Automation in Construction 130 (2021) 103850

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

Artificial intelligence quality inspection of steel bars installation by integrating mask R-CNN and stereo vision

Yurii Kardovskyi, Sungwoo Moon *

Department of Civil & Environmental Engineering, Pusan National University, Busan 46241, South Korea

ARTICLE INFO

Keywords:
Steel bar
Quality inspection
Artificial intelligence
Convolutional Neural Network (CNN)
Mask R-CNN
Stereo vision
Object detection
Object mask
Instance segmentation

ABSTRACT

Contractors should conduct strict quality inspection of the steel bars used in concrete structures and need to automate the process of quality inspection. The objective of this study is to develop an Artificial Intelligence Quality Inspection Model (AI-QIM) that can execute quality inspection on steel bars at the construction site. The proposed AI-QIM is built on the Mask Region-based Convolutional Neural Network (Mask R-CNN) technique, which can perform instance segmentation of steel bars. This object detection technique is integrated with a stereo vision camera to generate information on steel bar installation. A contractor can use the proposed AI-QIM to estimate the quantity, spacing, diameter, and length of steel bars during quality inspection. A sample case study indicated that the AI-QIM yielded a maximum relative error of 3% when measuring steel bar spacing and a maximum relative error of 8% when measuring steel bar lengths within a range of 1–2 m from a stereo camera.

1. Introduction

Steel bars are installed in the construction of a reinforced concrete structure and resist tensile forces. The strength of the concrete structure can be significantly affected by the reinforcement being even slightly out of place [1]. Therefore, a contractor should conduct strict quality inspection of the steel bars to achieve the desired level of structural safety. However, improper placement of steel bars can occur and reduce the bearing capacity of the structure [2]. This can lead to surface cracking in the structure, early rebar corrosion, and eventually structural failure.

Therefore, the contractor should ensure that all steel bars are installed in compliance with the corresponding shop drawings. Quality inspection should be performed continuously to reduce construction errors during the installation of steel bars. As the Concrete Reinforcing Steel Institute [1] stated, quality inspections should ensure that the quantity, size, location, and spacing of installed steel bars prior to concrete pouring.

However, manual and visual quality inspection is a time-consuming and error-prone process. Moreover, the quality inspector must be able to access the temporary structure to measure and count the installed steel bars. Therefore, the process needs to be automated to improve inspection quality and safety, and reduce the manpower required for steel bar quality inspection. More effort should be exerted to improve the effectiveness of steel bar quality inspection.

Studies have been conducted to help the quality inspector with steel bar installation by using state-of-the-art technologies. For example, Zhang, et al. [3] applied a stationary machine vision system by utilizing a level sensor, proximity switch, and stepper motor mounted on a guide rail for quality inspection of steel bars and achieved diameter detection and vertical spacing measurement. This application comes with multiple sensors and devices, which lowers practicality at the construction site. Moreover, the height for the measurements is fixed, reducing mobility, and it is time-consuming to set up the stationary machine vision system.

Han, et al. [4] used a laser scanning and vision-based 3D reconstruction method to construct point cloud representations for detecting steel bars and their spacing distances. They identified the locations and configurations of steel bars to calculate the spacing between them using a mapping algorithm for object points. Kim, et al. [5] used a terrestrial laser scanner to retrieve 3D point clouds for measuring the dimensions of formwork and steel bars. They developed a preset terrestrial laser scanning technique that can automatically assess reinforcement concrete elements including steel bar spacing and concrete cover in concrete formwork.

The unique aspect of these two approaches is that they use 3D point clouds to create 3D models for the steel bars to be installed for concrete structures. Although the 3D point cloud models afford precise assessment of steel bar conditions, this type of 3D reconstruction comes with a heavy computational burden and is unsuitable for practical applications

* Corresponding author.
E-mail address: sngwmoon@pusan.ac.kr (S. Moon).

https://doi.org/10.1016/j.autcon.2021.103850
Received 31 January 2021; Received in revised form 19 July 2021; Accepted 22 July 2021
Available online 4 August 2021

at construction sites.

Recently, the Convolutional Neural Network (CNN) has been widely used for object detection and segmentation in a variety of applications, such as the detection of workers and excavators at construction sites [6], safety guardrails [7], and concrete cracks [8]. CNNs are also applied to steel bar counting in construction. For example, researchers have applied a CNN-based deep learning artificial intelligence model to detect and count steel bar tips and heads [9–11]. However, all these researchers tried to detect steel bar tips and heads using the object detection and segmentation technique. Their main goal was to count the number of steel bar heads in a bundle before installation.

These previous studies showed that the existing models are effective for structured environments, such as precast concrete factories, but not for unstructured environments, such as construction sites. Steel bar quality inspection is often performed on the go, and the quality inspector should be able to use the information on steel bar installation directly at the construction site. Moreover, the quality inspector should be able to move around to evaluate the quality status. This mobility can help the quality inspector to assess the steel bar status.

The objective of this study is to provide an Artificial Intelligence Quality Inspection Model (AI-QIM) that can assist the quality inspector with quality inspection during steel bar installation. The proposed AI-QIM is built using the Mask Region-based Convolutional Neural Network (Mask R-CNN) technique, which can perform instance segmentation of steel bars. This object detection technique has been integrated with stereo vision to measure the attributes of steel bars. Contractors can use the AI-QIM to detect the quantity, spacing, diameter, and length of steel bars at the construction site for timely quality inspection.

2. Overview of the state-of-the-art technologies

In the application of artificial intelligence quality inspection to steel bar installation, efforts should be exerted to detect steel bars and estimate their attributes, including quantity, diameter, spacing, and length. The AI-QIM described in this study has been developed by integrating the state-of-the-art technologies of CNN and stereo vision to meet the abovementioned requirements. In this approach, CNN performs object detection and stereo vision performs attribute estimation. In this section, these technologies are reviewed in the context of artificial intelligence quality inspection.

2.1. Object detection and segmentation

Convolutional Neural Network (CNN) is another technology that can be applied to steel bar quality inspection because this technique can execute instance segmentation to delineate steel bar shapes. CNN is a deep neural network architecture that provides object recognition and localization on images. In general, a trained CNN model can perform image classification [12], classification and localization [13], object detection and semantic segmentation [14], and instance segmentation [15].

Mask R-CNN [15] is an extended version of Faster R-CNN [16]. The Mask R-CNN model has the additional capability to generate object masks on each region of interest, while Faster R-CNN can output with only bounding boxes. These object masks are a binary representation of the objects detected on input images. The Mask R-CNN technique has been applied to construction worker tracking [17], concrete crack detection and localization [8,18,19], safety distance identification for crane drivers [20], and detection of leakage water in shield tunnel segments [21].

