Point Cloud Semantic Segmentation of Concrete Surface Defects Using Dynamic Graph CNN

F. Bahreini[a] and A. Hammad[b]

[a]Department of Building, Civil & Environmental Engineering, Concordia University, Canada
[b]Concordia Institute for Information Systems Engineering, Concordia University, Canada
E-mail: F_bahrei@encs.concordia.ca, Amin.hammad@concordia.ca

Abstract –

Obtaining accurate information of defective areas of infrastructures helps to perform repair actions more efficiently. Recently, LiDAR scanners are used for the inspection of surface defects. Moreover, machine learning methods have attracted the attention of researchers for semantic segmentation and classification based on point cloud data. Although much work has been done in the area of computer vision based on images, research on machine learning methods for point cloud semantic segmentation is still in its early stages, and the current available deep learning methods for semantic segmentation of the concrete surface defects are based on converting point clouds to images or voxels. This paper proposes an approach for detecting concrete surface defects (i.e. cracks and spalls) using a Dynamic Graph Convolutional Neural Network (Dynamic Graph CNN) model. The proposed method is applied to a point cloud dataset from four concrete bridges in Montreal. The experimental results show the usefulness and robustness of the proposed method in detecting concrete surface defects from 3D point cloud data. Based on the sensitivity analysis of the model using three cases defined with different number of input points, the best test results show the detection recall for cracks and spalls are 55.20% and 89.77%, respectively.

Keywords –

Concrete Surface Defect; Semantic Segmentation; 3D Point Cloud; Dynamic Graph CNN

1 Introduction

Many of the old infrastructures that are near the end of their service life are still in use, which increases the need for regular inspection of these structures [1, 2]. Advanced technologies (e.g. LiDAR scanners, sensors) have made the inspection process more accurate and reliable [3, 4]. These technologies, such as LiDAR scanning, are a promising alternative to traditional visual inspection, which is unsafe, labor-intensive, costly, and subject to human errors [5]. Although much work has been done for processing images using computer vision, research on machine learning methods for point cloud semantic segmentation is still in its early stages [6]. Image-based methods have limitations, such as the need for appropriate lighting conditions and additional information to analyze images (such as focal length), and may also fail to analyze more complex geometric surfaces [7, 8, 9]. Furthermore, using the methods that are not applying raw point clouds as input will increase the dataset size by converting the point cloud to other data formats, resulting in missing information or causing heavy computing [6, 10]. Deep learning is one of the most effective machine learning techniques, which uses more than two hidden layers in order to acquire high-dimensional features from the training data [11, 12]. So far, deep learning with images has achieved acceptable results by learning complex structures. Currently, researchers are adapting these methods by using point cloud data as raw input.

In order to automate the inspection process, this paper proposes an approach using the 3D semantic segmentation technique by adapting a Dynamic Graph Convolutional Neural Network (Dynamic Graph CNN) model. The main purpose of the automated inspection process in this paper is to detect concrete surface defects, including cracks and spalls. The proposed method mainly consists of five steps: (1) data collection, (2) manual annotation, (3) data pre-processing, (4) training and evaluation, and (5) testing.

The rest of the paper is structured as follows: Section 2 contains the literature review. The methodology is explained in Section 3. Section 4 shows a case study and experimental results. Finally, the conclusions and future work are presented in Section 5.

2 Literature Review

2.1 LiDAR-based Defect Detection

LiDAR scanning is a non-contact measurement

technology that has proven its potential in capturing accurate and instant point cloud data from object surfaces [1, 13]. However, the resolution and noise level of point cloud data pause some challenges in detecting small cracks [14]. Therefore, to overcome this limitation, an additional feature, which is the RGB color, is considered in deep learning models [5, 6].

