Automated Defect Inspection in Building Construction with Multi-Sensor Fusion and Deep Learning

Juhyeon Kim[a], Jeehoon Kim[a], Yulin Lian[a], and Hyoungkwan Kim[a]

[a]Department of Civil and Environmental Engineering, Yonsei University, Korea
E-mail: kah5125@yonsei.ac.kr, john101010@yonsei.ac.kr, yulin.lian@yonsei.ac.kr, and hyoungkwan@yonsei.ac.kr

Abstract –

The occurrence of defects during the building construction process significantly impacts housing quality. One such defect, the distortion of a building's framework, affects both sustainability and aesthetics. This study presents an automated technique for inspecting framework distortion in building construction by measuring the angles between walls. The proposed method employs a portable data acquisition system that allows for dynamic data collection. The system's accuracy is enhanced through calibration based on terrestrial laser scanning (TLS) data as a reference. Point cloud data are registered to form a map of the interior space, leveraging a deep learning algorithm to visualize framework distortions. When tested in an apartment construction environment, the method reduces data acquisition time compared to the TLS-based approach, while maintaining precision with an average angular error of 0.28 degrees. This study demonstrates a cost-effective and accurate solution for defect inspection in the construction industry.

Keywords –

Defect Inspection; Mobile Data Acquisition System; Multi-Sensor Calibration; Point Cloud Registration; Defect Visualization

1 Introduction

Defects occurring during the building construction process profoundly affect housing quality. Critical defects encompass structural flaws, substandard finishing work, inadequate electrical and plumbing systems, insufficient moisture and waterproofing measures, and faulty mechanical systems [1]. Addressing these defects is vital for ensuring the building's safety and longevity. Among these, distortion of the building's framework is a critical structural defect. It directly impacts the building's long-term durability and its aesthetic integrity.

Recent studies have leveraged 2D and 3D computer vision technologies along with deep learning algorithms to detect building defects more effectively. Wang et al. [2] developed a photogrammetry-based pipeline for 3D reconstruction of buildings, enabling automated identification of building surface defects such as moulds and cracks on reconstructed 3D scenes. Guo et al. [3] proposed a rule-based deep learning approach for detecting façade defects, including cracks, delamination, peeling, and spalling. Chow et al. [4] presented an automated system for detecting cracks and spalling in buildings using mobile data collection, deep learning, and scene reconstruction. Tan et al. [5] developed a method for integrating crack data from unmanned aerial vehicle images into building information models, thereby improving the inspection of high-rise building façades. In addition, several studies have been conducted to monitor various defects in buildings, such as leakage and heat loss [6, 7, 8]. Despite the importance of monitoring framework distortion, research in this area remains scarce. Moreover, there is a lack of technologies that can inspect framework distortion accurately and efficiently for on-site application. Terrestrial Laser Scanning (TLS) is one of the accurate methods for quality assurance and control in construction [9], and is applicable for inspecting framework distortion. However, while this method is highly precise, it comes with considerable constraints in terms of cost and time.

To address the above issue, this paper proposes a novel approach for accurately and cost-effectively measuring framework distortion, as shown in Figure 1. We calibrate multiple Azure Kinect DK depth cameras using TLS data as a reference, and then acquire point cloud data. For visualization purposes, the point cloud data are registered to form a map. The angles between walls are analyzed from each point cloud data, and these values are visualized on the map. A deep learning-based registration algorithm is used for the calibration of the sensors and the formation of the map.

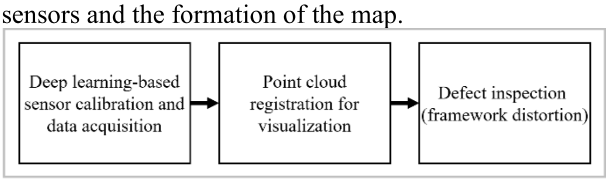
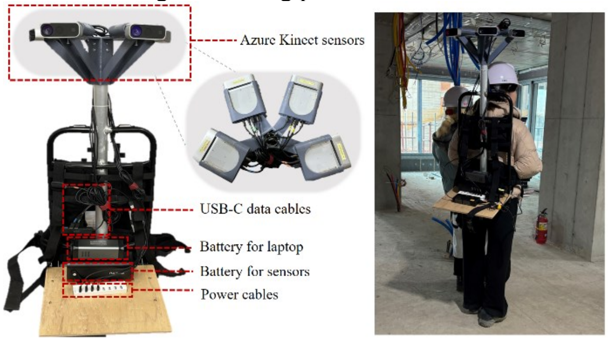


Figure 1. Overview of the proposed method.