Mask R-CNN is used in this study because the model can distinguish all instances of steel bar objects on an image, along with their respective borders. A specially labeled dataset called Common Objects In Context (COCO) [22] was used for pre-training the Mask R-CNN model because the transfer learning training technique is less time consuming. For example, in this study, a custom dataset with steel bar objects was harvested before training. Then, the Mask R-CNN setup configuration was slightly adjusted and trained on the custom dataset by using the transfer learning technique.

Fig. 1 shows the two stages of the Mask R-CNN model: 1) proposal and 2) segmentation. The proposal stage generates proposals as possible regions in which objects may exist. Feature Pyramid Network (FPN) [23] is an architecture that extracts rich semantic feature maps. Feature maps, also called activation maps, are rich semantic layers that are retrieved after the convolution process. Region Proposal Networks (RPN) [16] is a network that generates object proposals and the corresponding objectness scores. The Region of Interest (ROI) align approach properly positions the features to the input by extracting the feature map of each ROI for the detection and segmentation tasks.

After accepting the proposals generated in the proposal stage, the segmentation stage generates pixel-level object masks and bounding boxes and, subsequently, predicts the object classes of steel bars.

2.2. General description of stereo vision technology

Stereo vision is a technology for stereo image depth calculation. This technology detects an object in 3D space and determines its XYZ coordinates relative to the cameras. Stereo vision technology can be used for distance calculation by calculating the heights, widths, and distances of and to an object [24–27]. In the construction domain, stereo vision has been used to detect the width of cracks in concrete beams [28]; measuring small deformations of bridge girders [29]; and 3D model reconstruction of structural collapses [30].

Stereo vision computes object depth by utilizing the binocular disparity between the object images captured by the left and right cameras (Fig. 2). That is, two parallel cameras located at a known separation distance (i.e., baseline) provide a set of two coplanar images. Full-image point-by-point search and calculation of these matching features are performed to generate depth images [31].

Stereo vision can be implemented using two methods: 1) rectified and 2) unrectified [32]. With the former, the axes of two cameras are parallelly aligned, while with the latter, they are converged onto the single point. In this study, the rectified method was used because it is more applicable in real-time applications at construction sites. The rectified method uses a stereo rig where two cameras are aligned having close to perfectly parallel optical axes.

Stereo vision technology can generate depth map images using two cameras. This technology calculates the dimensions of the objects present on the depth map images. When a CNN is integrated with stereo vision, the former detects the objects of steel bars, and the latter estimates the attributes of the detected objects. Therefore, a CNN can be applied in conjunction with stereo vision to automate the process of steel bar inspection in real time.

2.3. Stereo vision-based distance measurement

Fig. 2 shows a visual representation of the rectified stereo vision where two aligned cameras with parallel optical axes are set up. The figure shows the parallel stereo geometry of the parallel optical axes using dash-dot lines. In this research, an Intel RealSense depth camera is used to implement the stereo vision technique. The focal length (f) and baseline (B) distance of the camera were set to 1.93 mm and 50 mm, respectively. The depth information retrieval resolution and RGB data resolution of the camera were 1280 × 720 px [33].

The Mask R-CNN technique is used in conjunction with the stereo vision technique for object detection and depth perception of steel bars. In the following section, the integration of these two techniques is discussed.

The point coordinates relative to the depth image are used to perform measurements at a given point or between several points on the two-dimensional (2D) plane of a red, green, and blue (RGB) color image.

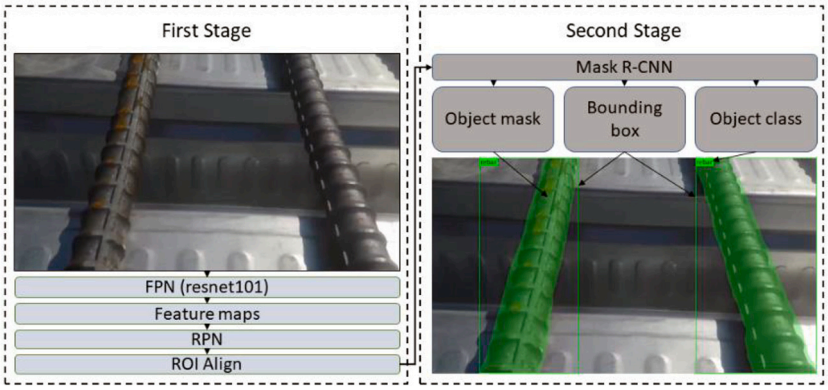

Fig. 1. Structure of a mask R-CNN model.

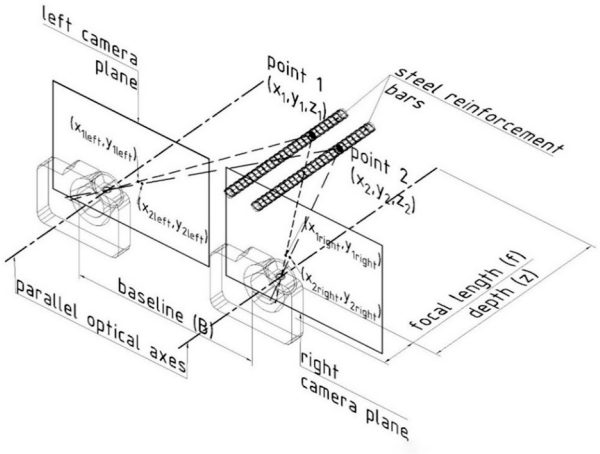

Fig. 2. Parallel stereo geometry with parallel optical axes (dash-dot line).

In this case, the RGB color image is aligned with the depth map image, which is relative to the center of a stereo camera. Then, the object can be detected to generate the information pertaining to heights, widths, and distances. Accurate measurement can be achieved when the input images are aligned precisely one with another.

Eq. (1) shows the relationship between disparity and depth. Here, z denotes the depth to the object from the image plane, f the focal length of the camera in mm, B the baseline between parallel camera lenses axes in mm, and d the disparity of the point in pixels [31].

$$depth\ (z) = \frac{f \bullet B}{d} \tag{1}$$

Eq. (2) is used to calculate the disparity d of a point between two images.

$$disparity\ (d) = x_{1left} - x_{1right} \tag{2}$$

In the figure, point 1 is projected onto both cameras along the short-dashed line, and point 2 is projected onto both cameras with the long-dashed line. In the figure, the distance between two steel bars at points 1 and 2 can be measured using the Pythagorean theorem expressed in Eq. (3).

