Automatic Detection of Air Bubbles with Deep Learning

Takuma Nakabayashi[a], Koji Wada[b] and Yoshikazu Utsumi[b]

[a]Technical Research Institute, Obayashi Corporation, Japan
[b]Construction Robotics Division, Obayashi Corporation, Japan
E-mail: nakabayashi.takuma@obayashi.co.jp, wada.koji@obayashi.co.jp, utsumi.yoshikazu@obayashi.co.jp

Abstract –

Recently, the demand for base-isolated structures has been increasing, especially in Japan. For reliable construction of seismic isolation devices, contractors must ensure that the area ratio of the air bubbles occupying the backside of the base plate does not cross the threshold, which is decided by the structural designer. However, the backside of the base plate can include larger and more air bubbles than ordinary concrete surfaces as it is difficult to pour concrete into the foundation properly. Additionally, the inspection process includes many time-consuming tasks. Therefore, it normally takes about one week or longer after concrete work to perform the inspection.

We present a method to automate the tasks relating to inspection, including image preprocessing, air bubble extraction, and calculation of the area ratio of the air bubbles, using conventional image processing and a convolutional neural network (CNN) to speed up the inspection.

While CNN normally requires a significant amount of training data to achieve high performance, it is generally difficult to collect such a large amount of high-quality data. We have conducted thorough accuracy inspections to evaluate the appropriate amount of training data required. Additionally, we have verified the effect of data augmentation and compared the performance of certain typical CNN architectures.

As a result, our method has obtained results that are close to those of manual inspection by a skilled inspector. We can conclude that our method can reduce the overall inspection time by 50% compared to conventional methods.

Keywords –

Base-Isolated Structure; Concrete; Visual Inspection; Air Bubble; Bug Hole; Machine Learning; Deep Learning; Image Processing

1 Introduction

The demand for seismic base isolation has increased due to the number of earthquakes experienced across the world, especially in Japan. Seismic isolation devices are installed at specific levels of the building in a base-isolated structure to reduce the tremble caused by earthquakes. As shown in Figure 1, seismic isolation devices are installed on the base plate, which is set on the base-isolated foundation. To integrate the base plate into the foundation correctly, it is necessary to pour concrete with the base plate in place, as shown in Figure 2. This pouring process could lead to the formation of air bubbles underneath the base plate, which should be avoided. As these bubbles negatively affect adhesion, their area ratio should be decreased to increase the adhesion force. However, the removal of the base plate is not allowed during the construction phase to check for the presence of air bubbles. Therefore, a full-scale construction field test is required before the base-isolated foundation work.

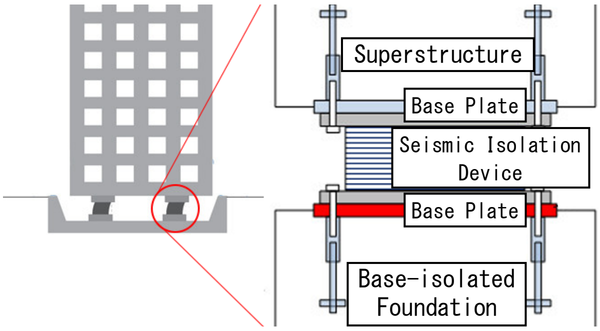

Figure 1. Cross-section of base-isolated foundation

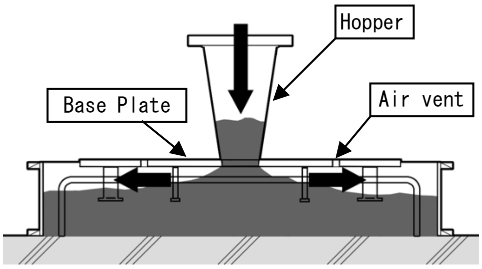

Figure 2. Methodology to cast concrete underneath a baseplate

A field test is conducted to inspect whether the ratio of the area of the air bubbles to that of the concrete surface satisfies the criteria that are generally set by structural designers. According to [1], the criteria for the ratio of the area of air bubbles is defined as less than 10%, which differs for each project. We need to locate air bubbles and measure each bubble's maximum diameter and area to calculate the area ratio. As there can sometimes be over 1000 air bubbles, the entire inspection process is time consuming and work intensive.