2 Methodology

2.1 Deep Learning-Based Sensor Calibration and Data Acquisition

A backpack platform was developed for data acquisition. As shown in Figure 2(a), the platform was equipped with multiple Azure Kinect sensors and batteries for powering the sensors and a laptop. Four Azure Kinect sensors were mounted to broaden the field of view for data acquisition. As shown in Figure 2(b), an operator can carry the platform to collect data while moving around. The point cloud data were saved in the MKV file format, and later PCD files were extracted from the MKV file. During data acquisition, we ensured that the starting and ending positions were identical.



(a) (b)

Figure 2. A backpack platform for data acquisition; (a) hardware configuration, and (b) operational demonstration of the backpack platform in field.

For the extrinsic calibration of multiple Azure Kinect sensors, TLS data were used as the ground truth. As shown in Figure 3, point cloud data from each of the four different Azure Kinect sensors were registered with the TLS data. We utilized deep global registration (DGR) [10], a deep learning-based registration algorithm, for aligning Azure Kinect data with TLS data. Through registration, four transformation matrices were obtained, which reveal the relative positional relationships among the Azure Kinect data.

The TLS data in Figure 3 were used solely for calibration purposes; the data were obtained in a laboratory setting. In this study, additional TLS data were acquired to serve as ground truth for calculating the performance of the proposed defect inspection method (Figure 6(b)). The TLS data are different from those in Figure 3 and were acquired from an actual apartment construction site.

Once the calibration of multiple Azure Kinect sensors was completed using TLS data, there was no need to repeat the calibration process. The relative transformations among the sensor coordinate systems were calculated through calibration, and these calculated values remain valid as long as the relative positions of the sensors do not change.

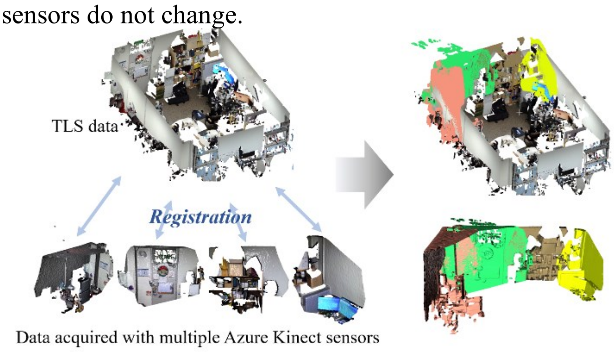


Figure 3. Calibration of multiple Azure Kinect sensors using TLS data and DGR.

2.2 Point Cloud Registration for Visualization

The point cloud data, obtained from the calibrated multiple Azure Kinect sensors, were employed to inspect the distortion of the building's framework (detailed in Section 2.3). A point cloud map was formed through registration, which was later used to visualize the results of the angle measurement between walls. Point cloud data extracted from the MKV file were sequentially registered using DGR. As shown in Figure 4, once all the point cloud data were registered, a map of the entire interior space could be generated.

In the initial map, as shown in Figure 4(a), misalignment occurred between the point cloud data sets; this was due to the failure to recognize that the data acquisition starting and ending positions were identical. Therefore, we used DGR to register two point cloud data obtained at the start and end points of data acquisition, thereby calculating the degree of discrepancy (Figure 4(c)). The amount of discrepancy was propagated across all point cloud data between the two data. Ultimately, a complete map without mismatch between the point cloud data was generated, as shown in Figure 4(b). This map was used solely for visualizing defect information and was not utilized in defect analysis.