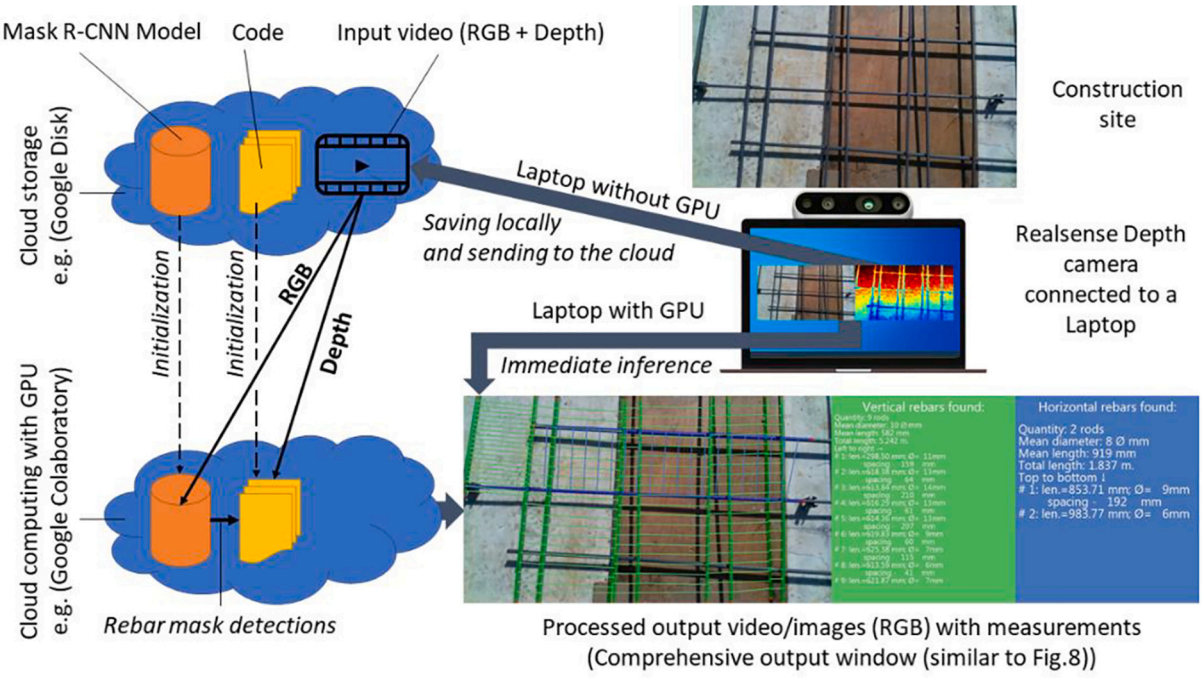

Fig. 3. Conceptual diagram of AI-QIM.

$$distance\ (point\ 1, point\ 2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \quad (3)$$

3. Structure of artificial intelligence-applied quality inspection

Fig. 3 shows the overall structure of the AI-QIM. The AI-QIM consists of two major parts: 1) Mask R-CNN and 2) stereo vision. The AI-QIM has been developed using multiple application programming interfaces (APIs), such as the TensorFlow API, Intel RealSense API, and Python libraries. After the model has been adequately trained, the program source codes can be uploaded to Google Drive cloud storage.

The first Mask R-CNN stage detects and localizes the steel bar objects by using the Mask R-CNN technique. That is, the stereo depth camera connected to a laptop captures both RGB and depth data at the construction site. The cloud computing service initializes the Mask R-CNN model. After initialization, the RGB data is handed to the trained Mask R-CNN model to perform object detection and segmentation of the steel bar objects. The detected and localized steel bar objects are subsequently handed to the stereo vision part.

The second stereo vision part creates a depth map out of the stereo image data. The cloud computing service initializes and fetches all of the stereo-vision-related source codes. Here, the RGB data are aligned with the depth map. Then, the stereo vision source codes measure and calculate object attributes from the aligned pair of RGB images and depth maps.

If the laptop has a built-in GPU with a sufficient VRAM memory and is CUDA-compatible, the AI-QIM can detect the steel bar objects in real-time for automated steel bar inspection at a construction site without the need to perform all of the actions in the cloud.

4. Convolutional neural network training

Convolutional neural network training is a crucial step in the development of the AI-QIM. The AI-QIM uses dataset of steel bar images to train itself. The training work should follow a preset procedure to achieve a high object detection capability. After the training work is completed, the AI-QIM should be validated to measure the level of model performance. When the training is completed, the newly captured data are used as inputs to AI-QIM, and the model, in turn, generates object detection masks. In this context, this section discusses the procedure for training the AI-QIM and the training results.

4.1. Data acquisition

CNN training requires a dataset consisting of many steel bar images. In this study, all 240 images were used in training the AI-QIM. The dataset was acquired using a GoPro camera and a stereo vision camera at two construction sites. To avoid repetition, the steel bar installations at the construction sites were filmed at different heights and later split into different frames apart from the captured images (Fig. 4). Moreover, videos of the steel bars were captured under different weather conditions (i.e., cloudy and sunny) to increase the dataset diversity and avoid false positive detections due to shadows of the steel bars.

The data used for actual distance measurements was recorded using an Intel Realsense Depth Camera. This camera has a special data file format called “rosbag”, which has the “.bag” extension. The .bag data file is used to hold a large amount of uncompressed data per second. The “.bag” file stores two data types, namely, 1) RGB stream and 2) stereo depth stream that can be used for the Mask R-CNN model and depth map construction, respectively, as well as for object mask alignment.

4.2. Data labeling

Once the dataset of steel bar images is obtained, the images in the dataset should be labeled manually. Here, labeling refers to the process of generating polygons that outline specific objects for each instance of steel bar on the image. An online image annotation tool is used to perform object labeling on the dataset images. Although labeling is a very time-consuming and tedious task, it is necessary to train the AI-QIM. The labeling process generates polygon object masks, and all of the data are stored in the JSON file format. This JSON file is later used to generate object masks and XML files for each respective labeled image.

Fig. 5 shows the structure of the labeled data after labeling of the RGB image, namely, the polygon object masks and the point coordinates in the XML file. Fig. 5(a) shows the XML file that stores file path data, image dimensions, and polygon object mask positions. Fig. 5(b) shows the captured image used for labeling. Fig. 5(c) and (d) shows the polygon object masks of the left and right steel bars, respectively.