Laefer et al. [9] used fundamental mathematics to define the smallest width of unit-based masonry cracks, which can be detected with terrestrial laser scanner by considering the main parameters of depth and orientation of crack, orthogonal offset, and interval scan angle. Anil et al. [15] focused on the performance of laser scanners by using an automated algorithm on point cloud data from reinforced concrete surfaces and asserted the possibility of detecting 1 *mm* crack based on point cloud data. Xu and Yang [16] used the Gaussian filtering method and image-generated data from the point cloud to detect the cracks of a concrete tunnel structure. Teza et al. [17] proposed an automatic method for the inspection of damaged areas of concrete bridge surfaces using a laser scanner and Gaussian mean curvature computation. Makuch and Gawronek [18] proposed an automatic inspection system for reinforced concrete cooling tower shells using point cloud data and local surface curvature computation. Olsen et al. [19] proposed using cross sectional analysis to detect surface damage based on laser scanner data. Liu et al. [20] utilized the distance and gradient-based method to detect the defective area of bridge surfaces using laser scanner data. Valença et al. [21] proposed a method combining image processing and terrestrial laser scanning technology to automate the process of capturing the geometrical characteristics of cracks on concrete bridges. Kim et al. [22] proposed a technique to indicate the location and measure the quantity of concrete surface spalling defects larger than 3 *mm* using laser scanner data. Truong-Hong et al. [13] presented an approach to detect the bridge cracks using a terrestrial laser scanner and developed a tool to measure the length and width of cracks based on point cloud data and RGB color produced from an external camera. Tsai et al. [23] assessed the probability of using point cloud data to detect cracks with the dynamic-optimization-based segmentation method and assessing the crack segmentation performance using the linear-buffered Hausdorff scoring method. Cabaleiro et al. [24] developed an automatic crack detection algorithm using LiDAR data for timber beams inspection to identify the crack geometrical characteristics. Mizoguchi et al. [25] proposed a customized region-growing algorithm along with an iterative closest point algorithm to detect the surface defects of concrete structures based on laser scanner data. Nasrollahi et al. [26] proposed a method for detecting concrete surface defects based on collecting point cloud data from LiDAR scanners and using a Deep Neural Network (DNN). Guldur and Hajjar [27] developed damage detection algorithms for automatic surface normal-based defect detection and quantification using LiDAR scanner data.

2.2 Dynamic Graph CNN

DNNs or deep feedforward networks utilize multiple deep layers along with highly optimized algorithms to learn from trained data sets without the process of manual feature extraction [28]. A CNN is a class of DNNs containing input, convolutional, subsampling, and output layers [29].

Dynamic Graph CNN, proposed by Wang et al. [30], is a new point-based CNN suitable for high-level tasks, such as object classification and semantic segmentation. Dynamic Graph CNN can improve capturing local geometric functions as it creates a local neighborhood graph and dynamically updates the graph with the nearest neighbors after each layer of the network. Rather than operating on individual points, the model iteratively performs convolution on edges associating the neighborhood point pairs. The operation layer for edge feature generation is called EdgeConv, which defines the relationships between a point and its neighbors. Figure 1 shows the mechanism of Dynamic Graph CNN edge feature generation. As shown in Figure 1(a), X_i and X_j are a point pair, and e_{ij} is $h_\theta(X_i, X_j)$, which is the edge feature function; h is the function parameterized by the set of learnable parameters θ. Figure 1(b) shows the channel-wise symmetric aggregation operation on the edge features associated with all the edges originating from each vertex, where X_i' is the EdgeConv operation, which is defined by applying asymmetric aggregation operation at the *i-th* vertex. The segmentation model of Dynamic Graph CNN involves a series of three EdgeConv layers and three fully connected layers. The parameter k in the model is the number of the edge features for each point, which is computed in each EdgeConv layer for the input of n points.

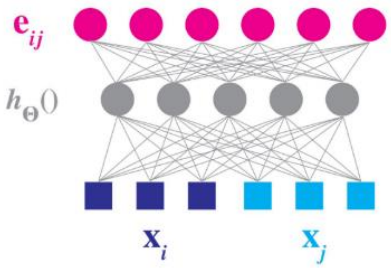
3 Methodology

Dynamic Graph CNN, originally designed to detect indoor building elements, is modified and adapted to automate the inspection process of concrete surface defects, including cracks and spalls. This model is selected because it considers the edge feature, which is the most valuable feature in concrete surface defects detection. Figure 2 shows the proposed method for 3D point cloud-based concrete surface defects detection.