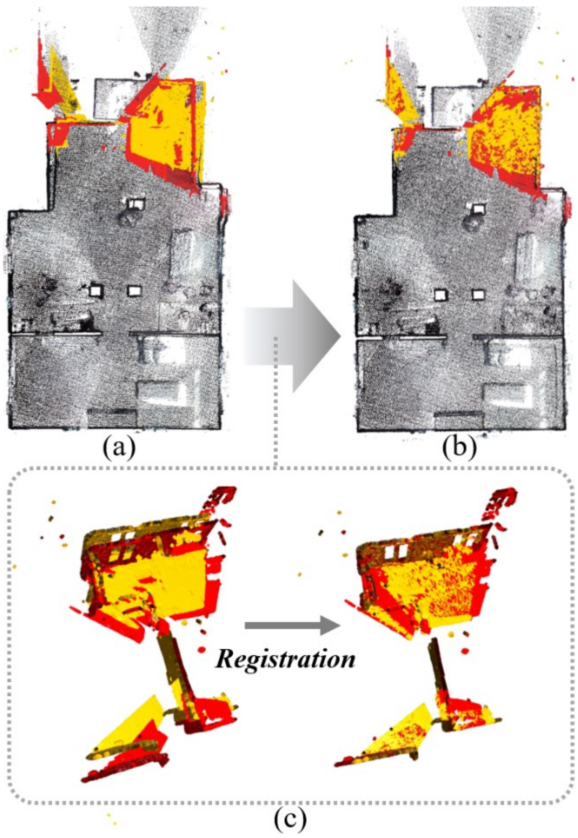


Figure 4. Point cloud registration and misalignment resolution for visualization using DGR; (a) initial map exhibiting misalignment, (b) refined map with aligned point clouds, and (c) calculation of the discrepancy between start and end point cloud data.

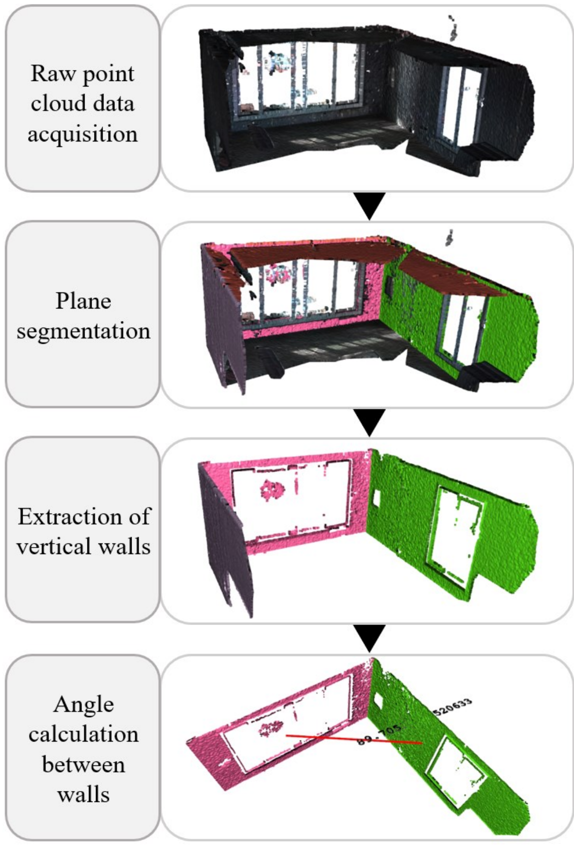


Figure 5. Workflow of point cloud data processing for defect inspection in apartment construction.

2.3 Defect Inspection

Each Azure Kinect data underwent a defect inspection process. Figure 5 shows the procedure for calculating angles between walls. Using random sample consensus (RANSAC), planes were segmented from the raw point cloud data. Subsequently, only the vertical walls were extracted from these planes, and the angles between them were calculated. The angles were visualized on the map generated in Section 2.2. During this process, angles located in close proximity on the map were merged and represented by their average value.

3 Experiments and Results

3.1 Datasets

Point cloud data for one unit of the apartment were acquired at an apartment construction site. The effectiveness of the proposed method was validated through a performance comparison with the TLS-based method. Figure 6(a) and Figure 6(b) show data acquisition using the backpack platform developed in this study and data acquisition using TLS, respectively. As shown in Table 1, a total of five scans were conducted using TLS, taking about 25 minutes excluding the time to move the sensor. With the proposed method, data were continuously acquired for about 3 minutes and saved in the MKV file format. From the acquired MKV file, 191 PCD files were extracted.

