Interactive System for Real-time Detection and GIS Registration of Damaged Houses using Mobile Devices to Streamline Damage Assessment

Yusei Kishiawa[a], Norihiko Kawai[a], Motoyuki Suzuki[a], Tadashi Ise[b], and Takeshi Isono[b]

[a]Osaka Institute of Technology, 1-79-1 Kitayama, Hirakata-shi, Osaka, Japan 573-0196
[b]National Research Institute for Earth Science and Disaster Resilience 3-1 Tennodai, Tsukuba, Ibaraki, Japan 305-0006

ABSTRACT

Streamlining damage surveys during disasters is essential for the prompt issuance of disaster certificates, thereby facilitating accurate assessment of damage conditions and the early implementation of support measures. This paper proposes an interactive mobile system for efficient damage assessment that estimates the absolute positions and damage levels of houses by combining camera-based house detection, SLAM-based device pose estimation, and device sensor data, and then automatically registers the results in the GIS. In experiments, we evaluated the effectiveness of the proposed system using both miniature house models and real buildings.

Keywords: SLAM, Object detection, Localization, Device Sensor, YOLO, Voice Recognition, GIS

1. INTRODUCTION

In Japan, when a disaster occurs, damage assessments are required to issue disaster certificates that determine the level of support for victims. Although mobile devices have been partially introduced, damage surveys are still largely conducted manually, with surveyors visiting individual houses, visually inspecting damage, and recording inspection results and locations.[1] In severely damaged areas, collapsed or deformed buildings make it difficult to identify original structures. Although GNSS provides surveyor positions, spatial offsets between surveyors and damaged houses often prevent accurate alignment with pre-disaster GIS data, especially when observations are conducted from distant or elevated locations, resulting in labor-intensive surveys with location inconsistencies.

To address this issue, we propose an interactive mobile system that automates the acquisition of house information during disaster surveys. The system detects houses from camera images, estimates camera poses using Simultaneous Localization and Mapping (SLAM), and computes absolute house positions by integrating relative measurements with device sensor information. The estimated damage levels and locations are directly registered into a Geographic Information System (GIS), while an interactive interface with touch-based and voice-based operations allows surveyors to efficiently correct and register information on site.

2. RELATED WORK

This study differs from conventional damage assessment methods in terms of practical accuracy, survey efficiency, and 3D localization of individual houses. This section briefly reviews representative approaches and clarifies the position of the proposed system.

Satellite-based damage assessment is a widely used approach. Representative methods analyze pre- and post-disaster satellite images using annotated datasets such as xBD[2] or combine optical and SAR (Synthetic Aperture Radar) images to improve robustness against weather conditions.[3] While these approaches enable large-scale analysis, their limited spatial resolution and dependence on aerial viewpoints make it difficult to evaluate detailed shape changes of individual buildings. Another line of research estimates house damage from aerial or drone

Further author information:
Yusei Kishikawa: E-mail: m1m24a12@st.oit.ac.jp
Norihiko Kawai: E-mail: norihiko.kawai@oit.ac.jp

images using deep learning and GIS-based building information.[4] Although the method allows rapid visualization of damage on GIS maps, it relies on the availability of aerial imagery and assumptions inherent to top-down observation, and misclassification can occur due to roof shape and material variations.

To improve survey efficiency, a remote assessment system has been proposed in which surveyors upload images and location data of damaged houses for off-site evaluation.[5] While the system reduces the need for on-site inspections, the accuracy of house location registration depends on surveyor positions, and the assessment workload increases significantly when large numbers of images must be reviewed.

In contrast, this study detects houses directly from camera images captured by mobile devices and estimates both damage status and 3D positions from ground-level viewpoints. Interactive touch-based and voice-based operations enable surveyors to correct recognition results on site. Furthermore, since house locations are estimated using SLAM-based camera pose estimation and device sensor information rather than aerial imagery, conventional assumptions used in satellite- or drone-based methods cannot be applied.

Regarding specific-object localization, YOLOStereo3D[6] integrates object detection with stereo-based 3D inference for applications such as autonomous driving. Unlike this approach, our work focuses on estimating absolute geographical positions of buildings from monocular mobile imagery and registering damage information directly into a GIS, addressing a fundamentally different problem setting.