The purpose of the field test is to confirm whether the planned workflow of the base-isolated foundation is appropriate. Base-isolated foundation work can begin when the result of the field test is verified. However, if the field test does not pass the criteria, the concrete placement plan must be reconsidered. Naturally, the field test must also be repeated.

This study introduces a method to automate the entire inspection process in the field test using conventional image processing and a convolutional neural network (CNN). Additionally, we have conducted many accuracy inspections to evaluate the appropriate amount of training data and to determine which architecture is suitable for the considered purpose. The remainder of this paper is organized as follows: related works are reviewed in Section 2, the conventional inspection method is presented in Section 3, the proposed method is introduced in Section 4, the experiments and results at the actual site are described in Section 5, and the conclusions are presented in Section 6.

2 Related Work

Many technologies have been developed to improve productivity in concrete work [2]. Their scope is not restricted to construction as they can be applied to inspection. Visual inspection is one of the important examinations in concrete work. Prasanna et al. [3] developed an automated crack detection algorithm for robot inspection.

As in the present study, previous research has used CNN and other machine learning approaches to tackle this issue. Cha et al. [4] developed a crack damage detection method based on a CNN architecture. They mentioned that CNN is more robust than conventional approaches, such as the Canny and Sobel edge detection methods. Gang et al. [5] tested certain typical CNN architectures for tunnel crack detection. As listed here, many studies have focused on automatic crack detection systems.

However, certain studies have also focused on automatic air bubble detection on the concrete surface. Yoshitake et al. [6] developed a method for tunnel lining. Gang et al. [7] developed a CNN-based method and compared it to Otsu's thresholding method and Laplacian of Gaussian (LoG) algorithm. They concluded that their method was more robust against various light conditions. Fujia et al. [8] tested a popular CNN architecture for semantic segmentation, which is a major field in computer vision. The goal of semantic segmentation is to classify the class labels at the pixel level.

In contrast to prior studies, this study specifically focuses on the concrete surface of the backside of the base plate in base-isolated structures, rather than the concrete surfaces on more classical architectural elements, such as walls and columns. The backside of the base plate normally includes more air bubbles than the surfaces of other architectural elements, as it is difficult to vent the air from the bottom of the base plate properly. Additionally, it is not possible to check how the concrete is being filled during the construction owing to the base plate structure. Therefore, the bottom of the base plate can include many air bubbles, of sizes larger than those on ordinary concrete surfaces. In a report focusing on the detection of air bubbles at the bottom of the base plate, Mitani [9] and Katoh [10] evaluated the accuracy of their method, which connected 2 CNN architectures that were used for object detection and semantic segmentation. However, the main targets of most object detection methods based on CNN are the objects, which are along the XY coordinates. This is due to the manner in which object detection methods are trained. In the training process, they normally evaluate if they properly detect the target objects by using bounding boxes that are along the XY coordinates. In contrast, air bubbles are multi-oriented objects. Therefore, they can be impediments in the training process as their bounding boxes can cover a wide area of the concrete surface and may intersect with other air bubbles.

In contrast, as semantic segmentation methods generally do not depend on the axes on which target objects are based. Thus, we only used a CNN architecture for semantic segmentation to deal with various types of air bubbles. Additionally, this paper presents a method to automate the entire inspection and air bubble detection processes. Here, we employ conventional image preprocessing approaches, followed by semantic segmentation with a CNN structure, to detect air bubbles accurately. Each of these processes will be explained in Section 3.

3 Conventional Inspection Method

In the field test, according to the construction plan, concrete work is conducted at the beginning, after which the base plate is removed. Marking work is then completed as a preparation for the area ratio calculation (Figure 3). If the area ratio satisfies the criteria, the construction site can proceed to the actual construction phase. The area ratio calculation in conventional inspection is composed of the following steps.

