment plans. However, these AR methods cannot be applied to the visualisation of the future landscape after the removal of buildings, which is the target of our research, because AR cannot virtually remove objects in the real space. In addition, when applying the VR method, it is necessary to create the surrounding landscape, so it takes a lot of effort and money to visualise the future landscape after the buildings are demolished.

Therefore, DR, which is capable of removing objects that exist in real space, is considered a necessary research field in architectural planning and related fields (Davila Delgado et al., 2020). DR is capable of diminish, see-through, replace, and inpaint functions (Mori et al., 2017). Also, DR can be classified into inpainting-based diminished reality (IB-DR) and observation-based diminished reality (OB-DR), depending on the method used to obtain information about the background behind the removed object (Eskandari & Motamedi, 2021). DR has been used in a variety of fields, including the medical field and the industrial field (Navab et al., 1999; Zokai et al., 2003). DR has also been used to remove people from images provided by services such as Google Street View (GSV) (Google, n.d.) for privacy reasons (Flores & Belongie, 2010).

DR is being researched for application in the field of landscape studies as a method to represent future landscapes (Inoue et al., 2018; Kido et al., 2020; Zhang et al., 2021). Inoue et al. (2018) proposed a DR-based method for visualising future landscapes that virtually demolishes the target building by superimposing its background over it. The background was obtained from a 3D model that includes the target building and its surrounding landscape and is created using SfM technology from a large number of photographs in advance. Kido et al. (2020) also incorporated object detection using machine learning into the method of Inoue et al. (2018). In this method, moving objects were detected using machine learning and automatic virtual removal by
superimposing their background over the objects. As a result, it is possible to visualise the future landscape correctly after the removal of target structures, such as elevated roads and steel bridges, even when objects are moving over them.

These OB-DR methods were able to produce compelling results. However, a common issue with these methods is that they require creating 3D models of the object to be removed, such as a building and its background landscape, in advance. Creating 3D models using SfM technology requires considerable time to take pictures of the target objects and their surroundings and to generate the 3D models. Therefore, users must make time to do these tasks in advance. Also, 3D models using 3D modelling software can be created from, for example, drawings without using SfM, but this also takes time and requires 3D information (drawings) to create 3D models. Thus, when 3D models cannot be created, these methods cannot be applied.

Zhang et al. (2021) proposed an IB-DR method that combines semantic segmentation and a GAN. This method can automatically and virtually remove pedestrians, bicycles, plants and cars in front of a building’s facade from photos. This method made it possible to automatically detect and virtually remove objects without prior preparation. However, the method has not yet been applied to the representation of future landscapes that include the demolition of buildings.

2.2. Detecting objects
Kido et al. (2020) detected moving objects with MobileNetSSD, a network for object detection (Krizhevsky et al., 2012) based on the Single Shot MultiBox Detector (Liu et al., 2016). Object detection was performed by a model that detects an object in an image; encloses the detected object in a bounding box; and outputs the detected position, size, and category as the result. However, because the object detection used a bounding box to output the object to be detected, pixels other than the object to be detected were also included. In contrast, semantic segmentation (Long et al., 2015) is a deep learning model that can classify and label photos pixel by pixel and colourize each label with a pre-specified colour. Many algorithms have been proposed to speed up the process and improve detection accuracy (Zhao et al., 2018; Wang et al., 2019). This method has been applied to the analysis of medical images and automatic driving of cars (Müller & Kramer, 2021; Chen et al., 2019). In the field of urban landscape rendering, it has been used to detect buildings and greenery from photographs and calculate the greenness ratio (Li et al., 2020; Ki & Lee, 2021).

2.3. Image inpainting
Patch-based (Barnes et al., 2009) and diffusion-based (Li et al., 2017) methods are mainly used for image completion (Elharrouss et al., 2020). The patch-based method finds a small rectangular area in the image that matches the missing area in the image and uses it to fill in the missing area. It can provide high-quality completion for any image size and missing area. However, if no part of the image matches the missing area, it is impossible to complete the image accurately. The diffusion-based method completes the missing region by smoothly propagating the image from the region boundary to its interior. This method has the problem of not being able to represent fine textures. Also, a method has been proposed to remove clouds in photographs by creating a reference image using a data fusion model from multiple cloud-free photographs and statistically processing the image (Li et al., 2019). However, this method can be applied only if cloud-free images are available.
To solve these problems, convolutional neural network (CNN)-based and GAN-based methods using deep learning have been proposed. Pathak et al. (2016) proposed a GAN based on a CNN to fill in missing regions with complementary patterns. The GAN consisted of two networks, namely, a generator and a discriminator (Goodfellow et al., 2014), that are trained while competing with each other: the generator is trained to generate realistic images, while the discriminator is trained to distinguish between real and fake images. By learning in this way, it is possible to generate images that do not actually exist but are “plausible”.

However, the model proposed by Pathak et al. (2016) had a fixed image size (256 × 256 pixels), and also the size, position and shape of the region to be complemented were fixed. In addition, the continuity of the region to be completed with imagery from the surrounding area was not considered. Iizuka et al. (2017) proposed a completion method that considers the continuity between the surrounding areas and the areas to be completed. In this method, when the areas to be completed are small, it is possible to generate an image that considers the continuity with the surrounding areas. In addition, the size, position and shape of the areas to be completed are not fixed. Since then, various methods have been proposed to accurately complement the area to be filled in. Various studies have applied these methods (Wei et al., 2022; Hazra et al., 2022; Li et al., 2022), and they have also been applied to the design of floor plans of buildings (Huang & Zheng, 2018). However, no paper reports whether GANs can be used for DR of buildings.

2.4. Research combining GANs and semantic segmentation
Research combining semantic segmentation and GANs has been conducted in various fields. For example, studies have been done to detect and remove masks covering parts of a person’s face (Ud Din et al., 2021), to remove microphones in front of a face (Khan et al., 2019), and to remove shadows in an image (Javed et al., 2022). In the field of urban landscape, the research described in Section 2.1 was conducted to virtually remove people and other objects in front of buildings (Zhang et al., 2021). However, these studies did not combine real-time communication between devices, nor were they required to perform the processing in real-time.

2.5. Distributed processing with client-server model
The client-server model has been used in various research and application fields as a method to reduce the limitations of and load on devices operated by the user (client device) (Kämäräinen et al., 2018; Soyata et al., 2012; Okada et al., 1993). The client-server model is a processing framework that separates the client device and the device for centralized processing (server device), reducing the load on the client devices. Processing can be performed much faster by using the client-server model than by performing the computations on the mobile device itself.