After all of the steps are completed, the resulting data are saved in the special TFRecord format, which is the standard format of the TensorFlow Object Detection API [34]. This TFRecord file stores all of the RGB images, PNG mask images, data from XML files, image list, and class list. Then, all of the data are combined to generate TFRecord files.

In training and validation, most researchers split the dataset 70–30% [35] or 80–20% [11,36]. In this study, the dataset is split into the ratio of 70–30% to focus more on verifying the accuracy of the AI-QIM.

4.3. Training performance

As aforementioned, all of the 240 images in the dataset were used to train the AI-QIM, and they were stored in two separate TFRecord files: 1) 70% of the images were used for training and 2) 30% were used for

a) Parking garage b) Pharmaceutical College building

Fig. 4. Acquisition of slab steel bar installation data at construction sites

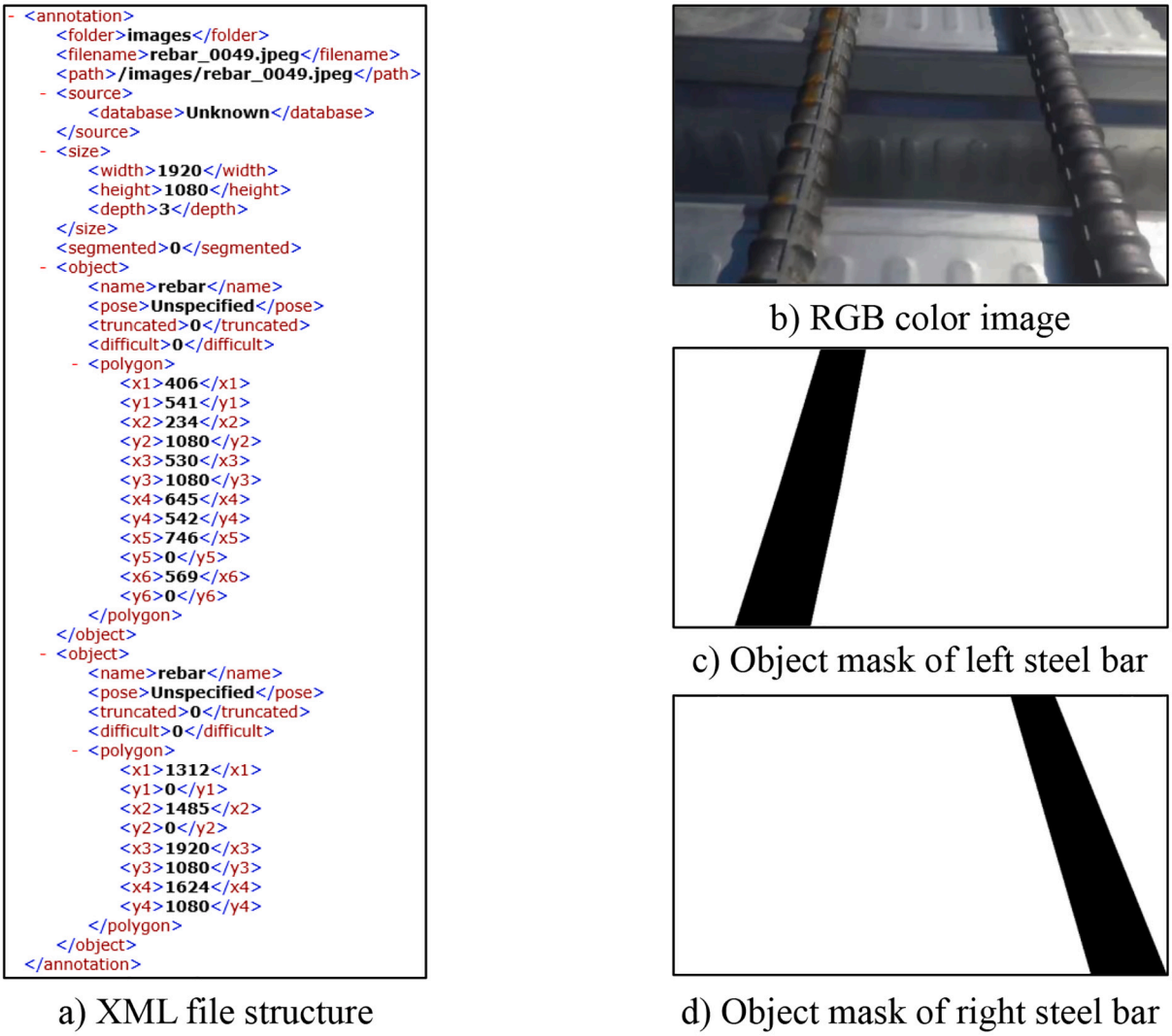

Fig. 5. Labeling of object mask polygon.

validation; an extra 10 images were used for testing. All the generated TFRecord files were uploaded to Google Cloud. The Google Cloud AI Platform was used for CNN training on a Tesla K80 GPU with the TensorFlow 1.10 runtime version.

Fig. 6 shows a learning curve with the training numbers (steps) on the horizontal axis and the mean Average Precision (mAP) on the vertical axis. This graph is used to evaluate the precision of the trained model and indicates how well the model performs object detection. Here, the mAP is a measure of model performance during training. Shanmugamani [37] stated that a detection is considered to be a true positive if the mAP value is above 0.5.

Fig. 7 shows a comparison of AI-QIM and manual labeling. Fig. 7(a) shows an AI-QIM-labeled image output obtained by using the test dataset as the input to the trained AI-QIM. Fig. 7(b) shows a manually labeled image, where polygons were drawn over the object boundaries of the steel bars.

After the initial training has been completed, the k-fold cross validation method was done to confirm the model prediction capability

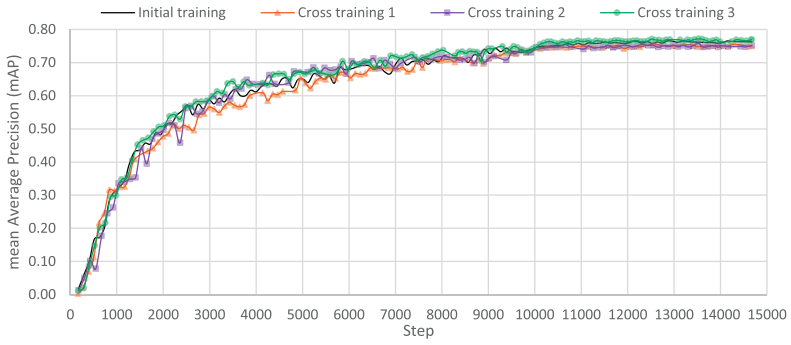

Fig. 6. AI-QIM training learning curves.