1. Color the air bubbles red by hand using markers. (Figure 4)
2. Take a picture of each grid.
3. Preprocess pictures, through keystone correction, cropping, and resizing of each picture using an image processing software.
4. Color the areas that are not on the backside of the base plate, as this is outside the scope areas of the inspection. (Figure 4)
5. Automatically extract the colored air bubbles identified in step 1 using the software. If necessary, correct the extracted results manually.
6. Exclude the air bubbles that are less than the specified diameter. (Figure 4)
7. Repeat steps 3 to 6 for each grid.
8. Automatically calculate the area ratio of the air bubbles against the area of the concrete surface.

It is important to note that certain tasks need to be carried out carefully. To perform accurate automatic extraction in step 5, it is necessary to color the air bubbles accurately in Step 1, and then take a picture from the front of the concrete surface as much as possible in Step 2. Although keystone correction in Step 3 is necessary irrespective of how carefully the pictures are taken, the lower the distortion, the higher the obtained image quality. The areas identified as out of scope in Step 5 indicate locations that can be ignored. These not only include the outside of the backside of the base plate but also on bolts, air vent holes, and pressure inlets. As shown in Figure 4, these areas are colored using different colors so that they are ignored during the color extraction process in Step 8. Additionally, a negligible minimum diameter is normally set for this inspection. The air bubbles that are smaller than the minimum diameter need to be colored in Step 6 (colored black in Figure 4). After all the grids are processed completely from Steps 1 to 6, the area ratio can be automatically calculated by using the color extraction method.

The conventional inspection method involves a lot of manual work. Additionally, the number of pictures to be processed can reach up to 100 per specimen. Further, if the construction site has multiple design types of base-isolated foundations, the field test must be conducted for each design. Thus, the tests need to be performed by a skilled inspector to hasten the entire process. However, despite having a skilled inspector for the field test, the process takes about a week after the completion of concrete work to obtain the test results when using the conventional inspection method.

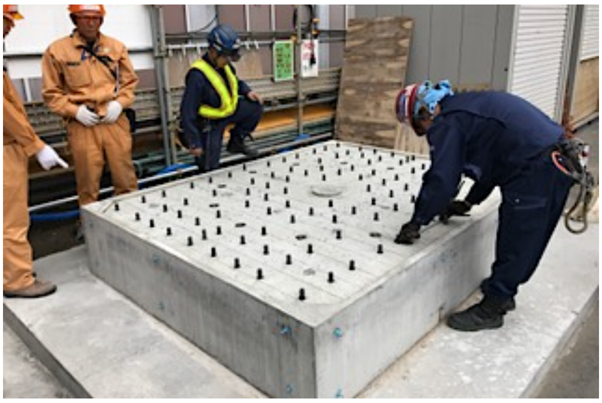

Figure 3. Concrete specimen and marking work

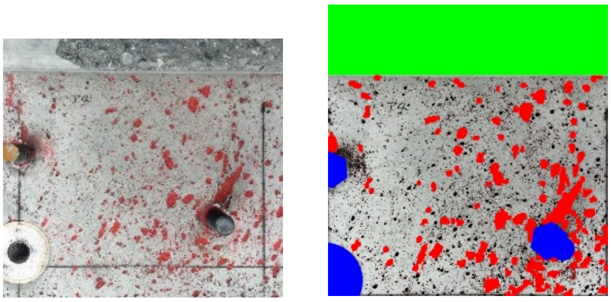

Figure 4. Before preprocessing (Left), result of preprocessing (Right)

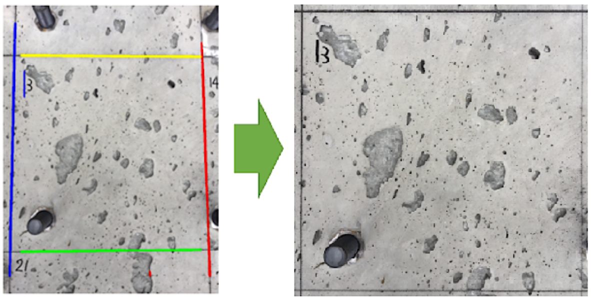

Figure 5. Example of processed marking recognition, keystone correction, and cropping

4 Methodology

4.1 Automation of preprocessing using a conventional image processing method

The preprocessing described in Step 3 in the previous section includes keystone correction, cropping, and resizing. This is necessary to calculate the area ratio in the real scale from a photograph. The preprocessing is performed manually using an image processing

