Automated Classification of Detected Surface Damage from Point Clouds with Supervised Learning

Burcu Guldur[1] and Jerome F. Hajjar[2]
[1]Assistant Professor, Department of Civil Engineering, Hacettepe University, Ankara, Turkey
burcuguldur@hacettepe.edu.tr
[2]CDM Smith Professor and Chair, Department of Civil and Environmental Engineering, Northeastern University, Boston, MA
JF.Hajjar@neu.edu

Abstract –

Recent advances in sensing technologies provide opportunity to utilize advanced data collection equipment such as laser scanners, high-resolution cameras, etc. for both short- and long-time monitoring of structures. Laser scanners, which are capable of collecting up-to two million points per second along with high-resolution images, are especially evolving rapidly and their usage for capturing and documenting the current condition of varying structural types is becoming increasingly feasible. The authors' previous research has focused on developing generalized algorithms for detecting surface damage on structures from captured laser scans and images. These algorithms are capable of automatically detecting surface damage on varying structural types as well as several construction materials since they use underlying surface geometry for performing damage detection. These damage types include concrete cracking, concrete spalling, steel delamination, steel section loss, bent members, and other misalignments. This paper investigates the use of supervised learning methods for determining whether a detected region represents an actual damage location. First, the input feature representations of the learning algorithms for each damage type are determined and the associated training sets, which are representative of real-world use of the algorithms, are prepared. Once the training and validation are completed, the learning algorithms are tested on the detected damage data, which is obtained by running the damage detection algorithms on both laser scans and texture-mapped images from a concrete test setup. Finally, the accuracy of these learning algorithms is investigated.

Keywords –

Terrestrial Laser Scanning; Surface Damage Detection; Damage Classification; Supervised Learning

1 Introduction

In the past two decades, researchers have developed several methodologies for both monitoring structures and detecting damage by using laser-scanning technology. The common methodologies used for structural monitoring include measuring high accuracy displacements, strains, pressures, or using a small number of points for computing related quantities or collecting visual inspection information [1]. Since the texture-mapped point cloud-capturing laser scanning is a recent technology, its usage for damage detection has not been investigated broadly. A couple of novel advances include the usage of point clouds to generate Building Information Model (BIM) of reinforced concrete walls and performed BIM–based earthquake damage assessment [2] and the utilization of LiDAR data and imagery for performing post-earthquake assessment, in order to determine the building damage degree [3].

Laser scanning technology provides a 3D representation of the entire structure under investigation. As has been discussed, with the current developments in this technology, it is possible to collect high-density point clouds that capture the surface properties accurately for structures. Authors' previous research focuses on the surface damage detection capabilities of the laser scanning technology that captures 3D point clouds with color information [4; 5; 6; 7]. The developed damage detection algorithms provide an adaptive system for damage detection, since the parameters used for the detection algorithms are computed automatically using the associated surface properties. The strategies developed for laser-based structural sensing and surface damage detection have been discussed extensively in [5].

For this paper, these developed damage detection algorithms are used to determine the damage locations on a concrete testing frame, and then, the detected surface damage is quantified. Since the main focus of this paper is to use supervised learning algorithms for damage classification, the detailed damage detection results obtained from the concrete testing frame are not included in the text, however, these results could be found in [5].

Once the damage detection and quantification are completed, the damage detection results obtained from the concrete testing frame are used for both training and validation of the learning algorithm, and finally, its accuracy is investigated.

The brief information on the methodologies used for surface damage detection and the detailed explanation on the use of supervised learning algorithms for damage classification are discussed in the following sections.

2 Laser-based Surface Damage Detection

Two main surface damage detection methodologies were developed in authors' previous research. The first methodology is a surface-normal based damage detection method that only uses the 3D coordinate information for locating rupture, spalling, delaminations. This method is later improved by using intensity values along with the 3D point information for locating small deformations such as cracks, corrosion.