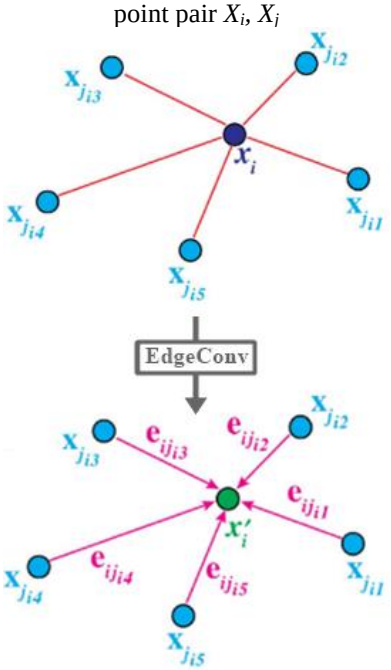
The following five main steps are used to automate the inspection process of concrete surface defects:

(1) Data collection: The geometric features of defects, particularly the depth, play a significant role in extracting

important features and having practical results. Therefore, data collection is an important step and has to be done accurately. The scanner position and the scanning parameters, such as resolution, quality, Field of View (FOV), and the number of scanned points, are the factors that can affect the visibility of defects in the collected point cloud data.



(a) Computing an edge feature e_{ij} from a neighboring point pair X_i, X_j



(b) Channel-wise asymmetric aggregation operation on the edge features associated with all the edges originating from each vertex

Figure 1. Mechanism of Dynamic Graph CNN edge feature generation [30]

(2) Manual 3D point cloud annotation: After data collection, the selected parts need to be manually annotated based on the types of targeted surface defects. In this paper, two main types of surface defects, which are cracks and spalling, are considered. Each part of the dataset is annotated into three categories of crack, spalling, and non-defect.

(3) Data pre-processing: The annotated dataset is prepared and augmented by adding flipped data. The original dataset files are converted into data label files, which are *2D* matrices with *XYZRGBL* in each line. Then, each part is split into blocks, and for each block, normalized location values on the *Y* surface are added [30]. Each point is represented as a 7*D* vector of *XYZ*, *RGB*, and N_y. The normalized location values over *X* and *Z* directions are not considered as the depth of defects is in the direction of the *Y*-axis, and normalized location values over *X* and *Z* directions are not valuable and mislead the network's learning process. The sizes of blocks are defined based on the sizes of the structural defects in the dataset. Hence, the selected block sizes in the data pre-processing step are assumed to be less than 40 *cm* × 40 *cm* on the *XZ* surface, with the depth of the defects as the third dimension, which is equal to the depth of the deepest defect in each segment. Moreover, in this step, the wrapped and normalized points inside the blocks are converted to Hierarchical Data Format (HDF) [31], and HDF5 files are used for the training process in the next step.

(4) Training and evaluation: As discussed in Section 2.2, a series of three EdgeConv layers followed by three fully-connected layers are included in the segmentation model of Dynamic Graph CNN, and the number of the *k*-nearest neighbors of a point for EdgeConv layers is specified for the input of *n* points in the model. In the adapted Dynamic Graph CNN, the input points variables are changed from a 9-dimensional vector to a 7-dimensional vector by removing the normalized location values of the *x*-axis and *z*-axis, and the network is fed by 7-dimensional input data. As the defect's numbers of points in this paper are less than the non-defect number of points, which is known as the issue of "imbalanced datasets", a weighted softmax loss function is utilized to adapt the model to our prepared dataset, and the corresponding weight vector is set based on the points distribution among the three classes.

(5) Testing: To validate the model accuracy, the unseen parts of the dataset, which are not used in Step 4, are used for the testing step. The confusion matrix is used to describe the model's performance using the equations presented in Table 1. In this paper, the term "overall accuracy" refers to the percentage of correct predictions for the test data. Furthermore, the recall is assumed to be more relevant than precision as the process of concrete surface inspection aims to minimize the chance of missing actual defect points, which can be achieved by minimizing the "False Negative" prediction of the model.