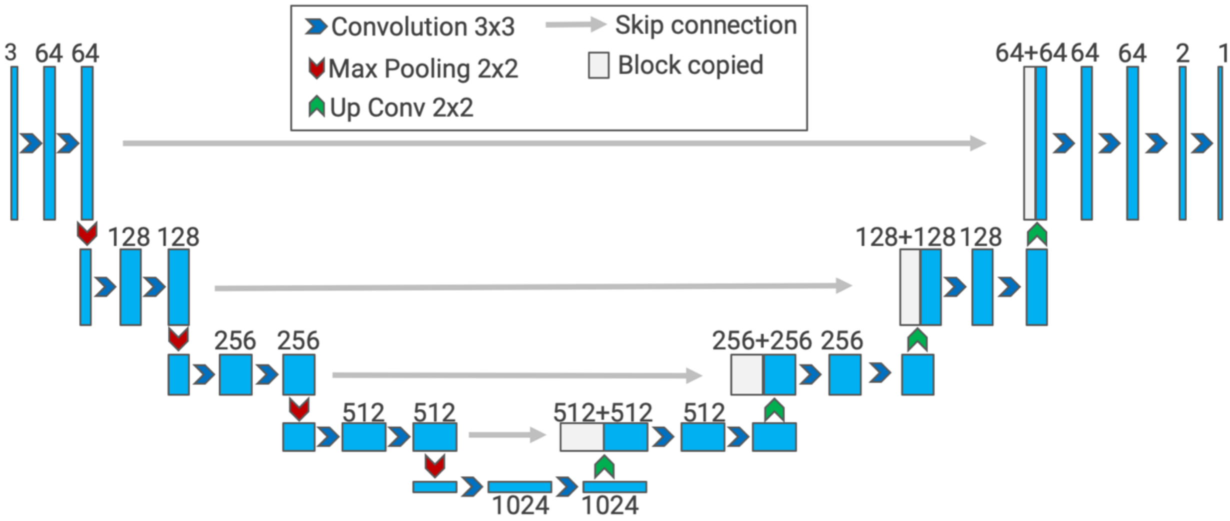

Figure 6. Network architecture based on U-Net. Each number describes the number of channels in each convolutional layer.

software. This process takes about one day, even for a skilled inspector.

To automate the keystone correction process, it is first necessary to recognize the marked grids automatically. In this study, we use a template-matching algorithm to achieve this. The template-matching algorithm can detect the areas in the image that have a pattern similar to the pattern of the input template. We prepared simple vertical and horizontal templates to detect the grids. As a result, the algorithm can detect the grid lines as four different colored lines, as shown on the left side in Figure 5.

Keystone correction utilizes the results of template matching. It needs to calculate the conversion matrix from a perspective image to an orthogonal image. It uses the four corner points detected by template matching for this. As a result, we obtain a square image that does not contain the perspective distortions, as shown on the right side in Figure 5.

The image is then resized to the specified resolution. This process clarifies the relationship between the pixels and the real scale. The resolution used in this study was set to 2000 px.

4.2 Automation of air bubble detection using a CNN

In this study, a U-Net-based architecture is utilized for automatic bubble detection. The U-Net was designed for semantic segmentation, which won the International Symposium on Biomedical Imaging (ISBC) cell tracking challenge in 2015 [11]. Semantic segmentation is a typical task in the machine learning field and is performed to classify images in pixel units.

As shown in Figure 6, the U-Net is mainly composed of two symmetry paths for encoding and decoding, where each path is composed of multiple layers of CNN blocks. Each layer has a connection from the encoding path to the decoding path. The connections contribute to maintaining a wide context in each focus area. The U-Net was originally designed for biomedical image segmentation, but its simple architecture can be utilized for many purposes, including air bubble detection. Therefore, we adopt the U-Net as the base architecture of our method.

CNN is a type of neural network that normally consists of many convolutional layers. Each layer has many filters, and each filter can be trained to extract various types of image features. In contrast to CNN and recent learning-based algorithms, conventional image processing algorithms are not robust in diverse environments. Several studies referenced in Section 2 report that CNN is more robust to changes in the light environment than conventional image processing algorithms.