3. PROPOSED METHODOLOGY

In the proposed system, Visual SLAM[7] estimates the relative pose of the mobile device. YOLOv8[8] then detects multiple houses in the camera images and classifies them into two classes: "Safe" and "Destroyed." The detected houses are tracked across frames, and their relative 3D positions are computed with respect to the camera's initial pose as the origin. Subsequently, the absolute 3D positions of the target houses are obtained through coordinate transformation using GPS, electronic compass, and gravity sensor data acquired from the device. During this process, the system also provides an interactive interface that enables surveyors to view and modify house attributes through either voice input or touch-based operations. Finally, the estimated house positions and damage classifications are registered in the GIS. The following sections describe each component in detail.

3.1 House Detection Using YOLO

This system employs YOLOv8, which supports real-time inference, for house detection. YOLOv8 outputs bounding-box (bbox) information along with class labels for each detected object. However, although pre-trained YOLOv8 models are publicly available, none of them are specifically trained for house detection. Therefore, this study trains a custom model through transfer learning using pre-trained YOLOv8 weights by annotating images containing houses into two classes: “Safe” and “Destroyed”. To further accelerate inference on mobile devices, we convert the trained model into TensorFlow Lite format, enabling lightweight and efficient on-device processing.

3.2 Object ID and Class Label Assignment and Tracking

Object tracking is performed by detecting houses in consecutive frames and associating detected houses across frames. Specifically, the bbox of each detected house is obtained for each frame. If the distance between the center coordinates of the bbox in the previous frame and those in the current frame falls within a predefined threshold, the same ID and class label as in the previous frame are assigned to those in the current frame. If the distance exceeds the threshold, a new ID and class label are assigned. It should be noted that the same house may occasionally be detected with different class labels for different frames. In this system, the class label assigned in the initial frame in which the house first appears is retained. However, the class label can later be modified by the user through the interactive interface described in a subsequent section.

3.3 Calculation of Relative 3D Positions

Using the 2D coordinates of houses tracked across frames and the device's camera pose estimated by Visual SLAM, the relative 3D position of each detected house with respect to the initial camera pose is calculated. Here, as shown in Figs. 1 and 2, in practice, due to deviations in the extracted house positions within the images and errors in the estimated camera pose, the intersection point of multiple rays in 3D space generally does not converge to a single point. Therefore, we estimate the 3D position of the target house by minimizing the sum of squared distances between the position to be estimated and points on each ray.

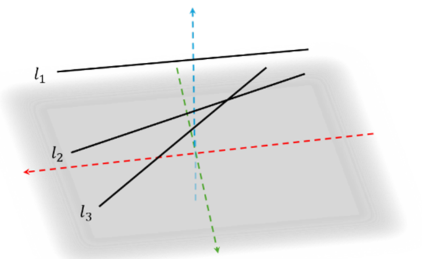

Figure 1. Multiple twisted straight lines in 3D space.

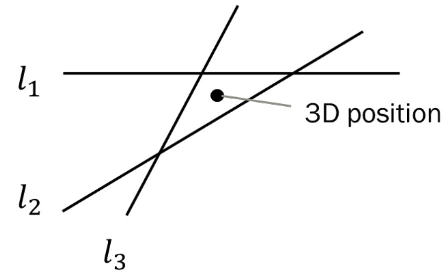

Figure 2. Multiple Lines as Seen in Fig.1.

3.4 Calculation of Absolute 3D Position (latitude and longitude)

By integrating the calculated relative 3D position with device information, the object's absolute 3D position (latitude and longitude) is calculated. Specifically, the device acquires GPS (latitude and longitude), electronic compass (azimuth), and gravity sensor (tilt) data. Using these data, 3D positions in the coordinate system within the application are converted to those in a plane rectangular coordinate system[9]

3.5 Display and Modification of Attributes via Touch-based Operation and Voice Input

In this system, since the recognized house class may not always be accurate, users can make on-the-spot corrections based on visual judgment via touch-based operations and voice input. For touch-based operations, when a user taps a detected house on the screen, the attribute information for the selected house and buttons for modification appear on the screen, allowing the user to modify the class label or ID as needed. For voice input, the user's spoken audio is sent from the device to the server. The server then performs voice recognition using Whisper Large model[10] and voice analysis using regular expressions. In paticular, it extracts words from the voice-recognition text to display attributes and modify IDs and class labels.

3.6 Automation of GIS Registration

The final calculated status and 3D position information for each house are registered in the GIS.[11] The e-Community Map (e-Map)[12] is used for the GIS, and automatic input of object information significantly streamlines the conventional house information input process. The process for registering to the e-Map involves temporarily holding the building information received from the client and sequentially registering stored building data to the e-Map.