2.6. Contributions of this research
The proposed method integrates semantic segmentation, a GAN, and real-time communication between devices and makes it possible to detect target objects automatically and demolish them virtually in real-time without preparing a 3D model in advance. By not requiring preparation such as the creation of a 3D model, it is possible to visualise the future landscape after the removal of a structure in real-time onsite, even when time or cost is limited, simply by preparing a personal computer (PC) in advance. Therefore, this research is expected to make the following two contributions: reducing the cost of consensus building for redevelopment planning by visualising
the post-removal landscape using GAN, and shortening the time required for consensus-building and the overall design process by reducing the preparation time. In addition, the ability to check the landscape after removal in real-time onsite during planning can be used as a reference to consider the exterior design of a building after reconstruction may benefit the design field as well. We also believe that it can contribute to the development of DR technology (Davila Delgado et al., 2020), which is needed in many fields, such as building planning.

Previous research on image completion using GANs has mainly focused on proposing GAN models, and there has been insufficient discussion of the quantitative evaluation of the generated results and evaluation of reproducibility. In the field of engineering, we are required to formulate the results and analyse how the results were achieved. In this research, we quantitatively evaluated and verified the feasibility of using GANs for DR. When evaluating the accuracy of the completion, we used an index that is close to human colour vision and a threshold value that is defined as a standard. This makes it easy to understand the meaning of the evaluation results in a quantitative evaluation.

3. Proposed Method
This section gives an overview of the proposed method, followed by details of each component of the DR system.

3.1. Overview
A limitation of previous research was that the 3D model required to virtually demolish existing structures had to be created in advance (Inoue et al., 2018; Kido et al., 2020). To meet this challenge, we propose a method to remove buildings in real-time by semantic segmentation and automatically filling the vacated areas with the output of a GAN without creating 3D models of both the buildings to be removed and their surrounding environment in advance. Furthermore, real-time communication between devices is used to connect a mobile device, such as a smartphone, tablet or notebook (hereinafter, the “client device”) and a PC equipped with a high-performance graphics processing unit (GPU) (hereinafter, the “server PC”) to perform these processes.
Figure 1 Workflow of the proposed method

High processing performance is required for automatic detection of objects and automatic completion of missing areas. However, high-performance PCs are generally not designed to be portable, so it is not possible to bring them to a site to perform image processing. Therefore, the client device and the server PC communicate via the internet. The client device requires two things: it must be equipped with a camera for acquiring the landscape, and it must be able to connect to the internet. This makes it possible to perform the study easily onsite.

The order of execution is shown below.

1. Establish a connection between the client device and the server PC.
2. Obtain the current view (hereinafter, referred to as the “input frame”) using a web camera in or connected to the client device.
3. Send the input frame from the client device to the server PC.
4. At the server PC, detect the object to be removed in the received input frame using semantic segmentation.
5. Perform mask processing based on the detection results and generate masks automatically.
6. Input the mask and input frame to the GAN, and complete (fill in) the missing area using the information from the image around it.

7. Send the completion result (hereinafter referred to as the “output frame”) from the server PC to the client device.

8. Display the received output frame on the screen of the client device.

   The output frame is the future landscape without the object to be removed (e.g., a building slated for demolition).

3.2. Detecting the target object

The detection of objects to be removed requires a drawing speed fast enough for real-time visualisation of the future landscape. In addition, high detection accuracy is required because the area to be replaced is based on the detection results. Furthermore, it is necessary to detect buildings and remove objects automatically without any prior preparation. Therefore, semantic segmentation (Long et al., 2015) is used to automatically detect the object to be removed without prior preparation. In addition, semantic segmentation can fill the area of the detected object with one specified colour. By setting different colours for the area of the object to be removed and the other areas to be filled in, it is possible to automatically generate the mask needed to remove the object with the GAN in subsequent processing.

3.3. Providing complementary fill for the area of the target object

A GAN (Goodfellow et al., 2014) is used to complete the area where the removed object existed (hereinafter referred to as the “completion target area”). The GAN makes it possible to automatically complete the target area according to the surrounding areas without prior preparation. In this process, the GAN recognises the completion target area based on the mask automatically generated by the semantic segmentation. The resulting mask is then superimposed on the input frame to create the future landscape after the object to be removed is eliminated.

3.4. Communication between devices over the internet

Real-time communication between the client device and the server PC via the internet is used to send and receive input and output frames between the client device and the server PC. The client device obtains the current scene of the real space using a camera connected to the client device and sends it to the server PC as an input frame. The server PC generates an output frame based on the received input frame and sends it to the client device. This makes it possible to visualise the future landscape in real-time at the redevelopment site by only taking a client device to the site, even for visualisation methods that require heavy processing loads.

4. Verification

In this section, we describe the verification method used in the experiments to demonstrate the usefulness of the proposed method and provide the results. The detail of the prototype system which is used in verifications is shown in Appendix A.
4.1. Making custom weights

We obtained images from GSV using the Google application programming interface. The details of the obtained images are shown in Table 1. The target area of these images was the entire Osaka prefecture in Japan, which includes the experimental area. While obtaining 5,500 images, we randomly generated values for the image acquisition point, camera heading and camera pitch. All obtained images were used to train the GAN and create the weights. Using weights learned from the ImageNet dataset (Russakovsky et al., 2015) and those learned from the GSV dataset created in this study, we evaluated whether the completion accuracy varied with the GAN training dataset, as described in Section 4.2.

Table 1 Image information

| Area       | Image size 512 × 512 pixels |
|------------|-----------------------------|
| Number of images | 5,500                      |
| Longitude  | E 135°6′0″ – E 135°42′0″    |
| Latitude   | N 34°13′48″ – N 35°2′0″     |
| Camera heading | Random (0°–360°)          |
| Camera pitch | Random (0°–60°)            |
| Camera field of view | 90° (default)        |

4.2. Verification of GAN completion accuracy

We verified the completion accuracy of the GAN depending on the ratio of the masked area to the whole image (called the “mask ratio”) and the elements to be completed. We also verified how the completion accuracy changes depending on the dataset used for training the GAN. In this way, we found the conditions under which good completion results can be obtained and verified how much the completion accuracy can be improved by using the landscape-specific dataset (GSV) created in Section 4.1.