Although CNN is more accurate and robust, its

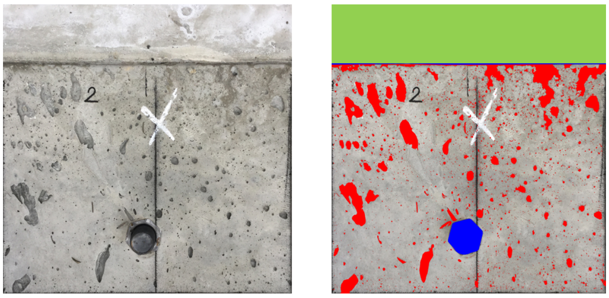

Figure 7. Sample pair of training data

training process needs a large amount of training data. Additionally, U-Net belongs to a supervised training algorithm, indicating that the training dataset needs to have not only images but also labels that describe the classes to which each pixel in the image belongs. Therefore, we collected a large amount of data, as shown in Figure 7.

While a learning-based algorithm is used for air bubble detection, the postprocessing, including calculation of each air bubble's maximum diameter and area, uses the thresholding and contour extraction method. As mentioned in Section 3, small air bubbles that have maximum diameters less than the specified length are ignored in the inspection. In Section 5.2, the negligible diameter is set to 5 mm.

5 Experiments

5.1 Basic Experiments

We evaluated different combinations of datasets to identify a suitable amount of training data to obtain a high-performance trained model. We collected 452 images from 4 construction sites, with varying characteristics and light environments. Additionally, four images from each construction site were used as test data. Therefore, in total, 436 training data points and 16 test data points were considered. Each image was divided into smaller images, as processing high-resolution images requires a large GPU memory. Thus, the resolution was set to 512 px.

Precision, recall, intersection over union (IoU), and F1 scores were used as metrics to evaluate each result. Equations (1), (2), (3), and (4) define these metrics. Figure 8 describes each value of the equations, such as true positive (TP), true negative (TN), false positive (FP), and false negative (FN). In this study, we assign the greatest importance to the F1 score, as it can evaluate not only precision but also recall.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$IoU = \frac{TP}{TP + FP + FN} \tag{3}$$

$$F1\ Score = \frac{2Precision * Recall}{(Precision + Recall)} \tag{4}$$

		Actual	
		Positive	Negative
Predicted	Positive	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

Figure 8. Confusion matrix used for the evaluation

Table 1. Training Dataset

Dataset	Site 1	Site 2	Site 3	Site 4	Total
S1	10	10	10	10	40
S2	30	30	30	30	120
S3	60	60	59	60	239
S4	223	94	59	60	436

Table 2. Results of Each Dataset (N = 16)

Dataset	Precision	IoU	Recall	F1 Score
S1	0.803	0.687	0.835	0.812
S2	0.874	**0.769**	0.869	**0.868**
S3	**0.888**	**0.769**	0.856	**0.868**
S4	0.839	0.763	**0.897**	0.863
Otsu	0.686	0.563	0.786	0.712

We prepared four datasets, as shown in Table 1, to evaluate the effect of the amount of training data. Table 2 presents the results obtained using the datasets. According to the results, the result of S2 was almost the same as that of S3 and slightly better than that of S4. The result indicates that 30 images from a construction site can contain most of the patterns in the construction site. Additionally, we evaluated Otsu's automatic image thresholding [12] as a baseline result of conventional image processing. The result is shown at the bottom of Table 2. The result is lower in all metrics compared to the results of our method, even if the dataset is as small as S1. This means that air bubbles are not always darker than concrete surfaces and have various characteristics.

To improve the results, we additionally conducted a study using data augmentation (DA). DA is known as the general technique to improve the results of machine learning algorithms. It includes a wide variety of manipulations, but we used only simple image processing methods, such as rotation, mirroring, expansion, contraction, color channel shifting, brightness, and random crop. The results are shown in Table 3. It increased the F1 score by 0.024. The result shows that DA can work well in our domain as well as in the other domains.

Table 3. Results of Data Augmentation (N = 16)

Dataset	Precision	IoU	Recall	F1 Score
S4	0.839	0.763	**0.897**	0.863
S4+DA	**0.885**	**0.800**	0.895	**0.887**

5.2 Additional Experiments

In addition to the experiments presented in the previous section, we conducted several experiments to identify the generalization of the trained model and prepared additional datasets, as shown in Table 4.