3.7 System Interface

Figure 3 shows the system interface. (1) The top left of the screen displays the GPS coordinates at system startup and the azimuth angle for each frame as device information. (2) The three buttons on the left side of the screen, from top to bottom, execute temporary data saving, voice processing, and registration to the GIS. (3) The bottom left shows the camera's position and orientation estimated by Visual SLAM. (4) The mini-map in the center of the screen uses Google Maps, allowing the user to intuitively confirm the positional relationship between the camera and objects by placing virtual objects on the map. (5) The right side of the screen displays the result of analyzing the camera image with YOLOv8 as a RawImage.

4. EXPERIMENTS AND CONSIDERATIONS

In this experiment, we verified the display and modification of object attributes detected on a mobile device using YOLOv8, and confirmed whether the object's state and position were correctly registered on the e-Map. The experiment utilized two models—a miniature house and a full-size house—both independently generated based on manually-annotated training images. This system employed the ROG Phone 8 as the mobile device and was developed using Unity 2021.3.33f1 with ARCore.

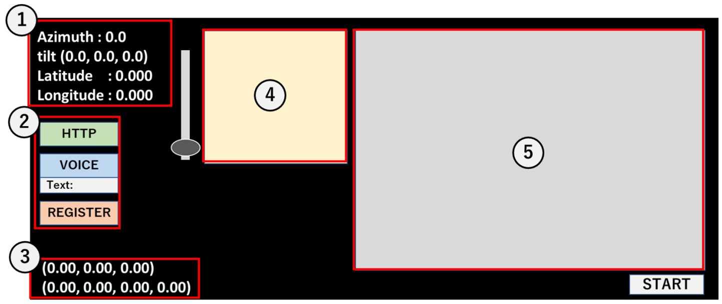

Figure 3. System interface.

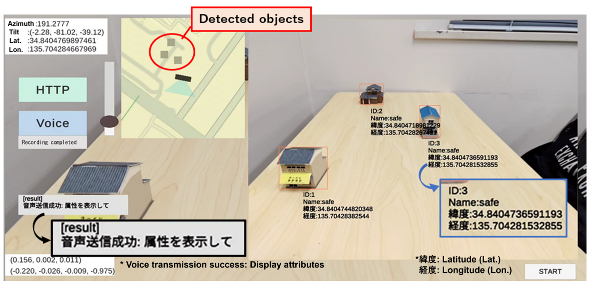

Figure 4. Attribute display.

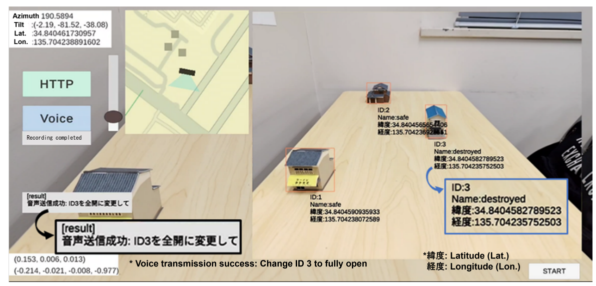

Figure 5. Attribute modifiers.

4.1 Experiment 1: Attribute Modification via Voice Processing

In Experiment 1, we verified whether the attributes of detected miniature houses were correctly displayed and corrected through voice processing. In this experiment, a miniature house model was generated using 410 training images of miniature houses. Figures 4 and 5 show the results of attribute display and correction via voice processing for the miniature house scene.

First, we pressed the Voice button and said "Display attributes" to show the attributes. As shown in Fig. 4, the voice recognition result "Display attributes" appears below the Voice button, confirming that the attributes of the detected miniature house are displayed correctly. Next, assuming that the state of obj3 was "destroyed," we performed the operation to change obj3's state from "safe" to "destroyed" by saying "Change ID 3 to destroyed" in Japanese. Figure 5 shows the voice analysis result displaying "Change ID 3 to fully open" in Japanese, but this occurred because the Japanese utterance "zenkai" is pronounced the same for both 全壊 (destroyed) and 全開 (fully open). Although the system initially recognized and displayed "fully open," corresponding to a different kanji, the regular-expression-based processing correctly interpreted the intended meaning as "destroyed."