4.2.1. Verification methodology of GAN completion accuracy

It is difficult to prepare many landscape images of existing buildings in the real world both before and after actual removal while changing the size of the object to be removed and the parameters of the surrounding landscape. Therefore, for verification, we created masks manually by imitating the masks generated by semantic segmentation, completed them with the GAN, and compared the output images with the captured photos (correct images). We used GIMP (The GIMP team, n.d.), an image editing software, to manually create the mask area. This method makes it possible to verify the results under a variety of condition combinations. In this verification, five photos were prepared, and five mask images with different mask percentages (up to 20%) were prepared for each photo.

The verification process is as follows:

1. Take a picture of the landscape.
2. Create masks manually.
3. Input the photo and the created mask image into the GAN.
4. Compare the output image with the correct image and evaluate the completion accuracy.
To compare and evaluate the output image and the correct image, the filled-in area was used as the evaluation target area, and only that area was compared, not the entire image. $\Delta E^*$ (Backhaus et al., 1998), which is used as a standard for colour reproduction on PCs and monitors (International Organization for Standardization, 2015), was used as an index to evaluate the completion accuracy. The closer this value is to zero, the more the two colours are similar. In the evaluation, the threshold values of $\Delta E_{00}^*$ were set to 6.5 and 13.0. When $\Delta E_{00}^*$ is higher than 13.0, the two colours being compared give the impression of being distinctly different. In other words, when $\Delta E_{00}^*$ is less than 13.0, we get the impression that the two colours are similar. When $\Delta E_{00}^*$ is less than 6.5, we get the impression that the two colours are the same. In this verification, $\Delta E_{00}^*$ was calculated for each pixel, and the percentage of pixels with $\Delta E_{00}^*$ of 6.5 or less and the percentage of pixels with $\Delta E_{00}^*$ of 13.0 or less in the area to be evaluated were calculated to evaluate the completion accuracy. The detail of $\Delta E^*$ and determine process of threshold values are described in Appendix. B.

4.2.2. Results of GAN completion accuracy

The results of the completion of the mask area using the GAN and the evaluation of the completion accuracy are shown below.

Figure 2 shows the results of the completion using GSV, and Figure 3 shows the results of the completion using ImageNet. Some of the results obtained by the validation are shown in Figure 4.
Figure 3 shows that the ratios of $\Delta E_{00}$ both less than 13.0 and less than 6.5 tend to decrease as the mask ratio increases. The same trend is observed in Figure 2, except for the case where the background elements are trees and buildings. In Figure 2 tree and Figure 3 sky-object, when the mask percentage is over 15%, the ratio of pixels with $\Delta E_{00}$ less than 13.0 is less than 50%, indicating that more than half of the pixels are images of a different colour. These results show that as the mask ratio increases, the percentage of pixels that people can
recognise as different colours increases. This is due to the failure to correctly estimate the landscape in the completion area from the surrounding area as the mask ratio increases (Figure 4c).

Focusing on the relationship between the elements around the fill-in area and the completion accuracy shows no clear difference. However, focusing on the appearance shows that if a regular pattern, such as a grid, is not continuous in and around the fill-in area, the result is unrealistic (Figure 5b).

Figure 5 Results that do not match the surrounding pattern (background: sky-object; mask ratio: 4.3%)

The completion accuracy differs depending on the dataset used for training, as shown in Figures 2 and 3. In particular, when the completion area is the sky, the weights learned using GSV are better than those learned using ImageNet. In contrast, when the complementary area is a tree, the results are worse, as shown by the comparison between Figure 4d and the tree results in Figures 2 and 3.

4.3. Field verification

In this section, we describe a field verification using the developed prototype system. In this verification, we confirmed the accuracy of removing the target object and measured the execution speed and latency. To demonstrate the usefulness of the proposed method, we verified the detection accuracy of the object to be removed by semantic segmentation and the completion accuracy of the removal area by the GAN. For the execution speed, we measured the time spent processing one frame and calculated the rate in frames per second.

The method and results of the verification are described below.

4.3.1. Methodology of field verification

First, we describe the method for verifying the accuracy of removing the target object. In this verification, the client device sends an input frame to the server PC every 2 sec, and the server PC saves the received input frame, the generated output frame and the mask image generated from the input frame using semantic segmentation. The interval between transmissions is set to 2 sec because if the interval between transmissions from the client device is shorter than the time required to process one frame, the communication between devices becomes unstable and loses accuracy. The time required to process one frame by the semantic segmentation and GAN used in this verification was considered. For the verification of the detection accuracy of the object to be removed, (1) the correct mask image was created manually based on the saved input frame and (2) the intersection over union (IoU) was calculated by comparing the correct mask image with the saved one. The formula for calculating the IoU is shown in Appendix. C. In the verification of the accuracy of the completion of the removal area, we (1) manually created a correct image after eliminating the object to be removed from the saved input frame, (2) compared the output frame with the created correct image, and (3) calculated $\Delta E_{\text{IoU}}$. In this verification, the image
quality was not degraded due to communication between the client device and the server PC because the images were sent and received in binary format and the image size was not changed.

Next, we describe the verification method for execution speed and latency. From the start of verification to the end of verification on the server PC, we (1) recorded the time when the input image was received ($t_o$), (2) recorded the time when the output image was sent ($t_e$), and (3) obtained the time required for processing (processing time $T$) by subtracting $t_o$ from $t_e$. In the verification of latency, on the client device side, we (1) took a picture of a stopwatch with a millisecond display, (2) measured the lag by comparing the time of the stopwatch on the screen sent from the server PC with the time of the stopwatch on the screen sent to the server PC, and (3) calculated the average value of the lag measured after the end of the process. The formulas for these calculations are shown in Appendix D. Figure 6 shows how the latency was measured.