(a) (b)

Figure 5. Data acquisition for comparative experiment; data acquisition with (a) backpack platform, and (b) TLS.

Table 1. Comparison of data acquisition methods using TLS and multiple Azure Kinect sensors: analysis of scan frequency, time, and cost.

Method	Number of scans	Total acquisition time (mm:ss)	Sensor prices (USD, in thousands)
TLS	5	24:40 (excluding sensor relocation time)	38.3
Four Azure Kinect sensors	1	2:34	2.5

3.2 Implementation Details

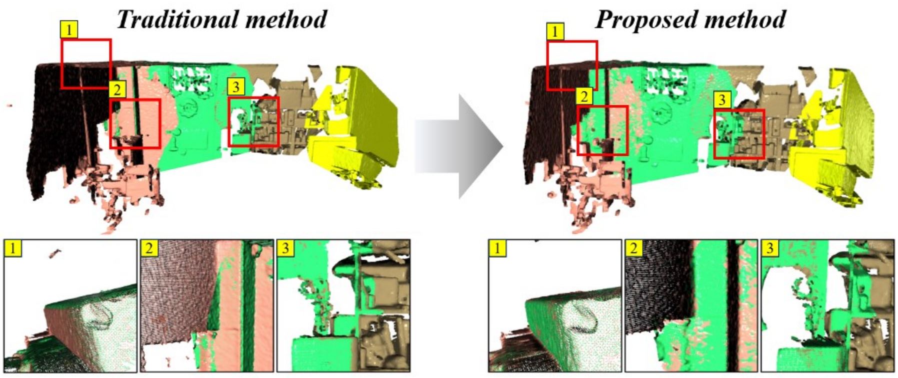
When acquiring Azure Kinect data, all sensors were synchronized and connected in a daisy-chain configuration; this configuration refers to the sequential interconnection of pairs of sensors [11]. For registration, we utilized the pre-trained DGR algorithm, which had been trained on the 3DMatch dataset [12]. For calculating the angles between walls, four planes were extracted from each point cloud data using RANSAC. When executing the RANSAC algorithm, the maximum distance for a point to be classified as an inlier was set to 2 cm and the number of points randomly sampled for plane estimation was set to 3. Data acquisition was conducted on a laptop equipped with an Intel Core i7-10750H CPU and an RTX 2060 GPU. Data processing was performed on the Ubuntu 16.04 operating system with an Intel Xeon Gold 6240M CPU processor and an RTX 3080 GPU based on the Python programming language.

3.3 Experimental Results

3.3.1 Sensor Calibration

We conducted comparative experiments to validate the proposed calibration method for multiple Azure Kinect sensors. Figure 7(a) shows the results of a typical calibration method using an AprilTag marker and the iterative closest point (ICP) algorithm. Figure 7(b) shows the results of calibration using the proposed method with TLS data and DGR. As shown in the figure, when the sensors were calibrated in a typical way, a misalignment occurred between the Azure Kinect data. The suggested calibration technique addressed this issue, thereby enhancing the data quality.

When performing calibration using the proposed method, it was possible to combine point cloud data with precision comparable to TLS data, and it eliminated the need for labor-intensive processes like AprilTag marker detection. Figure 8 shows examples of point cloud data from multiple Azure Kinect sensors calibrated using TLS data and DGR.



(a) (b)

Figure 6. Comparative analysis of Azure Kinect sensor calibration methods; (a) calibration with an AprilTag marker and ICP, and (b) calibration using TLS data and DGR.

Figure 7. Examples of point clouds from multiple Azure Kinect sensors calibrated using TLS data and DGR.

3.3.2 Point Cloud Registration for Visualization

All Azure Kinect data were combined using DGR to form a map, which was then utilized for visualizing the results of defect inspection. Figure 9 shows the registered point cloud map generated after adjusting the discrepancy between the point cloud data sets. The figure demonstrates that the created map was precise enough to visually comprehend the building's internal elements adequately.

3.3.3 Defect Inspection

The angles between walls measured from each Azure Kinect data were visualized on the registered point cloud map (Figure 10). As shown in Table 2, angles between a total of 10 pairs of wall surfaces were measured. To evaluate the accuracy of the proposed method, the same defect inspection process was applied to TLS data. Figure 11(a) and Figure 11(b) show examples of wall angle measurement using the TLS-based approach and the proposed method, respectively. The proposed method demonstrated an average angular measurement error of 0.28 degrees when compared against the TLS-based approach.