S4 was the same as the dataset used in the previous section. S5 included the data from S4 in addition to the data from Site 5, 6, 7, and 8, and S6 included data from S5 in addition to data from Site 9.

Table 4. Training dataset for generalization test

Dataset	Site 1–4	Site 5–8	Site 9	Total
S4	436	0	0	436
S5	436	494	0	960
S6	436	494	32	992

In this test, four images from Site 9 were used as the test data. Site 9 was not included in S4 and S5 and was included only in S6. To clarify the results, we used a simpler DA approach, such as slight rotation, expansion, contraction, mirroring, and random crop, in this test.

As shown in the result in Table 5, the higher the number of training data points, the more accurate the result. However, while S5 had almost the same number of training data points as S6, the result was not at the same level. This means that obtaining good results requires more diverse data or data from the construction site where our method was planned to be use.

Table 5. Results against test data from Site 9 (N=4)

Dataset	Precision	IoU	Recall	F1 Score
S4	0.726	0.615	0.792	0.756
S5	0.724	0.684	**0.927**	0.809
S6	**0.913**	**0.823**	0.894	**0.902**

Additionally, to evaluate the capabilities of different architectures, PSPNet [13] and DeepLabv3+ [14] were compared to the U-Net-based architecture. Both PSPNet and DeepLabv3+ are known for their well-designed architectures that enable them to exploit a broad image context. In this study, ResNet-101 [15] was used as the backbone architecture for PSPNet and DeepLabv3+.

We used the S6 dataset to train all the architectures during the test. Additionally, four images from 8 construction sites were used as test data, totaling 32 data points for the test. We also used a simple DA, similar to that used in the previous test.

As shown in Table 6, our U-Net-based architecture obtained the highest F1 score. This was almost the same as the results for S2, S3, and S4, as described in Table 2.

Table 6. Results of Architecture Comparison Including Negligible Air Bubbles (N = 32)

	Including negligible air bubbles			
Architecture	Precision	IoU	Recall	F1 Score
U-Net based	0.873	**0.768**	**0.869**	**0.865**
PSPNet	0.885	0.643	0.698	0.774
DeepLabv3+	**0.916**	0.704	0.754	0.821

Table 7. Results of Architecture Comparison Ignoring Negligible Air Bubbles (N = 32)

	Ignore negligible air bubbles			
Architecture	Precision	IoU	Recall	F1 Score
U-Net based	0.864	0.687	0.770	0.803
PSPNet	0.860	0.710	**0.805**	0.819
DeepLabv3+	**0.909**	**0.715**	0.770	**0.821**

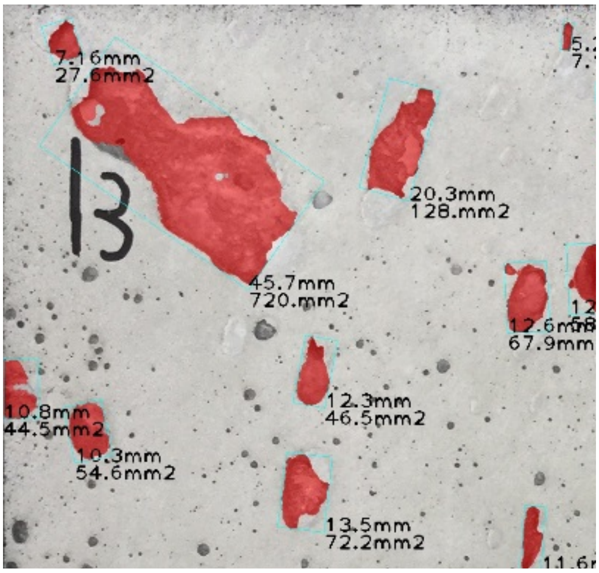

Figure 9. Result of the proposed method

However, while our U-Net-based architecture can detect almost all the smaller air bubbles, we found that it sometimes did not work well against larger air bubbles. Therefore, we reevaluated the results while ignoring the negligible air bubbles. As mentioned earlier, the negligible length was set to 5 mm. Table 7 shows the results, and Figure 9 shows the predicted result.