![Figure 6 (a) Verification scene, (b) verification display](https://example.com/figure6.png)

**Figure 6 (a) Verification scene, (b) verification display**

### 4.3.2. Results of field verification

The conditions of the verification are shown in Table 2 and Figure 7 for Target Object 1, and in Table 2 and Figure 8 for Target Object 2.

| Table 2 Conditions of verification |
|-------------------------------------|
| Date and time | Target Object 1 | Target Object 2 |
| Date and time | 15 Sep. 2021, 14:00~ | 15 Sep. 2021, 15:00~ |
| Weather | Sunny | Sunny |
| Distance from viewpoint to target object (m) | 52 | 172 |
| Internet speed (Mbps) | Server PC | Mobile device |
| Server PC | 530 | 18.8 |
| Mobile device | 530 | 15.1 |
First, for the verification of detection accuracy, the calculated IoU change is shown in Figure 9. In the verification for Target Object 1, the mean IoU was 0.914 and its standard deviation was 0.032. In the verification of Target Object 2, the mean IoU was 0.826 with a standard deviation of 0.088. One of the reasons for the difference in detection accuracy between Target Object 1 and Target Object 2 is the influence of the size of the target object on the screen. When the object to be detected was small, the detection accuracy decreased, and the detection accuracy of Target Object 2 by semantic segmentation, which occupied a small percentage of the screen, was not stable.
Figure 9 IoU results: (a) Target Object 1, (b) Target Object 2

Figure 10 shows the percentage of pixels with $\Delta E_{\text{00}}^*$ less than the threshold value in the entire completion target area. Both 6.5 and 13.0 were used as the threshold values for $\Delta E_{\text{00}}^*$, as described in Section 4.2.1. In the verification for Target Object 1, the average percentage of pixels with $\Delta E_{\text{00}}^*$ less than 6.5 was 3.9%, and the average percentage of pixels with $\Delta E_{\text{00}}^*$ less than 13.0 was 15.3%. In the verification of Target Object 2, the average percentage of pixels with $\Delta E_{\text{00}}^*$ less than 6.5 was 37.3%, and the average percentage of pixels with $\Delta E_{\text{00}}^*$ less than 13.0 was 52.2%. One of the reasons for the large variation in the completion accuracy of Target Object 2 is the variation in the detection accuracy of the object to be removed (Figure 11). The GAN fills in the completion target area based on a mask automatically generated using semantic segmentation. Therefore, areas that are not identified as objects to be removed are not removed by the GAN.

Figure 10 (a) Results for Target Object 1, (b) results for Target Object 2
Some results for the correct images and the input frames are shown in Figure 12. Correct images were created manually based on the frames saved during the verification using GIMP (The GIMP Team, n.d.). Target Object 1 covers 13.0% of the screen in Figure 12a, indicating that the completion target area is large. As described in Section 4.2.2, the completion accuracy decreases as the target area becomes larger. Therefore, the cause of the poor completion accuracy shown in Figure 10a is the size of the completion target area. The results shown in Figure 12b also indicate the cause of the difference in completion accuracy, even with the same level of IoU in the graph shown in Figure 11. In Figure 12b, the IoU of the 10th and 16th frames is about 0.9, but the percentage of pixels with $\Delta E_{oo}^*$ less than 6.5 in the entire completion target area is about 70% in the 10th frame, while it is about 60% in the 16th frame. This is because areas that were not detected correctly as areas to be removed affected the completion process. The GAN decides how to complete the target area based on the area surrounding it. Therefore, if all the areas containing the object to be removed are not the completion target area, the object will not be eliminated, as shown in the output frame of Figure 12b.
Next, Figure 13 shows the change in processing time measured on the server PC during execution. The average processing time measured for each frame was 0.23 sec for Target Object 1 and 0.16 sec for Target Object 2. However, because the processing time was unstable immediately after the system was started and varied widely, it is not included in the average processing time. In this verification, when the difference between the processing time before and after system startup was less than 0.1 sec, it was judged that the processing time had stabilized. Considering this, in the case of Target Object 1, the process was judged to be stable after the 12th frame, and the average processing time was 0.21 sec. In the same way, for Target Object 2, it was judged to be stable after the 12th frame, and the average processing time was 0.14 sec. The average drawing time is the reciprocal of the average processing time, so the average drawing time for Target Object 1 was 4.76 fps and for Target Object 2 was 7.14 fps.

Next, the latency measured on the client device is shown in Figure 14. The results of the latency measurement showed that it took a maximum of 0.93 sec, a minimum of 0.63 sec, and an average of 0.77 sec to
transmit an image from the client device, process the image, return the modified image to the client, and display the results. Subtracting the average processing time calculated from the processing speed measured for each frame (the average of the average processing times measured for Target Object 1 and Target Object 2) from this average latency showed that the latency due to communication was approximately 0.60 sec, as determined from the average measured latency.

![Figure 14 Latency between input frame and output frame](image)

5. Discussion

The proposed method can automatically visualise the future landscape after the demolition and removal of existing buildings without any prior preparation. However, some problems were identified. In this section, we describe the benefits and limitations of the proposed method based on the results obtained in the verification experiments.

5.1. GAN completion accuracy

The completion accuracy of the GAN was evaluated by $\Delta E_{00}$ (Backhaus et al., 1998). The evaluation results (Figure 4 and Figure 10) and output images (Figure 4 and Figure 12) indicate that $\Delta E_{00}$-based evaluation correlates with human visual perception. $\Delta E_{00}$ can be used as an evaluation index because it provides absolute evaluation criteria useful as a standard for evaluating the difference between two colours.

The results of the evaluation by $\Delta E_{00}$ indicate that the accuracy of the completion of the target area by the GAN depends on the mask ratio (Figures 2 and 3). In addition, the appearance of the hazy area increases as the mask ratio increases (Figure 4c). This means that when the completion target area is large, the completion accuracy deteriorates and the appearance becomes uncomfortable. Also, if a regular pattern, such as a lattice, in both the completion area and its surrounding area do not match, the result is uncomfortable (Figure 5). However, these challenges are expected to be solved by the proportion of landscape types included in the photos of the dataset used for training the GAN. By adding photos of various building facades to the various landscape elements, it will be possible to create weights that can also correctly complement the patterns on building surfaces.

Next, we discuss the effect of the GAN dataset on the completion accuracy and compare the completion results of the GSV dataset with those of the ImageNet dataset. When the completion area is the sky, the GSV results are better than those of the ImageNet dataset, but when buildings are included, the results are worse.
(Figures 2 and 3). When we inspected the photos in the prepared GSV dataset, we found that 5,198 out of 5,500 photos included the sky. From this, it was inferred that the weights were specialized for sky completion. Therefore, the differences that occurred for each element around the completion area in this verification are inferred to be due to the difference in the percentage of landscape elements in the photos included in the dataset used for training.

When the object to be removed occupied more than 15% of the entire screen, the percentage of pixels with $\Delta E_{00}$ less than 13.0 in the completion area was less than 50%, depending on the background elements (Figure 2 tree and Figure 3 sky-object). Therefore, when the object to be removed occupies a high percentage of the screen, virtual removal by the proposed method is difficult. To reduce the limitation due to the percentage of the screen occupied by the object to be removed, a method is required to allow the GAN to focus on the target area during completion or to reduce the percentage of the overall completion target area.