4 Case Study and Implementation

This paper used point cloud datasets from four reinforced concrete bridges in Montreal, scanned using a FARO

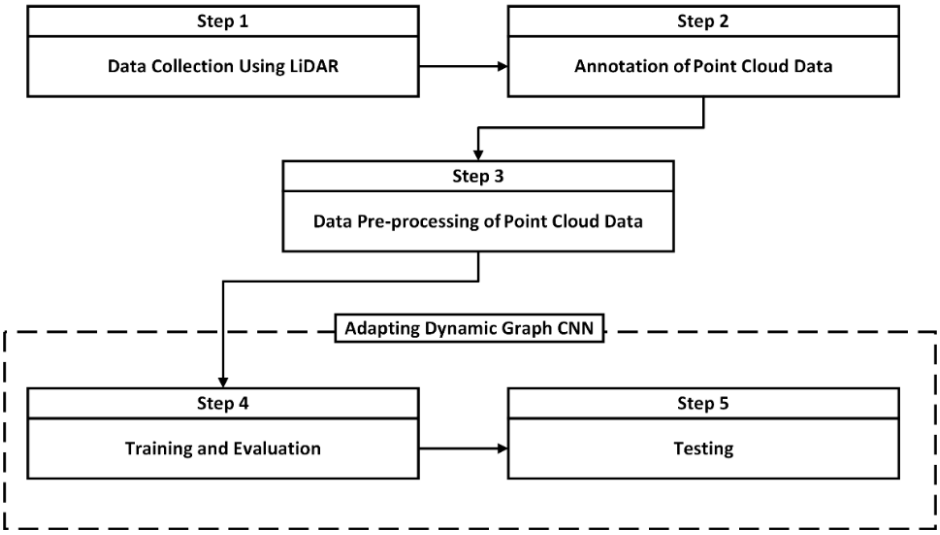


Figure 2. Proposed method for 3D point cloud-based defect detection

Table 1. Model Performance metrics

Performance metrics	Equation
Precision	$\frac{TP}{TP + FP}$
Recall	$\frac{TP}{TP + FN}$
F1 score	$2 \times \frac{Precision \times Recall}{Precision + Recall}$
Intersection over Union (IoU)	$\frac{TP}{FP + TP + FN}$
Overall accuracy	$\frac{TP_{Crack} + TP_{Spall} + TP_{crack}}{All\ perdiction\ of\ the\ points}$

Note: *TP* refers to true positives, *FP* refers to false positives, and *FN* refers to false negatives

Focus3D scanner [32]. Table 2 shows the scanning parameters. Furthermore, CloudCompare software [33] is used to register and eliminate the irrelevant points of the point cloud data. The scanning process in this step is affected by several factors, such as the battery capacity and performance limitations, especially in severe weather conditions, scanning time, and traffic constraints. For this reason, different settings, including different numbers of stations, were used to scan each of the bridges. In some scans, the FOV was reduced to avoid scanning irrelevant objects (e.g. moving vehicles).

The prepared dataset includes 102 selected segmented parts from the scanned bridges. The number of annotated cracks in the selected parts is 595, and the number of annotated spalls is 773. The annotation process is done manually in CloudCompare software using the following rules based on experience: (1) a specific range of 150,000 pts to 400,000 pts is considered for the number of points of each selected part; (2) the scanned surfaces are classified into rectangular parts because of the box shape of the blocks in the model; and (3) the part size should consider the higher density of points in some parts and it should not contain more than the maximum defined number of points, which is 400,000 pts. The annotated datasets are split into five areas. Area 1 to 3 are used for training, Area 4 is used for evaluation, and Area 5 is dedicated to testing. The *X*-axis is set along the concrete surface, the *Z*-axis is set in the vertical direction of the canonical coordinate system, and the *Y*-axis is set perpendicular to the surface and in the direction of the depth of the defects. The depth of defects is set to have positive *Y* values. Furthermore, in this paper, to enlarge the size of the dataset, the augmenting method of flipping the point cloud data is used. In this regard, the annotated parts are flipped with respect to the *YZ* plane. The total number of segmented parts after adding the flipped data is 204 parts. The statistical information of the dataset, including the flipped data, is given in Table 3.