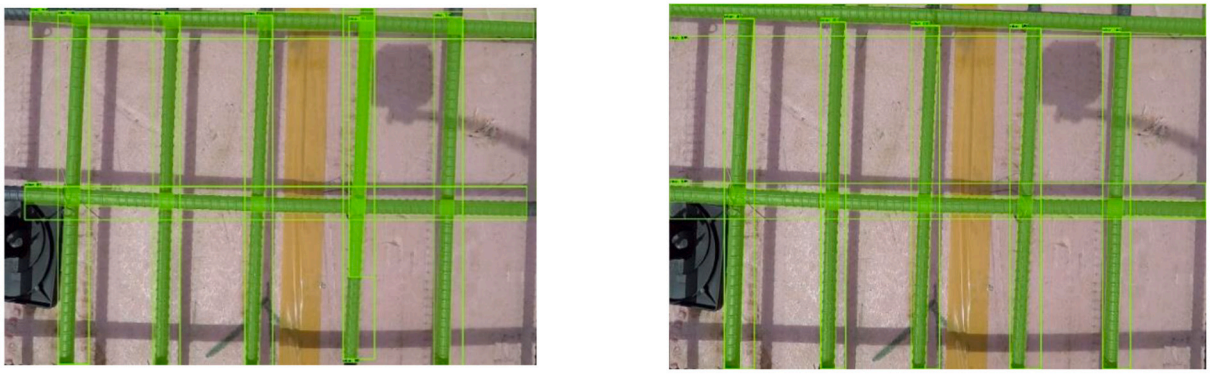

Fig. 7. Test set validation.

[38]. In this method, the dataset of 240 images was reshuffled in random orders to generate three extra datasets. Then, the AI-QIM was trained again using each of these datasets with 14,600 iterations over 23 h on average.

Fig. 6 also shows cross training learning curves along with the initial training learning curve. In the figure, each cross training learning curve reached their peak performance after approximately 10,000–11,000 steps. These results show that all the cross training achieved an average mAP = 0.75 with the lowest mAP = 0.75 and the highest mAP = 0.76. These mAP numbers are all above 0.5 and indicate the AI-QIM can perform consistently well in prediction.

4.4. Post-processing

The prediction capability of the AI-QIM can be improved by conducting post-processing to remove low-confidence predictions and overlapping predictions. The post-processing checks all of the pixel intersections for longitudinally and transversely placed steel bars to avoid improper computations of steel bar diameters and spacings. These errors occur when one of the points used for distance measurement lands on one of these intersections, which results in either inaccurate diameter measurement or spacing calculation.

Fig. 8(a) shows an example involving the fourth vertical steel bar from the left side, where two object masks are stacked on top of each other. In this study, the post-processing operation deleted objects of the same class when their object masks had more than 30% intersecting pixel values. Moreover, the post-processing operation deleted those detected object masks that had an area of more than 10% of total number of pixels on an image. This means that these steel bar image data were captured too close to the camera. When the image data were captured at distances of less than 30–40 cm away from the steel bars, no information was retrieved due to the limitations of stereo vision.

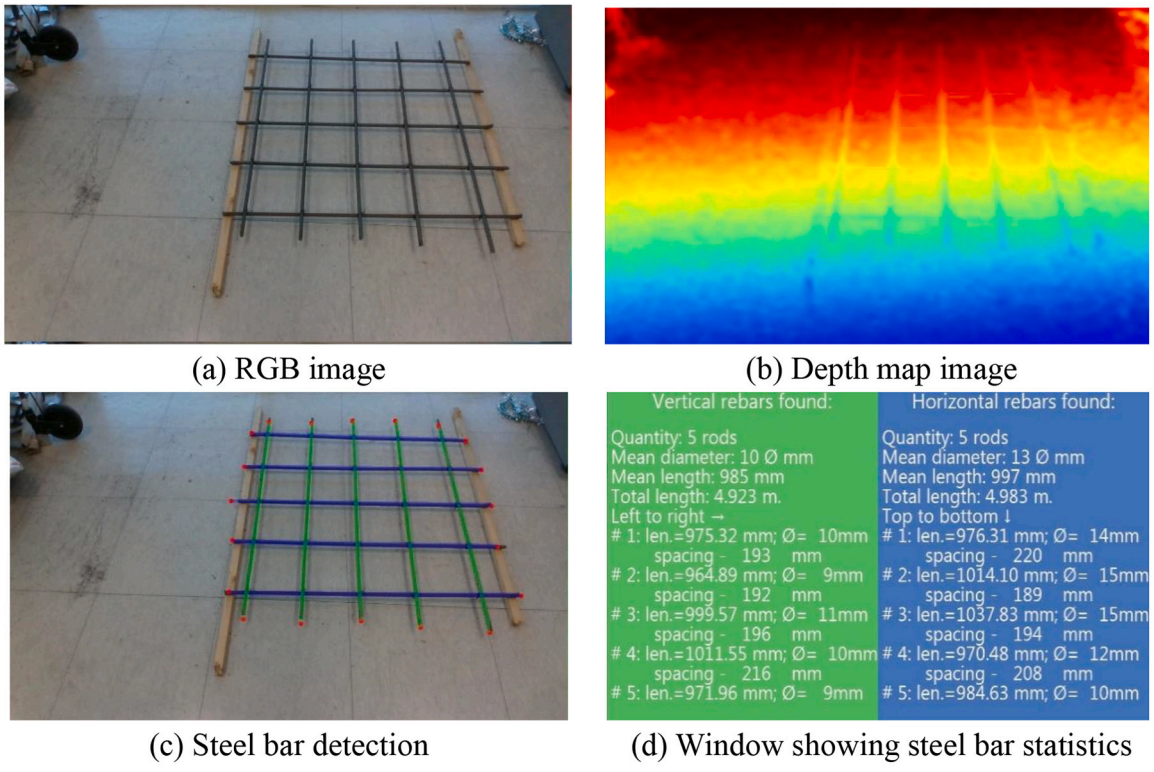

Fig. 8. Comprehensive input (upper half) output (lower half) window.