5.2. Field verification

It is possible to detect an object to be removed by semantic segmentation and complete the target area using a GAN. This means that the proposed method can virtually remove an object without any prior preparation. However, if the object to be removed could not be detected accurately, the completion accuracy and appearance of the removal result were poor (Figures 11 and 12b). This indicates that it is necessary for the object to be large enough (percentage of the screen) to be detected by semantic segmentation. If a part of the object to be removed cannot be detected from the output frame, as shown in Figure 15, it cannot be removed by the semantic segmentation and GAN. The solution to this challenge is to define an area slightly larger than the area where the object to be removed is detected using semantic segmentation as the completion target area.

The advantages of the proposed method are next described in comparison with previous methods (Inoue et al., 2018; Kido et al., 2020). Previous methods have slow processing speeds because the amount of data in the model increases as the area under study grows larger. In contrast, the proposed method does not use a 3D model, so the amount of data does not change and the processing speed can be kept constant. In addition, previous methods require prior preparation of a 3D city model created using SfM or provided to the public and alignment of the real and virtual worlds before execution. However, the proposed method does not require these tasks, so it reduces the time needed for pre-processing.

The accuracy of the landscape after virtual removal is a challenge for the proposed method because it visualises the future landscape using GAN-generated plausible backgrounds after virtual removal rather than 3D models. Also, as mentioned in Section 5.1, the completion accuracy for the completion target area is influenced by the size of the object to be removed. As Figure 9a, Figure 10a and Figure 12a indicate, if the object to be removed occupies a large percentage of the screen, then the completion accuracy becomes poor.
The latency of input and output frames displayed on client devices can be improved by increasing the communication speed between devices. As described in Section 4.3.2, the communication time accounts for most of the identified latency (Figures 13 and 14). To increase the speed of communication, the next-generation communication standards, such as 5G and Wi-Fi 6, can be used for the client device communication, and fast Ethernet with communication speeds exceeding 1 Gbps can be used for the server communication. For example, if the client device used in the verification communicating at Wi-Fi 4 (maximum speed of 600 Mbps) (IEEE, 2021) is replaced with a device that communicates at Wi-Fi 6 (maximum speed of 9,600 Mbps) (IEEE, 2021), it will be able to communicate 16 times faster.

One limitation of the proposed DR method is that it cannot be applied to structures surrounded by buildings. The semantic segmentation used to detect objects in the proposed method fills different objects with the same colour if they are in the same category. Therefore, if both removable and non-removable objects are in the same category, all are identified as removable objects (Figure 16b). This problem can be solved by using instance segmentation (He et al., 2017) instead of semantic segmentation. Semantic segmentation performs labelling for each type of object, while instance segmentation performs labelling for each object. Therefore, even when objects of the same type overlap, they can be labelled as separate objects. However, no dataset exists for instance segmentation. Therefore, to detect buildings using instance segmentation, it is necessary to annotate a large number of images that contain buildings.

![Figure 16 Example of removing an object that is not the target of removal](image)

**Figure 16 Example of removing an object that is not the target of removal**

**6. Conclusion**

The objective of this research is to develop a method for detecting and virtually removing representations of existing buildings from a video stream in real-time to be able to visualise a future scenario without these buildings. This is done by semantic segmentation, which eliminates the need to create 3D models of the buildings and the surrounding scenery, and a GAN. This method solves the challenge of preparing 3D models of the buildings to be removed and their surrounding landscape in advance, which is a limitation in previous research (Inoue et al., 2018; Kido et al., 2020). In addition, by integrating the real-time communication between devices, the future landscape after the demolition and removal of buildings can be visualised in real-time by simply bringing a portable device to the location. Evaluation of the accuracy of the GAN using \( \Delta E^* \) found that this proposed method can be used to generate future landscapes that are visually pleasing. In redevelopment projects in existing urban areas, the proposed method enables stakeholders to intuitively understand the future vision and reduces the time required to reach consensus. Also, by reducing the time required for consensus building, the overall design work time can be reduced. The proposed method will also contribute to the development of DR technology.
(Davila Delgado et al., 2020), which is needed in many fields, such as architectural planning. Future research is expected to include the development of an advanced landscape simulation system with the proposed method combined with AR to visualise the future landscape after reconstruction.

The conclusions of this study are as follows:

- By combining semantic segmentation and a GAN, automatic visualisation of the future landscape can be achieved without any prior preparation, such as creating 3D models.
- By employing real-time communication between the server PC and client device, virtual demolition of objects to be removed is possible by only bringing a mobile device to the redevelopment site.
- If the object to be removed occupies less than 15% of the screen, the percentage of the target completion area with $\Delta E_{0*} > 13.0$ is more than 50%, making the completion result less visually satisfying.
- The proposed method visualises the future landscape with an average rendering speed of 5.71 fps.

Future work is replacing semantic segmentation with instance segmentation. With instance segmentation, the proposed method can be used even when the same type of objects to be removed and objects to remain exist in the study area.

**Author contributions**

Takuya Kikuchi: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Writing - Review & Editing, Visualisation.

Takayoshi Fukuda: Conceptualization, Methodology, Formal analysis, Resources, Writing - Review & Editing, Visualisation, Project administration, Funding acquisition.

Nobuyoshi Yabuki: Methodology, Resources, Writing - Review & Editing, Supervision.

**Acknowledgements**

This research has been partly supported by the Japan Society for the Promotion of Science, KAKENHI Grant Number JP19K12681. All images were created by the authors from 2020 to 2021.

**References**

Abualdenien, J., & Borrmann, A. (2020). Vagueness visualization in building models across different design stages. Advanced Engineering Informatics, 45, Article 101107. https://doi.org/10.1016/j.aei.2020.101107.

Backhaus, W., Kliegl, R., & Werner, J. S. (1998). Color Vision: Perspectives from Different Disciplines. Walter de Gruyter, 188. ISBN 9783110154313.

Barnes, C., Shechtman, E., Finkelstein, A., & Goldman, D.B. (2009). PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing. ACM Transactions on Graphics, 28, 3, Article 24. https://doi.org/10.1145/1531326.1531330.

Bishop, I.D., & Lange, E. (2005). Visualization in landscape and environmental planning. Technology and applications. ISBN 9780415305105.