It can be seen that when the negligible length is ignored, the results change significantly. The recall of PSPNet was increased by 0.097, indicating that it was good at detecting larger air bubbles but not smaller ones. However, the recall of our U-Net-based architecture declined by 0.139. This reveals how well this architecture worked against small air bubbles. Additionally, Table 7 shows that our U-Net-based architecture maintained a comparative F1 score in this study.

From these results, we found that the dataset used in this paper needs not only the capability of detecting details, but also the capability of wide perception. For future work, continuous exploration of CNN architectures will be required to achieve higher performance.

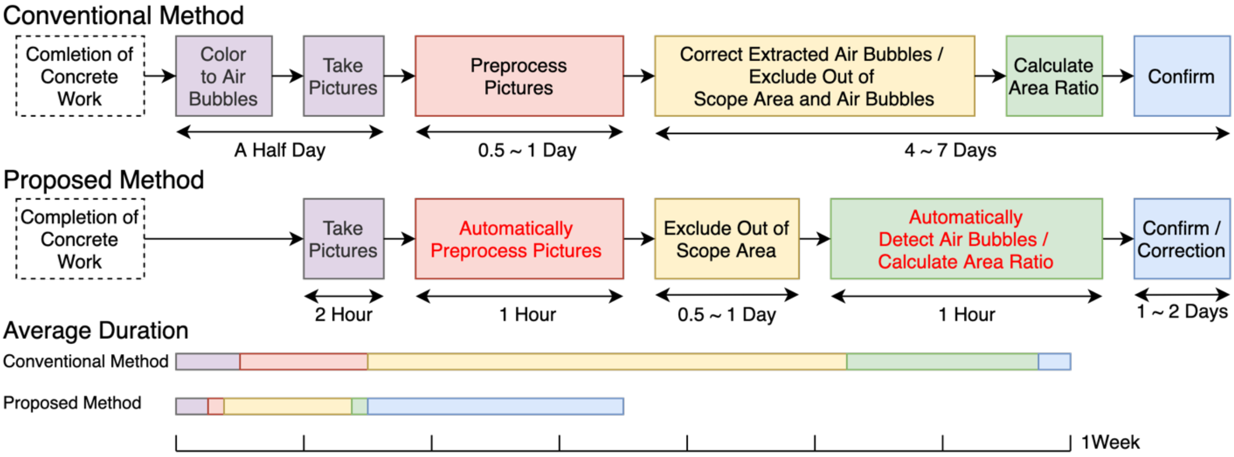

Figure 10. Workload Comparison between Conventional Method and Proposed Method, assuming that 100 pictures have to be processed.

5.3 Workload Comparison between the Conventional Method and Our Method

In this section, the workload between the conventional method and the proposed method is compared to evaluate the effect of the proposed method. The best-performing model in Section 5.1 was used for air bubble detection.

Figure 10 shows the workflow and workload of the conventional method and the proposed method. As shown here, the conventional method generally took approximately one week. The main tasks in the conventional method included preprocessing images and coloring of the air bubbles by hand. However, these manual steps were avoided in our method. At this moment, the predicted results still contain some false detections. However, as mentioned earlier, the U-Net based architecture works well against smaller air bubbles.

Therefore, in this comparison, the inspector did not have to care about the smaller air bubbles. The area ratio was obtained with a high degree of accuracy by correcting the oversight of large air bubbles. The surface of the base-isolated foundation used in this comparison has 64 grids. Considering these grids, the area ratio obtained manually was 6.19%, and that calculated by our method, with oversight correction, was 6.56%. Additionally, the oversight correction took less than 2 hours since the inspector only focused on correcting the oversight of large air bubbles.

Although the performance of our method needs to be improved, we have confirmed that it is effective in reducing the overall inspection time. Additionally, regarding the automation of preprocessing, we found that it worked almost perfectly for the expected situation where the grid was marked. Although it sometimes does not work well when including traces of the base plate's boundary, we conclude that it contributes to improving the speed of the preprocessing step.

From the above results, we conclude that our method can reduce the overall inspection time by 50%.