Wang et al. [30] used the block size of 1 $m \times 1\ m$ on the *XY* surface for rooms with a height of 3 *m* to detect indoor building elements. The number of points of 4,096 is used for their training process. This setting results in a very low density of points for detecting most types of defects in this paper (e.g. medium-sized spalls). In the adapted Dynamic Graph CNN, the block size of 40 $cm \times 40\ cm$ is set based on the sizes of the structural defects in the dataset. Moreover, the density of points in each block is increased by raising the number of points. This paper defines three cases with different number of input points of 8,192, 10,240, and 12,288, which are sampled for each block during the training process.

Table 2. Scanning parameters of four scanned bridges in Montréal

Scans		Number of Stations	Resolution	Quality	Horizontal FoV	Vertical FoV	Number of Points (Mpts)
Bridge 1	Scan 1	8	1/4	6x	23° to 259°	-42.5° to 71°	25.5
	Scan 2	4	1/4	6x	23° to 259 °	-42.5° to 71°	25.5
Bridge 2	Scan 3	6	1/1	2x	0° to 360°	-60° to 90°	710.7
Bridge 3	Scan 5	4	1/2	4x	0° to 360°	-45° to 71°	134.5
Bridge 4	Scan 6	2	1/2	4x	0° to 360°	-60° to 90°	177.7

Table 3. The statistics of the prepared dataset

Dataset		Number of segmented parts	Number of points	Defects				Non-defects
				Crack		Spalling		Number of points
				Number of cracks	Number of points	Number of spalls	Number of points	
Training (59.5%)	Area 1	32	10,418,902	264	104,256	226	715,768	9,598,878
	Area 2	44	11,003,768	334	112,436	266	282,822	10,608,510
	Area 3	42	10,651,316	160	67,714	356	744,356	9,839,246
Evaluation (19.6%)	Area 4	44	10,552,584	192	80,454	328	762,156	9,709,974
Testing (20.9%)	Area 5	42	11,257,240	240	128,538	370	1,365,228	9,763,474
Total		204	53,883,810	1,190	493398	1,546	3,870,330	49,520,082

The number of the *k*-nearest neighbors of a point for EdgeConv layers is set equal to 20 following the suggested value by Wang et al. [30].

The training and evaluation results, including the overall accuracy and mean loss of defined cases, are presented in Table 4. Precision, recall, F1 score, IoU, and overall accuracy are calculated to evaluate detection results for the defined cases. The test results from the three samples of 3D point cloud semantic segmentation of adapted Dynamic Graph CNN are shown in Figure 3. The test results (Table 5) show the detecting recall for cracks for spalls for Case A (8,192 points) are 55.20% and 89.77%, respectively. Increasing the number of points from 8,192 to 12,288 improved the crack detection recall from 55.20% to 58.67%. However, this increase resulted in decreasing the spall recall from 89.77% to 87.40%, and non-defect recall from 97.17% to 96.64%. This is because increasing the number of points sometimes can cause overfitting.

Furthermore, as the depth of each segmented part are different, and the learning process depends on the maximum depth of the part's defects, the recall result of the tests is categorized based on the depth of segmented parts used in the test. As shown in Table 6, the parts with more than 7 cm depth can increase recall up to 80.04% for crack and 93.33% for spall.

5 Conclusions and Future Work

This paper proposes an approach using the 3D semantic segmentation technique and a modified Dynamic Graph CNN model to automate the inspection process of concrete surface defects, including cracks and spalls. The prepared dataset includes 204 segmented parts from four scanned concrete bridges in Montreal. Three types of segments (i.e. crack, spall, and non-defect) are annotated in the training dataset. The performance of the network is improved by modifying the setting of the network (e.g. modifying the loss function) and by augmenting the dataset (i.e. by flipping the point cloud data).

The case study shows the usefulness and robustness of the proposed method in detecting concrete surface defects from 3D point cloud data. The best test results show the detection recall for cracks and spalls are 55.20% and 89.77%, respectively.