Caudell, T.P., & Mizell, D.W. (1992). Augmented reality: an application of heads-up display technology to manual manufacturing processes. Proceedings of the Twenty-Fifth Hawaii International Conference on System Sciences, 2, 659-669. https://doi.org/10.1109/HICSS.1992.183317.
Chao, P., Kao, C.-Y., Ruan, Y., Huang, C.-H., & Lin, Y.-L. (2019). HarDNet: A low memory traffic network. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 3551-3560. https://doi.org/10.1109/ICCV.2019.00365.

Chen, B., Gong, C., & Yang, J. (2019). Importance-Aware Semantic Segmentation for Autonomous Vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 20, 1, 137-148. https://doi.org/10.1109/TITS.2018.2801309.

Cordts, M., Omran, M, Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., & Schiele, B. (2016). The Cityscapes Dataset for Semantic Urban Scene Understanding. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3213-3223. https://doi.org/10.1109/CVPR.2016.350.

Davila Delgado, J.M., Oyedele, L., Demian, P., & Beach, T. (2020). A research agenda for augmented and virtual reality in architecture, engineering and construction. *Advanced Engineering Informatics*, 45, Article 101122. https://doi.org/10.1016/j.aei.2020.101122.

Elharrouss, O., Almaadeed, N., Al-Maadeed, S. (2020). Image Inpainting: A Review. *Neural Process Lett.*, 51, 2007-2028. https://doi.org/10.1007/s11063-019-10163-0.

Eskandari, R., & Motamedi, A. (2021). Diminished reality in architectural and environmental design: Literature review of techniques, applications, and challenges. *Proceedings of the International Symposium on Automation and Robotics in Construction*, 2021-November, 995-1001. https://doi.org/10.22260/ISARC2021/0134.

Flores, A., & Belongie, S. (2010). Removing pedestrians from Google Street View images. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops (CVPRW 2010)*, 53-58. https://doi.org/10.1109/CVPRW.2010.5543255.

Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. *Advances in Neural Information Processing Systems*, 3, 2672-2680. https://proceedings.neurips.cc/paper/2014/file/5c9e396b22b618f06494c97b1afecc3-Paper.pdf.

Google. (n.d.). *Street View*. Google Developers. Retrieved October 16, 2021, from https://developers.google.com/streetview.

Han, B., Ma, J.W., & Leite, F. (2021). A framework for semi-automatically identifying fully occluded objects in 3D models: Towards comprehensive construction design review in virtual reality. *Advanced Engineering Informatics*, 50, Article 101398. https://doi.org/10.1016/j.aei.2021.101398.

Hartmann, T., & Trappey, A. (2020). Advanced Engineering Informatics - Philosophical and methodological foundations with examples from civil and construction engineering. *Developments in the Built Environment*, 4, Article 100020. https://doi.org/10.1016/j.dibe.2020.100020.

Hazra, D., Shafqat W., & Byun, Y.-C. (2022). Generating synthetic data to reduce prediction error of energy consumption. *Computers, Materials and Continua*, 70, 2, 3151-3167. https://doi.org/10.32604/cmc.2022.020143.

He K., Gkioxari G., Dollar P., & Girshick R. (2017). Mask R-CNN. *2017 IEEE International Conference on Computer Vision (ICCV)*, 2980-2988. https://doi.org/10.1109/ICCV.2017.322.

Huang, W., & Zheng, H. (2018). Architectural drawings recognition and generation through machine learning. *Recalibration on Imprecision and Infidelity - Proceedings of the 38th Annual Conference of the Association*.
IEEE. (2021). IEEE Standard for Information Technology--Telecommunications and Information Exchange between Systems - Local and Metropolitan Area Networks--Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. In IEEE Std 802.11-2020 (Revision of IEEE Std 802.11-2016), 1-4379. https://doi.org/10.1109/IEEESTD.2021.9363693.

Iizuka, S., Simo-Serra, E., & Ishikawa, H. (2017). Globally and locally consistent image completion. ACM Transactions on Graphics, 36, 4, Article 107. https://doi.org/10.1145/3072959.3073659.

Inoue, K., Fukuda, T., & Yabuki, N. (2018). Tracking Robustness and Green View Index Estimation of Augmented and Diminished Reality for Environmental Design: PhotoAR+DR 2017 project. Proceedings of the 23rd International Conference on Computer-Aided Architectural Design Research in Asia (CAADRIA 2018), 339-348. https://doi.org/10.52842/conf.caadria.2018.1.339.

International Commission on Illumination. (2014). Colorimetry — Part 6: CIEDE2000 Colour-difference formula (ISO/CIE Standard No. 11664-6:2014), https://www.iso.org/standard/63731.html.

International Organization for Standardization. (1988). Black leads for wood-cased pencils — Classification and diameters (ISO Standard No. 9180:1988). https://www.iso.org/standard/16787.html.

International Organization for Standardization. (2008). Colorimetry — Part 4: CIE 1976 L*a*b* Colour space (ISO Standard No. 11664-4:2008). https://www.iso.org/standard/52497.html.

International Organization for Standardization. (2015). Graphic technology — Displays for colour proofing — Characteristics (ISO Standard No. 12646:2015). https://www.iso.org/standard/57311.html.

Japanese Industrial Standard Committee. (2007). Water colour paints (JIS Standard No. S6028:2007). https://webdesk.jsa.or.jp/books/W11M0070/index.

Japanese Industrial Standard Committee. (2020). Pencils, coloured pencils and leads for them (JIS Standard No. S6006:2020). https://webdesk.jsa.or.jp/books/W11M0070/index.

Javed, K., Ud Din, N., Hussain, G., & Farooq, T. (2022). Throwaway Shadows Using Parallel Encoders Generative Adversarial Network. Applied Sciences, 12(2), 824. https://doi.org/10.3390/app12020824.

Kämäräinen, T., Eerikäinen, J., Siekkinen, M., & Ylä-Jääski, A. (2018). CloudVR: Cloud Accelerated Interactive Mobile Virtual Reality. MM 2018 - Proceedings of the 2018 ACM Multimedia Conference, 1181-1189. https://doi.org/10.1145/3240508.3240620.

Khan, M. K. J., Ud Din, N., Bae, S., & Yi, J. (2019). Interactive Removal of Microphone Object in Facial Images. Electronics, 8(10), 1115. https://doi.org/10.3390/electronics8101115.

Ki, D., & Lee, S. (2021). Analyzing the effects of Green View Index of neighborhood streets on walking time using Google Street View and deep learning. Landscape and Urban Planning, 205, Article 103920. https://doi.org/10.1016/j.landurbplan.2020.103920.