4.2 Experiment 2: Verification of Calculated Positions Using Miniature Houses

Experiment 2 verified whether the calculated 3D positions and states of miniature houses were correctly registered on the e-Map using the same miniature house model as in Experiment 1. Figures 6 and 7 show the ground truth of the miniature house scene and the detection results. We confirmed that the class labels matched the ground truth. Table 1 shows that the information during the processing, which was temporarily saved on the server, for subsequent registration to a GIS.

Figure 8 shows the result of registering the information of each detected miniature house. This figure indicates that the distances from the initial position to the detected objects are close to the ground truth, and the objects are correctly classified by the miniature house model. This suggests that the calculated relative positions of the miniature houses are generally accurate, and the relative locations of objects can be discerned on the e-Map. However, despite the experiment being conducted indoors, some detected objects were registered at locations outside the building in which the experiment took place. This is thought to be caused by a shift in the map acquisition position due to GPS error at the time the camera was started.

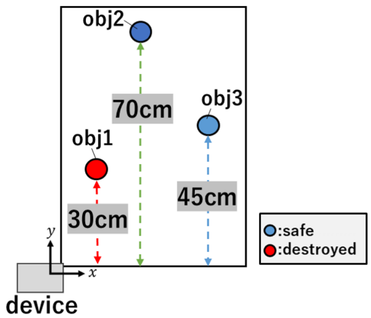

Figure 6. Ground truth of miniature house scene 2.

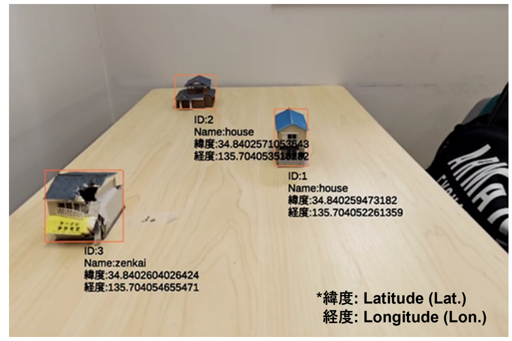

Figure 7. Detection results.

Table 1. Information on each miniature house.

timestamp	ID	status	lat	lon
2025/11/11 14:32	1	house	34.84025946	135.7040523
2025/11/11 14:32	2	house	34.84025709	135.7040536
2025/11/11 14:32	3	zenkai	34.8402604	135.7040547

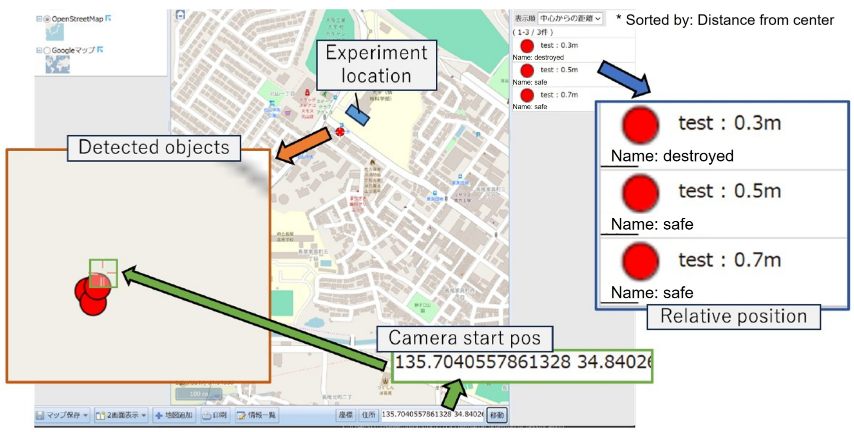

Figure 8. Miniature house registration results.

4.3 Experiment 3: Verification in an Actual House Environment

In Experiment 3, we verified whether the detected house information correctly reflected the actual houses in a real environment on the e-Map. In this experiment, a house model was generated using 225 house images as training data, and YOLOv8's data augmentation functions were used for training.

Figure 9 shows detection results in the real environment. The detection and tracking of houses were not sufficiently stable. Consequently, as seen in Fig. 10, the house positions could not be correctly registered on the e-Map. Several factors likely contributed to this issue, including GPS acquisition errors, insufficient house detection accuracy, and inadequate parallax, all of which prevent the correct estimation of 3D positions.