The small size of the dataset is one of the main limitations of this paper, and a larger dataset is expected to improve the learning process resulting in better performance and accuracy of the model. Moreover, due to computing resource limitation (i.e. memory and processors limitation), it was not possible to study the effect of increasing the number of input points of the model to more than 12,288 and increasing the density of sampled points in each block by reducing the size of each

Table 4. Training and evaluation results

Case	Number of sampled points for each block	Block size (cm)	Training		Evaluation		Training time
			Mean loss	Overall accuracy (%)	Mean loss	Overall accuracy (%)	
A	8,192	40×40	0.0022	97.54	0.0081	97.50	13h 44m
B	10,240	40×40	0.0024	97.39	0.0090	97.65	16h 35m
C	12,288	40×40	0.0030	97.04	0.0082	96.88	20h 18m

Table 5. Testing results (%)

Case	Overall accuracy	Crack				Spalling				Non-defect			
		Precision	Recall	F1 score	IOU	Precision	Recall	F1 score	IOU	Precision	Recall	F1	IOU
A	95.94	69.98	55.20	61.76	44.68	79.30	89.77	84.2	72.72	98.54	97.17	97.85	95.79
B	95.59	68.95	55.31	61.38	44.28	77.47	89.41	83.0	71.0	98.48	96.82	97.64	95.39
C	95.24	49.73	58.67	53.83	36.83	77.00	87.40	81.9	69.3	98.48	96.64	97.55	95.22

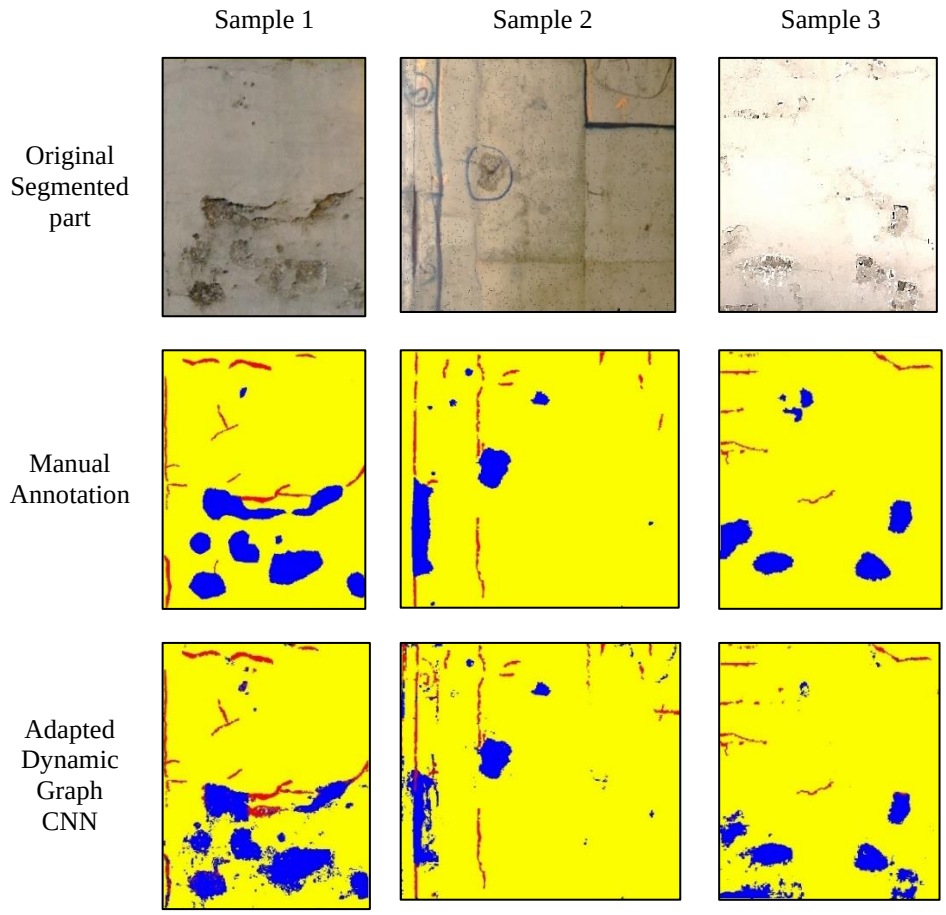


Figure 3. Test results from three samples of 3D point cloud semantic segmentation

Table 6. Defect detection recall based on the depth of defects (%)

Case	Depth (cm)					
	D≤3		3<D<7		7≤D	
	Crack	Spall	Crack	Spall	Crack	Spall
A	35.22	90.78	44.87	87.59	76.91	92.68
B	36.65	88.48	42.52	86.99	79.00	93.22
C	39.22	81.91	48.07	84.39	80.04	93.33

block. Future work will focus on collecting and preparing more data to enlarge the dataset. The proposed method can also be applied to other types of concrete surface defects and other types of material surfaces.