Kido, D., Fukuda, T., & Yabuki, N. (2020). Diminished reality system with real-time object detection using deep learning for onsite landscape simulation during redevelopment. Environmental Modelling and software, 131, Article 104759. https://doi.org/10.1016/j.envsoft.2020.104759.

Kido, D., Fukuda, T., & Yabuki, N. (2021). Assessing future landscapes using enhanced mixed reality with semantic segmentation by deep learning. Advanced Engineering Informatics, 48, Article 101281. https://doi.org/10.1016/j.aei.2021.101281.
Kikuchi, T., Fukuda, T., & Yabuki, N. (2021). Automatic diminished reality-based virtual demolition method using semantic segmentation and generative adversarial network for landscape assessment. *Proceedings of the 39th eCAADe Conference, 2*, 529-538. https://doi.org/10.52842/conf.ecaade.2021.2.529.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *the Advances in Neural Information Processing Systems, 1*, 1097-1105. https://doi.org/10.5555/2999134.2999257.

Li, F., Yu, Z., & Qin, C. (2022). GAN-based spatial image steganography with cross feedback mechanism. *Signal Processing, 190*, Article 108341. https://doi.org/10.1016/j.sigpro.2021.108341.

Li, H., Luo, W., & Huang, J. (2017). Localization of diffusion-based inpainting in digital images. *IEEE Transactions on Information Forensics and Security, 12*(12), Article 7987733, 3050-3064. https://doi.org/10.1109/TIFS.2017.2730822.

Li, Q., Shi, Y., Auer, S., Roschlaub, R., Möst, K., Schmitt, M., Glock, C., & Zhu, X.X. (2020). Detection of undocumented building constructions from official geodata using a convolutional neural network. *Remote Sensing, 12*, 21, Article 3537, 1-21. https://doi.org/10.3390/rs12213537.

Li, X., Wang, L., Cheng, Q., Wu, P., Gan, W., & Fang, L. (2019). Cloud removal in remote sensing images using nonnegative matrix factorization and error correction. *ISPRS Journal of Photogrammetry and Remote Sensing, 148*, 103-113. https://doi.org/10.1016/j.isprsjprs.2018.12.013

Lin, Y.-C., Chen, Y.-P., Yien, H.-W., & Su, Y.-C. (2018). Integrated BIM, game engine and VR technologies for healthcare design: A case study in cancer hospital. *Advanced Engineering Informatics, 36*, 130-145. https://doi.org/10.1016/j.aei.2018.03.005.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C., & Berg, A. (2016). SSD: Single Shot MultiBox Detector. *In: Proceedings of the 14th European Conference on Computer Vision (ECCV 2016)*, 9905, 21-37. https://doi.org/10.1007/978-3-319-46448-0_2.

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 431-440. https://doi.org/10.1109/CVPR.2015.7298965.

Müller, D., & Kramer, F. (2021). MIScnn: a framework for medical image segmentation with convolutional neural networks and deep learning. *BMC Medical Imaging, 21*, 1, Article 12. https://doi.org/10.1186/s12880-020-00543-7.

Mann, S. (eds) (1994). Mediated Reality. *M.I.T. Media Lab Perceptual Computing Section Cambridge, Technical Report TR 260.*

Mori, S., Ikeda, S., & Saito, H. (2017). A survey of diminished reality: Techniques for visually concealing, eliminating, and seeing through real objects. *IPSJ Transactions on Computer Vision and Applications, 9*. https://doi.org/10.1186/s41074-017-0028-1.

Mutis, I., & Ambekar, A. (2020). Challenges and enablers of augmented reality technology for in situ walkthrough applications. *Journal of Information Technology in Construction, 25*, 55-71. https://doi.org/10.36680/j.iticon.2020.003.

Navab, N., Bani-Kashemi, A., & Mitschke, M. (1999). Merging visible and invisible: Two Camera- Augmented Mobile C-arm (CAMC) applications. *Proceedings 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR'99), 134-141*. https://doi.org/10.1109/IWAR.1999.803814.
Okada, M., Yokoi, S., & Toriwaki, J. (1993). A study of distributed environment for three-dimensional image processing using extended server-client model. *Systems and Computers in Japan, 24*, 4, 13-22. https://doi.org/10.1002/scj.4690240402.

Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., & Efros, A.A. (2016). Context Encoders: Feature Learning by Inpainting. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2536-2544. https://doi.org/10.1109/CVPR.2016.278.

Pejic, P., Lakicevic, M., Krasic, S., & Predrag, S. (2017). Application of Augmented and Virtual Reality in Residential Complex Presentation, Case Study: Energoprojekt Sunnyville. *Journal of Industrial Design and Engineering Graphics, 12*, 1, 127-132. http://sorgin.ro/jideg/index.php/jideg/article/view/160/156.

Rød, J. K., & van der Meer, D. (2009). Visibility and Dominance Analysis: Assessing a High-Rise Building Project in Trondheim. *Environment and Planning B: Planning and Design, 36*, 4, 698-710. https://doi.org/10.1068/b34118.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., & Li, F.F. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision, 115*, 3, 211-252. https://doi.org/10.1007/s11263-015-0816-y.

Schnabel, M. A., Kobayashi, Y., Pencreath, Y., Bennadji, A., Choi, D., Fiamma, P., Fukuda, T., Lo, T.T., Narahara, T., Novak, M., Ron, R., Swarts, M., Terzidis, K., Tucker, T., & Vital, R. (2021). Virtual World16 - Virtual Design Collaboration for the Intersection of Academia and Industry. *Proceedings of the 26th International Conference of the Association for Computer-Aided Architectural Design Research in Asia, CAADRIA 2021*, 2, 203 – 212. https://doi.org/10.52842/conf.caadria.2021.2.203.

Sharma, G., Wu, W., & Dalal, E.N. (2005). The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. *Color Research and Application, 30*, 1, 21-30. https://doi.org/10.1002/col.20070.

Sheppard, S.J. (1989). Visual Simulation: A Users Guide for Architects, Engineers, and Planners. *356 Van Nostrand Reinhold*. ISBN 9780442278274.

Soyata, T., Muraleedharan, R., Funai, C., & Kwon, M. (2012). Cloud-Vision: Real-time face recognition using a mobile-cloudlet-cloud acceleration architecture. *2012 IEEE Symposium on Computers and Communications (ISCC)*, 59-66. https://doi.org/10.1109/ISCC.2012.6249269.