5. CONCLUSION

This study proposed a system that automatically detects houses using mobile devices and registers their estimated positions obtained through SLAM, multi-view geometry, and device sensors onto e-Map. This aims to streamline damage surveys required for issuing disaster certificates during disasters. In experiments using miniature houses, map registration based on calculated 3D positions yielded values close to the ground truth. However, absolute positional errors occurred due to GPS inaccuracies acquired during camera startup. In experiments conducted in real house environments, although the YOLOv8 model trained with data augmentation was used to improve

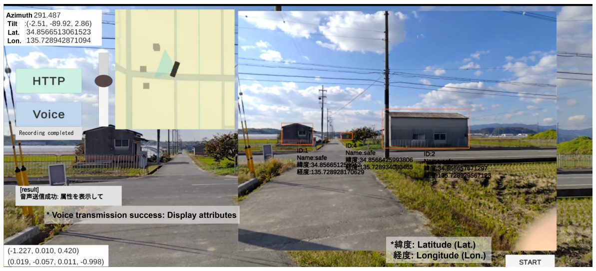

Figure 9. Detection Results Using House Models.

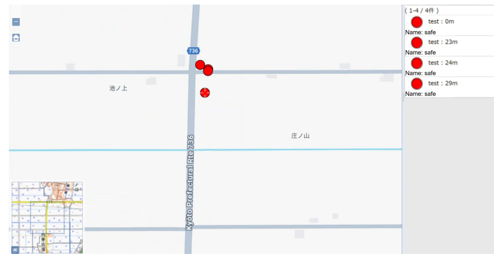

Figure 10. House Registration Results.

training robustness for generating house models, house detection and tracking proved unstable, preventing correct registration of house positions. Factors contributing to this include not only GPS estimation errors but also insufficient house detection accuracy and inadequate parallax. Therefore, while the proposed method is effective in limited indoor environments, further improvements in detection accuracy and position estimation are necessary for application in real-world environments.

ACKNOWLEDGMENTS

This research was supported by JSPS KAKENHI Grant Number JP22H01749.

REFERENCES

[1] Ise, T., Kazuki, K., Takeshi, I., and Yuitiro, U., "Basic research on disaster situation grasping tools using mr device (in japanese)," Tech. Rep. 478, Disaster Prevention Science and Technology Research Report (2022).

[2] Gupta, R., Hosfelt, R., Sajeev, S., Patel, N., Bryce Goodman, J. D., Heim, E., Choset, H., and Gaston, M., "Creating xbd: A dataset for assessing building damage from satellite imagery," in [*Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*], 10–17 (2019).

[3] Chen, H., Song, J., Dietrich, O., Broni-Bediako, C., Xuan, W., Junjue Wang, X. S., Wei, Y., Xia, J., Lan, C., Schindler, K., and Yokoya, N., "Bright: a globally distributed multimodal building damage assessment dataset with very-high-resolution for all-weather disaster response," *Earth System Science Data* **17**, 6217–6253 (2025).

[4] Fujita, S. and Hatayama, M., "Estimation method of roof-injured buildings from aerial photo images using deep learning in earthquake disaster," *Doboku Gakkai Ronbunshu D3 (Civil Engineering Planning)* **75**(6), 127–136 (2020).

[5] Makoto, F., Miho, O., and Kimiro, M., "Development of remote system for supporting building damage assessment during large-scale earthquake disaster (in japanese)," *Journal of Japan Association for Earthquake Engineering* **12**(7), 19–37 (2012).

[6] Liu, Y., Wang, L., and Liu, M., "Yolostereo3d: A step back to 2d for efficient stereo 3d detection," in [*Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*], 13018–13024 (2021).

[7] Developers, G., "Slam using arcore." https://developers.google.com/ar/develop (2023). Accessed: 2023-10-22.

[8] Ultralytics, "Ultralytics yolov8." https://github.com/ultralytics/ultralytics/blob/main/docs/en/models/yolov8.md (2025). Accessed: 2025-08-07.

[9] Geospatial Information Authority of Japan, "Japan plane rectangular coordinate system (notification no. 9 of the ministry of land, infrastructure, and transport, 2002)." https://www.gsi.go.jp/LAW/heimencho.html (2002).

[10] Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., and Sutskever, I., "Robust speech recognition via large-scale weak supervision," *OpenAI Technical Report* (2022).

[11] Geospatial Information Authority of Japan, "What is gis." https://www.gsi.go.jp/GIS/whatisgis.html (2025). Accessed: 2025-11-13.

[12] National Research Institute for Earth Science and Disaster Resilience, "e-community platform (e-comi) — regional information sharing system." https://ecom-plat.jp/index.php?gid=10457 (2025). Accessed: 2025-10-20.