In addition, each object should have its own information separately in a proper object hierarchy because mask R-CNN invokes inferences in a random order. Therefore, the quantity, diameter, spacing, length of the steel bars were calculated sequentially from left to right and top to bottom. This post-processing was performed after the mask R-CNN inference in the first stage of the AI-QIM, and it can increase the predictability of the AI-QIM. Thus, the process can remove low-confidence predictions and overlapping predictions and enhance the prediction capability of the AI-QIM.

4.5. Estimation of steel bar object attributes

After training and post-processing, all of the required information can be calculated, including 1) quantity, 2) diameter, 3) spacing, and 4) length of the detected steel bars. First, it is relatively simple to count the number of steel bars because the quantity is equal to the total number of horizontally and vertically placed object masks. The steel bar length can be measured using the imaginary lines that should be placed over the object masks. The two tips at which the line intersects the object mask denote the end points of a steel bar.

Second, determination of steel bar diameter requires the same imaginary line on top of the object mask and a certain number (e.g., 20) of equidistant imaginary lines perpendicular to the pivot line, which are bounded by the object mask, thus yielding two points (from each normalized line) to measure the diameter. Then, the steel bar diameter can be calculated using eq. 3. However, since the object mask is usually marginally larger than the steel bar, these points will yield incorrect measurements. Therefore, these points should be compressed within the object mask to be closer to the pivot line and away from each side of mask edges by 5–10%.

Third, the bar spacing can be measured using the same imaginary pivot lines as those used to measure the length of the steel bars. In this case, the intersection points of the pivot line with its equidistant perpendicular lines constitutes a set of initial points. Fourth, the spacing distance can be calculated using eq. 3.

Fig. 8 shows detailed input and output images of the AI-QIM. Here, Fig. 8(a) shows an input RGB image; Fig. 8(b) a depth map image; Fig. 8 (c) the output object mask of the steel bars, as obtained using mask R-CNN; and Fig. 8(d) a computer window with statistics related to the steel bars. This window provides summary statistics about the quantity, length, diameter, and spacing of the vertical and horizontal steel bars.

5. Example case study

The AI-QIM was applied to a test bed comprising steel bar installation. For the evaluation, two 1 × 1 m testbeds comprising steel bars having two different diameters, 13 mm and 16 mm, were built. Fig. 9 shows a drawing representation of the assembled testbed. This testbed comprises 10 steel bars, with 5 steel bars placed vertically and 5 steel bars placed horizontally. The spacing between two steel bars was 200 mm. Small deviations of up to 10 mm were present when the steel bars were tied. Fig. 9(a) shows the top view, where "x" represents the horizontal distance from the camera to the edge of the testbed. Fig. 9(b) shows the side view, where "y" represents the vertical distance from the testbed to the stereo vision camera.

Accuracy tests were conducted at the [(x, y, z)] positions, where x denotes the horizontal distance from the camera, y the camera height, and z the side shift from the center of the testbed. The z value was varied from −20 cm to +20 cm during image capture to improve visibility. The observation camera was located at a height of 1.1 m, which is the typical height at which people hold mobile devices such as phones and tablets. This configuration provided appropriate visibility for capturing images of the steel bars used in this case study.

Performance evaluation of the AI-QIM was conducted at different ranges by using the testbed. Table 1 summarizes the results of the performance evaluation in terms of four performance factors: 1) quantity, 2) diameter, 3) spacing, and 4) length.

First, the AI-QIM estimated the quantity of steel bars accurately as 5 each in the horizontal direction and 5 each in the vertical direction. Second, the maximum relative error in the estimated diameter was 15.4%, and the absolute error ranged from 0 to 2 mm. However, the conditions at the test site were considerably good, thus providing good output results. Third, the estimated spacing between two steel bars had a maximum relative error of 3% and a 11% maximum error when detecting a single steel bar spacing distance between two bars within a range of 1 and 2 m from the placed steel bars. Fourth, the maximum relative error in the estimated steel bar length was 8%, with the absolute error ranging from 0 to 80 cm.

Overall, the AI-QIM showed fluctuations in estimating the diameter. These errors occurred due to the small image resolution of the depth map images, which was only 1280 × 720 px in this study. The depth map image resolution is important because a large pixel does not allow one to retrieve precise object positions, especially at longer distances. Yet another source of error is the detection model itself. That is, when the object mask is received, it has jagged edges that do not usually match with the positions of the steel bars. Some optical distortion also contributed to the error.

For example, the accuracy of a stereo vision camera gradually decreases with increasing distance from the object to be imaged. Therefore, the steel bars in the middle were detected quite precisely, whereas

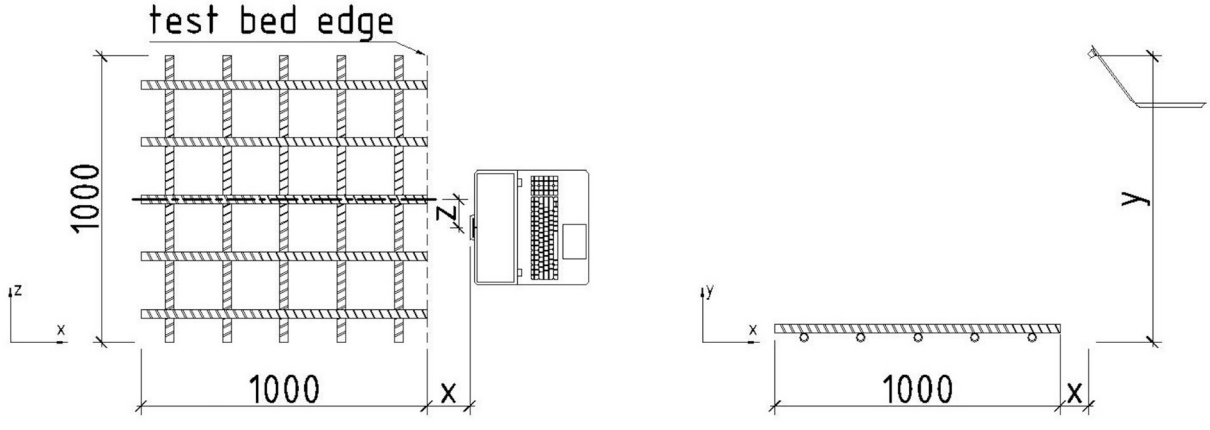

Fig. 9. Test bed setup (in millimeters).

Table 1
Evaluation table showing the AI-QIM performance results.