The GIMP team. (n.d.). *GIMP-GNU IMAGE MANIPULATION PROGRAM*. GIMP. Retrieved October 16, 2021, from https://www.gimp.org/.

Ud Din, N., Javed, K., Bae, S., & Yi, J. (2020). A Novel GAN-Based Network for Unmasking of Masked Face, in *IEEE Access, 8*, 44276-44287. https://doi.org/10.1109/ACCESS.2020.2977386.

Ullman, S. (1979). The interpretation of structure from motion. *Proceedings of the Royal Society of London. Series B. Biological Sciences, 203*, 1153, 405-426. https://doi.org/10.1098/rspb.1979.0006.

Wang, Y., Zhou, Q., Xiong, J., Wu, X., & Jin, X. (2019). ESNet: An efficient symmetric network for real-time semantic segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 11858*, 41-52. https://doi.org/10.1007/978-3-030-31723-2_4.

WebRTC. (n.d.). WebRTC. Retrieved October 16, 2021, from https://webrtc.org/.
Wei, K., Li, T., Huang, F., Chen, J., & He, Z. (2022). Cancer classification with data augmentation based on generative adversarial networks. *Frontiers of Computer Science, 16*, 2, Article 162601. https://doi.org/10.1007/s11704-020-0025-x.

Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., & Huang, T.S. (2018). Generative Image Inpainting with Contextual Attention. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5505-5514. https://doi.org/10.1109/CVPR.2018.00577.

Zeltzer, D. (1992). Autonomy, Interaction, and Presence. *Presence: Teleoperators and Virtual Environment, 1*, 1, 127-132. https://doi.org/10.1162/pres.1992.1.1.127.

Zhang, J., Fukuda, T., & Yabuki, N. (2021). Automatic object removal with obstructed facades completion using semantic segmentation and generative adversarial inpainting. *in IEEE Access, 9*, 117486-117495. https://doi.org/10.1109/ACCESS.2021.3106124.

Zhao, H., Qi, X., Shen, X., Shi, J., & Jia, J. (2018). ICNet for Real-Time Semantic Segmentation on High-Resolution Images. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 11207*, 418-434. https://doi.org/10.1007/978-3-030-01219-9_25.

Zokai, S., Esteve, J., Genc, Y., & Navab, N. (2003). Multiview paraperspective projection model for diminished reality. *The Second IEEE and ACM International Symposium on Mixed and Augmented Reality*, 217-226. https://doi.org/10.1109/ISMAR.2003.1240705.
Appendix

A. The detail of prototype system

A prototype system was developed; we used HarDNet (Chao et al., 2019) for semantic segmentation and generative image inpainting with contextual attention (Yu et al., 2018) for the GAN. The reason for using HarDNet is that the mean IoU, which evaluates the detection accuracy, is as high as 75.9% and the rendering speed is as fast as 53 fps, which is suitable for a method that requires both detection accuracy and fast rendering. In addition, HarDNet is prepared with weights trained on the Cityscapes dataset (Cordts et al., 2016), so it is suitable for the proposed method, which needs to detect buildings and remove objects without any prior preparation. The reason for using generative image inpainting with contextual attention is that it takes only 0.2 sec to execute the entire program for a single image (when applied to an area 512 pixels square), and it is designed to reduce the consumption of video random access memory. The low memory usage allows users to use a lower-performance GPU to run the proposed method. Web real-time communication (WebRTC, n.d.) was used for real-time communication between devices. By using WebRTC, real-time communication between devices can be realised with an internet browser, and other applications do not need to be pre-installed. Moreover, any type of client device can be used if the device can use an internet browser. The environment used in this verification is shown in Table A1. The horizontal angle of view of the client device was not disclosed in its specifications. Therefore, we measured the distance between the object and the camera necessary to capture a constant range, and obtained an angle of 71°.

| Table A1 Development environment |
|----------------------------------|
| **Server PC,** lab-made          |
| CPU                              |
| Intel Core i9-9900KF              |
| GPU                              |
| NVIDIA GeForce GTX 1080 Ti (11 GB)|
| RAM                              |
| DDR4-3200 16 GB × 2               |
| OS                               |
| Ubuntu 18.04.5 LTS               |
| **Client device,** Apple iPad Pro 11-inch, 1st generation |
| System on a chip                |
| Apple A12X Bionic                |
| Camera                           |
| 12MP width, f/1.8 aperture       |
| OS                               |
| iPad OS 14.7.1                   |
| Horizontal field of view         |
| 71° (measured value)             |

B. The detail of \( \Delta E \) and the determine process of threshold values

\( \Delta E^* \) is an index for evaluating the difference of two colours. There are three methods to calculate \( \Delta E^* \) for the CIELAB colour space, of which CIEDE2000 (\( \Delta E_{2000}^* \)) (Sharma et al., 2005) is defined as a standard in ISO/CIE 11664-6 (International Commission on Illumination, 2014).

Threshold values were determined based on the allowable errors specified in Japan Industrial Standard (JIS) S6006 (Japan Industrial Standard Committee, 2020) and JIS S6028 (Japan Industrial Standard Committee, 2007). JIS S6006 is specified to correspond to ISO 9180 (International Organization for Standardization, 1988), but ISO 9180 does not have an entry for colour errors, so JIS S6006 and JIS S6028 were used. JIS S6006 and JIS S6028 are standards for coloured pencils and watercolours. Although the colours within the image were originally
acquired in the RGB colour space, after we confirmed that they were in the range of colours that can be expressed in the CMYK colour space, we decided to apply the allowable errors specified in JIS S6006 and JIS S6028.

C. How to calculate IoU
The formula for calculating the IoU is shown in Equation (C.1).

$$IoU = \frac{TP}{TP + FP + FN}$$ (C.1)

Here, $TP$ (true positive) indicates the number of pixels that could be detected correctly, $FP$ (false positive) indicates the number of pixels that were detected incorrectly, and $FN$ (false negative) indicates the number of pixels that could not be detected.

D. How to calculate the execution speed
The formula for calculating the processing time is shown in Equation (D.1), and the formula for calculating the average processing time is shown in Equation (D.2). In the formula, $t_{ef}$, $t_{sf}$ and $T_f$ indicate the time when the frame is $f$ and $F$ indicates the total number of frames.

$$T_f = t_{ef} - t_{sf}$$ (D.1)

$$T_{ave} = \frac{1}{F} \sum_{f=1}^{F} T_f = \frac{1}{F} \sum_{f=1}^{F} (t_{ef} - t_{sf})$$ (D.2)