Acknowledgments

We would like to thank Mr. Majid Nasrollahi and Ms. Neshat Bolourian for their help in data collection, manual annotation, and contributions to a related paper.

References

[1] Rashidi, M., Mohammadi, M., Sadeghlou Kivi, S., Abdolvand, M.M., Truong-Hong, L. and Samali, B., "A decade of modern bridge monitoring using terrestrial laser scanning: Review and future directions," *Remote Sensing*, 12(22):3796/1-34, 2020.

[2] Riveiro, B. and Lindenbergh, R. eds., *Laser Scanning: An Emerging Technology in Structural Engineering*, CRC Press, Leiden, Netherland, 2019.

[3] Jovančević, I., Pham, H.H., Orteu, J.J., Gilblas, R., Harvent, J., Maurice, X. and Brèthes, L., "3D point cloud analysis for detection and characterization of defects on airplane exterior surface," *Journal of Nondestructive Evaluation*, 36(4):1-17, 2017.

[4] Balaguer, C., Gimenez, A. and Abderrahim, C.M., "ROMA robots for inspection of steel based infrastructures," *Industrial Robot*, 29(3):246-251, 2002.

[5] Guldur, B., Yan, Y. and Hajjar, J.F., "Condition assessment of bridges using terrestrial laser scanners," In *Proceedings of Structures Congress on Bridges and Other Structures*, pages 355-366, Portland, Oregon, 2015.

[6] Liu, W., Sun, J., Li, W., Hu, T. and Wang, P., "Deep learning on point clouds and its application: A survey," *Sensors*, 19(9):4188/1-22, 2019.

[7] Koch, C., Georgieva, K., Kasireddy, V., Akinci, B. and Fieguth, P., "A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure," *Advanced Engineering Informatics*, 29(2):196-210, 2015.

[8] Smith, C.J. and Adendorff, K.K., "Advantages and limitations of an automated visual inspection system," *The South Africa Journal of Industrial Engineering*, 5(1):27-36, 1991.

[9] Laefer, D.F., Truong-Hong, L., Carr, H. and Singh, M., "Crack detection limits in unit based masonry with terrestrial laser scanning," *Ndt & E International*, 62(1):66-76, 2014.

[10] Grilli, E., Menna, F. and Remondino, F., "A review of point clouds segmentation and classification algorithms," In *Proceedings of the ISPRS International Archives on the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pages 339-344, Wuhan, China, 2017.

[11] Xie, Y., Tian, J. and Zhu, X.X., "Linking points with labels in 3D: A review of point cloud semantic segmentation," *IEEE Geoscience and Remote Sensing Magazine*, 8(4):38-59, 2020.

[12] Zhu, X.X., Tuia, D., Mou, L., Xia, G.S., Zhang, L., Xu, F. and Fraundorfer, F., "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geoscience and Remote Sensing Magazine*, 5(4):8-36, 2017.

[13] Truong-Hong, L., Falter, H., Lennon, D. and Laefer, D.F., "Framework for bridge inspection with laser scanning," In *Proceedings of the EASEC-14 Conference on Structural Engineering and Construction*, pages 6-8, Minh City, Vietnam, 2016.

[14] Tang, P., Akinci, B. and Huber, D., "Quantification of edge loss of laser scanned data at spatial discontinuities," *Automation in Construction*, 18(8):1070-1083, 2009.

[15] Anil, E.B., Akinci, B., Garrett, J.H. and Kurc, O., "Characterization of laser scanners for detecting cracks for post-earthquake damage inspection," In *Proceedings of the International Symposium on Automation and Robotics in Construction*, pages 1-8, Montreal, Canada, 2013.