Distance to the test bed (mm)		# of steel bars				Diameter (mm)				Spacing (mm)				Length (m)			
X	Y	Actual	Detected	Absolute error	Relative error (%)	Actual	Detected	Absolute error	Relative error (%)	Actual	Detected	Absolute error	Relative error (%)	Actual	Detected	Absolute error	Relative error (%)
200 mm	1100 mm	10	10	0	0	13	11	2	15.4	200	198.4	1.6	0.8	10	9.8	0.2	2.0
		10	10	0	0	16	14	2	12.5	200	199.6	0.4	0.2	10	10.04	0.04	0.4
500 mm	1100 mm	10	10	0	0	13	12	1	7.7	200	199.6	0.4	0.2	10	10.0	0	0
		10	10	0	0	16	17	1	6.3	200	201.6	1.6	0.8	10	10.26	0.26	2.6
700 mm	1100 mm	10	10	0	0	13	13	0	0	200	204.4	4.4	2.2	10	10.32	0.32	3.2
		10	10	0	0	16	17	1	6.3	200	202.2	2.2	1.1	10	10.14	0.14	1.4
900 mm	1100 mm	10	10	0	0	13	13	0	0	200	203.4	3.4	1.7	10	10.26	0.26	2.67
		10	10	0	0	16	14	2	12.5	200	206.0	6.0	3.0	10	10.8	0.8	8.0

the ones to the sides were detected relatively imprecisely. Table 1 shows that the AI-QIM has better precision at shorter distances. Moreover, the object mask size is usually larger than the object itself, which adds to the error. Although these extra spaces were removed during post-processing, the error did affect model performance.

This case study demonstrates the three main benefits of the AI-QIM. First, the AI-QIM can assist the quality inspection process in steel bar installation because the real-time inference in the model can detect and segment steel bar objects directly at construction sites. As the case study results show, the automated quality inspection can help the human quality inspector to better understand the quantity, diameter, spacing, and length of the steel bars. Second, the quality inspector can use the AI-QIM in an unstructured environment at the construction site during steel bar installation. The quality inspector can videotape the status of steel bar installation and extract quality inspection data directly at the construction site. Third, the AI-QIM can provide mobility in quality inspection because the quality inspector needs to carry only the stereo vision camera and laptop computer or even a tablet to evaluate the quality status. The fifth-generation internet connection available nowadays enables the AI-QIM to be wirelessly connected to the internet from construction sites.

6. Conclusion

In this study, the AI-QIM is presented by integrating two state-of-the-art technologies: 1) mask R-CNN and 2) stereo vision. Based on artificial intelligence, the AI-QIM can detect steel bars that are installed for concrete placement. This combined effort is the first step in replacing manual and visual inspection with automated inspection. When the AI-QIM is fully implemented, this automated inspection system can assist a human inspector with steel bar inspection.

The case study indicated that the AI-QIM must be refined for full implementation in practice. Nevertheless, the AI-QIM can 1) reduce the labor requirement compared to manual and visual inspection; 2) assist the quality inspector with automated data output; 3) improve safety by reducing the need for human quality inspection to access dangerous work zones for observing the installation of steel bars; and 4) provide opportunities for various applications, such as drone inspection. Other than these benefits, the AI-QIM can reuse the data collected during steel bar inspection for training to improve its prediction capability.

In future, the object detection capability of the AI-QIM can be enhanced by overcoming certain limitations. For example, the low depth image resolution of 1280 × 720 px does not permit precise detection at distances greater than 2 m from the viewer. Moreover, the object mask precision decreases because of its jagged edges and the low image resolution, which introduces additional errors. Furthermore, the mAP of the AI-QIM can be improved by using a larger dataset for training. When these limitations are overcome, the AI-QIM can be generalized and potentially be used for steel bar inspection in practical scenarios.

Declaration of Competing Interest

- All authors have participated in (a) conception and design, or analysis and interpretation of the data; (b) drafting the article or revising it critically for important intellectual content; and (c) approval of the final version.
- This manuscript has not been submitted to, nor is under review at, another journal or other publishing venue.
- The authors have no affiliation with any organization with a direct or indirect financial interest in the subject matter discussed in the manuscript.

Acknowledgement

This work was supported by a National Research Foundation of

Korea (NRF) grant funded by the Korean Government (No. NRF-2019R1A2C201120212).

References

[1] C.R.S. Institute, Placing Reinforcing Bars Recommended Practices. https://books.google.co.kr/books?id=fXCFxwEACAAJ, Concrete Reinforcing Steel Institute, Schaumburg, Ill, , p. 1 (Band. (Accessed 12 May, 2021)).

[2] M.S. Shetty, A.K. Jain, Concrete Technology (Theory and Practice), 8 ed, S. Chand Publishing, 2019 (ISBN/ISSN: 9789352533800).

[3] X. Zhang, J. Zhang, M. Ma, Z. Chen, S. Yue, T. He, X. Xu, A high precision quality inspection system for steel bars based on machine vision, Sensors 18 (8) (2018), https://doi.org/10.3390/s18082732.

[4] K. Han, J. Gwak, M. Golparvar-Fard, K. Saidi, G. Cheok, M. Franaszek, R. Lipman, Vision-based field inspection of concrete reinforcing bars, in: 13th International Conference on Construction Applications of Virtual Reality, London, UK, Oct. 30–31, 2013, pp. 272–281. https://itc.scix.net/pdfs/convr-2013-28.pdf (Accessed 12 May, 2021).

[5] M.-K. Kim, J.P.P. Thedja, Q. Wang, Automated dimensional quality assessment for formwork and rebar of reinforced concrete components using 3D point cloud data, Automat. Construct. 112 (2020), 103077, https://doi.org/10.1016/j.autcon.2020.103077.

[6] W. Fang, L. Ding, B. Zhong, P.E.D. Love, H. Luo, Automated detection of workers and heavy equipment on construction sites: a convolutional neural network approach, Adv. Eng. Inform. 37 (2018) 139–149, https://doi.org/10.1016/j.aei.2018.05.003.

[7] Z. Kolar, H. Chen, X. Luo, Transfer learning and deep convolutional neural networks for safety guardrail detection in 2D images, Autom. Constr. 89 (2018) 58–70, https://doi.org/10.1016/j.autcon.2018.01.003.

[8] C.V. Dung, L.D. Anh, Autonomous concrete crack detection using deep fully convolutional neural network, Autom. Constr. 99 (2019) 52–58, https://doi.org/10.1016/j.autcon.2018.11.028.

[9] Z. Fan, J. Lu, B. Qiu, T. Jiang, K. An, A.N. Josephraj, C. Wei, Automated Steel Bar Counting and Center Localization With Convolutional Neural Networks, arXiv preprint arXiv:.00891, https://arxiv.org/abs/1906.00891, 2019 (Accessed 12 May, 2021).