6 Conclusion

In this study, we developed a method to automate concrete surface inspection of a base-isolated structure. This paper presented a method that uses conventional image processing algorithms and a learning-based algorithm. Conventional image processing algorithms, such as template matching, thresholding, and contour extraction, were used for preprocessing and postprocessing of the images. For the learning-based algorithm, we used a U-Net-based architecture for automatic air bubble detection.

In the experiments, we evaluated the performance of our method, recording a best F1 score of 0.887. When the salient false detections are corrected by hand, the difference between the area ratio calculated using our method and the area ratio calculated manually by a skilled inspector was 0.37%. Furthermore, the workload between the conventional method and our method was compared. As a result, although the results predicted using our method needed to be corrected by hand, our method can reduce the overall inspection time by 50%.

In conclusion, this paper demonstrates the promising performance of the proposed method. While there is still room for improvement in terms of performance, this paper shows that our approach can automate the entire inspection process. There are many other promising CNN architectures and techniques available to improve the performance. Accordingly, future work should focus on exploring better architectures and correcting more training data.

References

[1] JSSI Menshin Kouzou Sekou Hyojun 2017 [JSSI Construction standard of base isolation structure 2017], The Japan Society of Seismic Isolation, pp.84, 2017.

[2] Gharbia. M., Chang-Richards. Y. A., and Zhong. R., Robotic Technologies in Concrete Building Construction: A Systematic Review, 2019 Proceedings of the 36th ISARC, pp. 10-19, 2019.

[3] Prasanna. P., et al., Automated Crack Detection on Concrete Bridges, in IEEE Transactions on Automation Science and Engineering, vol. 13, no. 2, pp. 591-599, 2016.

[4] Cha. Y.-J., Choi. W. and Büyüköztürk, O., Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks. Computer-Aided Civil and Infrastructure Engineering, 32: 361-378. 2017.

[5] Gang. L., Biao. M., Shuannhai. H., Xueli. R., and Qiangwei. L., Automatic Tunnel Crack Detection Based on U-Net and a Convolutional Neural Network with Alternately Updated Clique, Sensors. 20. 717, 2020.

[6] Yoshitake. I., Maeda. T., and Hieda. M., Image analysis for the detection and quantification of concrete bugholes in a tunnel lining, Case Studies in Construction Materials, Vol. 8, pp. 116-130, 2018.

[7] Gang. Y., Fujia. W., Yang. Y., and Yujia. S., Deep-Learning based bughole detection for concrete surface image," Advances in Civil Engineering, vol. 2019, 2019.

[8] Fujia. W., Gang. Y., Yang. Y., and Yujia. S., Instance-level recognition and quantification for concrete surface bughole based on deep learning. Automation in Construction. vol. 107, 2019

[9] Mitani. K., and Katoh. Y., Study on evaluation for surface air of concrete by DeepLearning. Part1 : Outline and Surface Air Detection, Summaries of technical papers of annual meeting, Architectural Institute of Japan, pp. 1227-1228, 2019.

[10] Katoh. Y., and Mitani. K., Study on evaluation for surface air of concrete by DeepLearning. Part 2 : Estimation of binary image of surface air and filling, Summaries of technical papers of annual meeting, Architectural Institute of Japan, pp. 1229-1230, 2019.

[11] Ronneberger. O., Fischer. P. and Brox. T., U-net: Convolutional networks for biomedical image segmentation, Medical Image Computing and Computer-Assisted Intervention (MICCAI), Springer, LNCS, Vol.9351, pp. 234–241, 2015.

[12] Otsu. N, A Threshold Selection Method from Gray-Level Histograms, IEEE Transactions on Systems, Man, and Cybernetics, Vol.9, 1979

[13] Zhao. H., Shi. J., Qi. X., Wang. X. and Jia. J., Pyramid Scene Parsing Network, 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, pp. 6230-6239, 2017.

[14] Chen. L., Zhu. Y., Papandreou, G., Schroff, F., and Adam, H., Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. ArXiv, https://arxiv.org/pdf/1802.02611.pdf, Accessed: 06/03/2020

[15] He, K., Zhang, X., Ren, S., and Sun, J., Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.770-778, 2016