Figure 8. Results of point cloud registration for visualization using DGR.

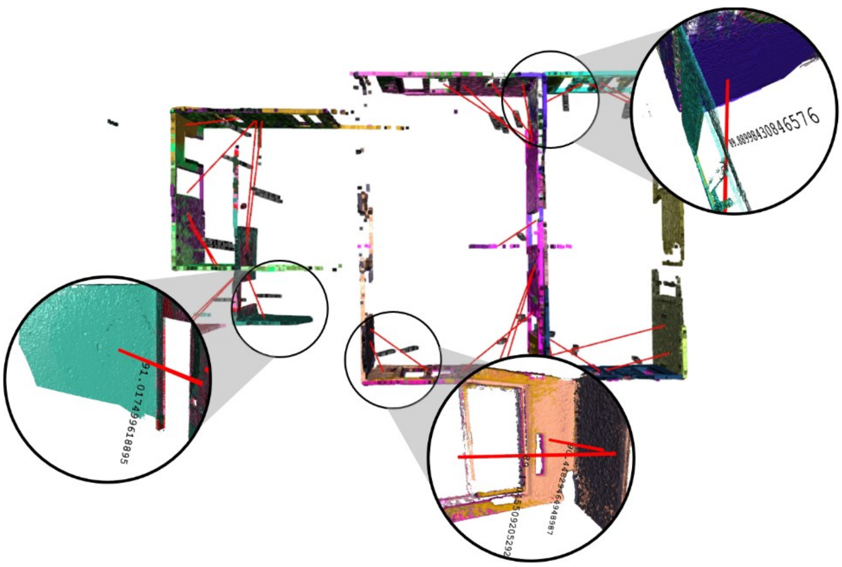
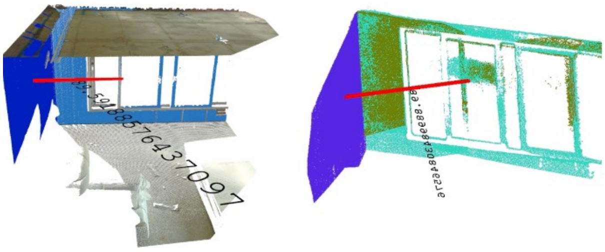


Figure 9. Visualization of calculated angles between walls in a registered point cloud map.



(a) (b)

Figure 10. Examples of wall angle measurements from point cloud data; angle measurement using (a) TLS data, and (b) Azure Kinect data.

Table 2. Comparative analysis of wall angle measurements from TLS and Azure Kinect data.

Wall pair	TLS angle (degrees)	Azure Kinect angle (degrees)	Error (degrees)
1	90.92	90.34	0.58
2	89.89	89.59	0.30
3	89.78	89.90	0.12
4	90.44	89.98	0.46
5	90.01	89.75	0.27
6	89.65	89.76	0.12
7	89.81	90.05	0.24
8	90.05	89.73	0.33
9	91.02	91.01	0.01
10	89.64	90.00	0.36

3.4 Discussion

The experimental results demonstrate the potential of the developed backpack platform and defect inspection technique. However, there is still room for improvement in the proposed method through future studies.

First, there is a need for preprocessing point cloud data in the defect inspection process. Noise may occur when acquiring data from a distance using Azure Kinect sensors. Removing such noise based on the data acquisition distance can enhance defect detection performance. Statistical outlier removal and radius outlier removal are common methods used to remove noise from point cloud data. However, these methods do not effectively remove scattered noise points that are acquired from distant ranges in Azure Kinect data. Therefore, using density-based clustering for noise removal can be an effective solution.

Second, the data acquisition platform can be improved to increase its on-site applicability. To achieve this, several strategies can be employed: using a Mini PC instead of a laptop, utilizing efficient batteries to reduce weight, implementing real-time data processing for defect inspection, redesigning the backpack to reduce worker fatigue, and providing visual guides to for data acquisition.