[16] Xu, X. and Yang, H., "Intelligent crack extraction and analysis for tunnel structures with terrestrial laser scanning measurement," *Advances in Mechanical Engineering*, 11(9), pp. 1-7, 2019.

[17] Teza, G., Galgaro, A. and Moro, F., "Contactless recognition of concrete surface damage from laser scanning and curvature computation," *NDT & E International*, 42(4):240-249, 2009.

[18] Makuch, M. and Gawronek, P., "3D point cloud analysis for damage detection on hyperboloid cooling tower shells," *Remote Sensing*, 12(10): 1542/1-23, 2020.

[19] Olsen, M.J., Kuester, F., Chang, B.J. and Hutchinson, T.C., "Terrestrial laser scanning-based structural damage assessment," *Journal of Computing in Civil Engineering*, 24(3):264-272, 2010.

[20] Liu, W., Chen, S. and Hauser, E., "LiDAR-based

bridge structure defect detection," *Experimental Techniques*, 35(6):27-34, 2011.

[21] Valença, J., Puente, I., Júlio, E., González-Jorge, H. and Arias-Sánchez, P., "Assessment of cracks on concrete bridges using image processing supported by laser scanning survey," *Construction and Building Materials*, 146(1):668-678, 2017.

[22] Kim, M.K., Sohn, H. and Chang, C.C., "Localization and quantification of concrete spalling defects using terrestrial laser scanning," *Journal of Computing in Civil Engineering*, 29(6):04014086, 2015.

[23] Tsai, Y.C.J. and Li, F., "Critical assessment of detecting asphalt pavement cracks under different lighting and low intensity contrast conditions using emerging 3D laser technology," *Journal of Transportation Engineering*, 138(5):649-656, 2012.

[24] Cabaleiro, M., Lindenbergh, R., Gard, W.F., Arias, P. and Van de Kuilen, J.W.G., "Algorithm for automatic detection and analysis of cracks in timber beams from LiDAR data," *Construction and Building Materials*, 130(1):41-53, 2017.

[25] Mizoguchi, T., Koda, Y., Iwaki, I., Wakabayashi, H., Kobayashi, Y., Shirai, K., Hara, Y. and Lee, H.S., "Quantitative scaling evaluation of concrete structures based on terrestrial laser scanning," *Automation in Construction*, 35(1):263-274, 2013.

[26] Nasrollahi, M., Bolourian, N. and Hammad, A., "Concrete surface defect detection using deep neural network based on lidar scanning," In *Proceedings of the CSCE Annual Conference on AI and machine learning*, pages 12-15, Montreal, Canada, 2019.

[27] Erkal, B.G. and Hajjar, J.F., "Laser-based surface damage detection and quantification using predicted surface properties," *Automation in Construction*, 83(1):285-302, 2017.

[28] Shrestha, A. and Mahmood, A., "Review of deep learning algorithms and architectures," *IEEE Access*, 7(1):53040-53065, 2019.

[29] LeCun, Y. and Bengio, Y., "Pattern recognition and neural networks," *The Handbook of Brain Theory and Neural Networks*, MIT Press, Cambridge, MA, USA, 1995.

[30] Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M. and Solomon, J.M., "Dynamic graph CNN for learning on point clouds," *ACM Transactions on Graphics*, 38(5):1-12, 2019.

[31] HDF-Group, "HDF5 User's Guide," 2018. Online: https://portal.hdfgroup.org/display/HDF5/HDF5+User%27s+Guide, Accessed: 04/May/2021.

[32] FARO Technologies Inc., "Faro Laser Scanner Focus 3D X 130 -NEO-Tech," 2010. Online: https://www.faro.com/products/construction-bim-cim/faro-focus/, Accessed: 01/May/2020.

[33] Girardeau-Montaut, D., Roux, M., Marc, R. and Thibault, G., "Change detection on points cloud data acquired with a ground laser scanner," In *Proceedings of the International Archives on Photogrammetry, Remote Sensing and Spatial Information Sciences*, pages 1-6, Hannover, Germany, 2005.