[10] H. Yang, C. Fu, Quantity detection of steel bars based on deep learning, OALib 06 (10) (2019) 1–9, https://doi.org/10.4236/oalib.1105784.

[11] Y. Zhu, C. Tang, H. Liu, P. Huang, End-face localization and segmentation of steel bar based on convolution neural network, IEEE Access 8 (2020) 74679–74690, https://doi.org/10.1109/ACCESS.2020.2989300.

[12] Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324, https://doi.org/10.1109/5.726791.

[13] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778, https://doi.org/10.1109/cvpr.2016.90.

[14] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 580–587, https://doi.org/10.1109/CVPR.2014.81.

[15] K. He, G. Gkioxari, P. Dollar, R. Girshick, Mask R-CNN, IEEE Trans. Pattern Anal. Mach. Intell. (2018), https://doi.org/10.1109/TPAMI.2018.2844175. https://www.ncbi.nlm.nih.gov/pubmed/29994331.

[16] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: towards real-time object detection with region proposal networks, IEEE Trans. Pattern Anal. Mach. Intell. 39 (6) (2017) 1137–1149, https://doi.org/10.1109/TPAMI.2016.2577031.

[17] O. Angah, A.Y. Chen, Tracking multiple construction workers through deep learning and the gradient based method with re-matching based on multi-object tracking accuracy, Automat. Construct. (2020) 119, https://doi.org/10.1016/j.autcon.2020.103308.

[18] R. Kalfarisi, Z.Y. Wu, K. Soh, Crack detection and segmentation using deep learning with 3D reality mesh model for quantitative assessment and integrated visualization, J. Comput. Civ. Eng. 34 (3) (2020), https://doi.org/10.1061/(asce)cp.1943-5487.0000890.

[19] B. Kim, S. Cho, Image-based concrete crack assessment using mask and region-based convolutional neural network, Struct. Control Health Monitor. (2019), https://doi.org/10.1002/stc.2381.

[20] Z. Yang, Y. Yuan, M. Zhang, X. Zhao, Y. Zhang, B. Tian, Safety distance identification for crane drivers based on mask R-CNN, Sensors 19 (12) (2019), https://doi.org/10.3390/s19122789.

[21] Y. Wu, M. Hu, G. Xu, X. Zhou, Z. Li, Detecting leakage water of shield tunnel segments based on mask R-CNN, in: 2019 IEEE International Conference on Architecture, Construction, Environment and Hydraulics (ICACEH), 2019, pp. 25–28, https://doi.org/10.1109/icaceh48424.2019.9042088.

[22] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft COCO: Common Objects in Context, Computer Vision – ECCV 2014, 2014, pp. 740–755, https://doi.org/10.1007/978-3-319-10602-1_48.

[23] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2117–2125. https://openaccess.thecvf.com/content_cvpr_2017/html/Lin_Feature_Pyramid_Networks_CVPR_2017_paper.html (Accessed 12 May, 2021).

[24] K. Adi, C. Widodo, Distance measurement with a stereo camera, Int. J. Innovat. Res. Adv. Eng. 4 (11) (2017) 24–27, https://doi.org/10.26562/IJIRAE.2017.NVAE10087.

[25] C. Kollmitzer, Object Detection and Measurement Using Stereo Images, Multimedia Communications, Services and Security, 2012, pp. 159–167, https://doi.org/10.1007/978-3-642-30721-8_16.

[26] Y.M. Mustafah, R. Noor, H. Hasbi, A.W. Azma, Stereo vision images processing for real-time object distance and size measurements, in: 2012 International Conference on Computer and Communication Engineering (ICCCE), 2012, pp. 659–663, https://doi.org/10.1109/iccce.2012.6271270.

[27] M. Zivingy, Object distance measurement by stereo vision, Int. J. Sci. Appl. Inform. Technol. 2 (2013), 05–08, 2278–3083, https://www.researchgate.net/profile/Manaf-Zivingy/publication/305308988_Object_distance_measurement_by_stereo_vision/links/5788c12d08aeef933e1b9b35/Object-distance-measurement-by-stereo-vision.pdf (Accessed 12 May, 2021).

[28] B. Shan, S. Zheng, J. Ou, A stereovision-based crack width detection approach for concrete surface assessment, KSCE J. Civ. Eng. 20 (2) (2015) 803–812, https://doi.org/10.1007/s12205-015-0461-6.

[29] T. Yokoyama, T. Matsumoto, Development of Stereo Image Analysis for Measuring Small Deformation, Proc. Eng. 171 (2017) 1256–1262, https://doi.org/10.1016/j.proeng.2017.01.419.

[30] C. Kim, W. Lee, Developing stereo-vision based drone for 3D model reconstruction of collapsed structures in disaster sites, J. Korea Acad. Industr. Cooperat. Soc. 17 (6) (2016) 33–38, https://doi.org/10.5762/kais.2016.17.6.33.

[31] R.C. Jain, R. Kasturi, B.G. Schunck, Machine Vision, McGraw-Hill, New York etc, 1995 (ISBN/ISSN: 0-07-032018-7).

[32] A.N. Belbachir, Smart Cameras, Springer US, 2009 (ISBN/ISSN: 9781441909534).

[33] Intel, Intel® RealSense™ D400 Series Product Family Datasheet. https://dev.intelrealsense.com/docs/intel-realsense-d400-series-product-family-datasheet, 2020 (Accessed 12 May, 2021).

[34] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, K. Murphy, Speed/accuracy trade-offs for modern convolutional object detectors, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 3296–3297, https://doi.org/10.1109/CVPR.2017.351.

[35] D. Acharya, A. Gowreddygari, R. Bhatia, V. Shaju, S. Aparna, A. Bhardwaj, Epileptic seizure detection using CNN, Adv. Comput. (2021) 3–16, https://doi.org/10.1007/978-981-16-0401-0_1.

[36] S. Raschka, Python Machine Learning, Packt Publishing, 2015 (ISBN/ISSN: 9781783555147).

[37] R. Shanmugamani, Deep Learning for Computer Vision: Expert techniques to train advanced neural networks using TensorFlow and Keras, Packt Publishing Ltd, 2018 (ISBN/ISSN: 1788293355).

[38] A. Fuentes, Mastering Predictive Analytics With Scikit-learn and TensorFlow: Implement Machine Learning Techniques to Build Advanced Predictive Models Using Python, Packt Publishing, 2018 (ISBN/ISSN: 9781789612240).