Third, in addition to analyzing angles between walls, a wider array of defects should be addressed in future studies. Sagging in ceilings and floors is also critical defect information, and such defects can be detected using the proposed method. In addition to structural defects, surface anomalies such as cracks, voids, and spalling on the structure's surface can also be detected. These surface defects can be inspected using not only point cloud data but also by applying vision-based methods that utilize images.

Fourth, the developed technology should be validated in more field applications to increase its robustness. The proposed method was validated on a single type of apartment construction site. There is a need to apply the method to a wider variety of building construction sites and address the various challenges that arise in the process. Such diverse real-world implementations will aid in enhancing the applicability of the proposed method.

Addressing these four key improvement areas could significantly enhance the efficacy of the proposed method, making it a highly valuable tool in the field of defect inspection and building analysis.

4 Conclusion

This study proposed a novel pipeline for inspecting framework distortion in building construction employing multiple Azure Kinect sensors. By calibrating the sensors against TLS data and implementing a deep learning algorithm for registration, the system created a comprehensive 3D map of the building's interior. The angles between walls analyzed from the Azure Kinect data were visualized on the 3D map. This approach not only significantly reduced the time required for data acquisition but also maintained a high level of accuracy. If the proposed method is further developed, it could revolutionize the way building construction projects are managed by ensuring effective defect inspection and enhanced safety.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Ministry of Education (No. 2018R1A6A1A08025348) and the "National R&D Project for Smart Construction Technology (No.23SMIP-A156488-04)" funded by the Korea Agency for Infrastructure Technology Advancement under the Ministry of Land, Infrastructure and Transport, and managed by the Korea Expressway Corporation. This research was conducted with the support of i-thetto Co., Ltd., and was funded by DL E&C Co., Ltd.

References

[1] Ministry of Land, Infrastructure and Transport. Investigation of Defects in Apartment Buildings, Estimation of Repair Costs, and Standards for Determining Defects. On-line: http://www.molit.go.kr/USR/policyData/m_34681/dtl.jsp?id=4503, Accessed: 14/12/2023.

[2] Wang, T., Gan, V. J., Automated joint 3D reconstruction and visual inspection for buildings using computer vision and transfer learning. *Automation in Construction*, 149: 104810, 2023.

[3] Guo, J., Wang, Q., Li, Y., Evaluation-oriented façade defects detection using rule-based deep learning method. *Automation in Construction*, 131: 103910, 2021.

[4] Chow, J. K., Liu, K. F., Tan, P. S., Su, Z., Wu, J., Li, Z., Wang, Y. H., Automated defect inspection of concrete structures. *Automation in Construction*, 132: 103959, 2021.

[5] Tan, Y., Li, G., Cai, R., Ma, J., Wang, M., Mapping and modelling defect data from UAV captured images to BIM for building external wall inspection. *Automation in Construction*, 139: 104284, 2022.

[6] Lee, K., Lee, S., Kim, H. Y., Bounding-box object augmentation with random transformations for automated defect detection in residential building façades. *Automation in Construction*, 135: 104138, 2022.

[7] Chen, J., Lu, W., Lou, J., Automatic concrete defect detection and reconstruction by aligning aerial images onto semantic-rich building information model. *Computer-Aided Civil and Infrastructure Engineering*, 38(8): 1079-1098, 2023.

[8] Puliti, M., Montaggioli, G., Sabato, A., Automated subsurface defects' detection using point cloud reconstruction from infrared images. *Automation in Construction*, 129: 103829, 2021.

[9] Tang, X., Wang, M., Wang, Q., Guo, J., Zhang, J., Benefits of terrestrial laser scanning for construction QA/QC: a time and cost analysis. *Journal of Management in Engineering*, 38(2): 05022001, 2022.

[10] Choy, C., Dong, W., Koltun, V. Deep global registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2514-2523, Seattle, Washington, United States, 2020.

[11] Microsoft. Synchronize multiple Azure Kinect DK devices. On-line: https://learn.microsoft.com/en-us/azure/kinect-dk/multi-camera-sync, Accessed: 14/12/2023.

[12] Zeng, A., Song, S., Nießner, M., Fisher, M., Xiao, J., Funkhouser, T. 3DMatch: Learning local geometric descriptors from RGB-D reconstructions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1802-1811, Honolulu, Hawaii, 2017.