The second methodology, on the other hand, is a graph-based damage detection method that is used for detecting alignment issues and points of discontinuity. This method is an extension of the graph-based object detection method that generates skeletons from cross-section cuts of a voxelized cluster, where a voxel represents a single sample or data point on a regularly spaced, three-dimensional grid, through extracting skeleton of an object in order to detect common structural members. The deviations from the predicted object alignments are used for extracting problematic locations on structures. Then, another method is introduced that converts cross-section voxel representation automatically into a polygon for computing the changes in the cross-section through area calculation and determining the total volume change on the investigated member.

Detected defects are automatically clustered and a mesh grid-based defect area and volume extraction method is developed in order to obtain quantifiable defect outputs for further investigation. For smaller defects such as cracks, an additional methodology is proposed for automated crack length and width extraction.

2.1.1 Surface Normal-based Damage Detection

The surface normal-based damage detection method relies on the modal properties of the detected surfaces and/or objects. The relative orientation of the estimated surface normal with respect to a reference normal is used to locate the defected areas on the surface of structures. The reference normal can be a surface normal computed via surface patches; the normal representing the skeleton of the detected object; or the normal vector between a reference point and the current query point.

The usage of only the normal variations on the surface would be sufficient for detecting large surface defects. However, another parameter should be introduced if certain defects with sizes close to the resolution of the scanner, such as cracks, are to be detected. Thus, the pixel information (intensity) obtained from texture-mapped point clouds is used for enhancing the detection capabilities for smaller defects.

In order to calculate both area and volume of a detected defect, a mesh-grid is fitted to the defect surface. Later, each part of the grid (for the area) and the corresponding rectangular prism (for the volume) are used for quantifying the detected damage. This damage quantification method is used to compute the area and volume associated with each detected defect.

For the concrete testing frame, the developed damage detection methods are used for both locating and quantifying concrete spalling. Several point cloud patches, which are extracted from the surface of the concrete testing frame, are processed. A set of representative dimensions are extracted from the detected damage areas, and these dimensions are compared with the corresponding hand-measurements in order to validate the efficiency of the detection algorithms.

Even though the mentioned quantification method is capable of both locating and quantifying relatively large defects, it is not suitable for recording required crack dimensions, which can be listed as length and thickness. Thus, a new methodology is developed for computing the necessary crack dimensions automatically.

2.1.2 Improvements for Crack and Spalling Detection

In most of the previous crack detection studies, which are predominantly image-based, some of the important parameters, such as camera-object distance, are not considered or are assumed to be constant. This prevents most of the current approaches from being used for crack quantification, since these methods are specifically developed for crack detection rather than quantification. For current approaches, it is required to maintain a constant focal length, resolution, or distance to the object in order to be able to extract crack dimensions ([8]; [9];

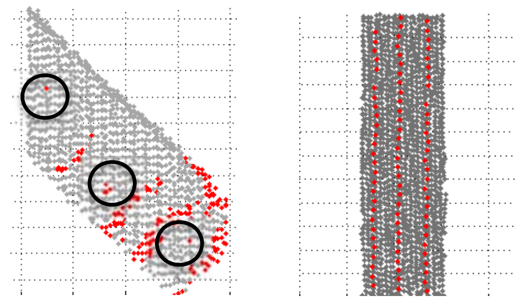
[10]; [11]; [12]; [13]). The damage detection method discussed in authors' previous research eliminates the requirement for prior knowledge on focal length, resolution, or distance to the investigated object, since all the required parameters for the defect detection are extracted from the point cloud automatically. However, it should be noted that the proposed method, similar to many other damage detection methods, especially crack detection methods, results in false positives along with actual defects.

The first step towards automated crack dimension extraction is to adjust the existing clustering algorithm, cluster tuning, which separates detected defect regions into individual defects, for cracks, and then quantify the dimensions of each separated point cluster with the proposed crack dimension extraction method. The crack dimensions are computed by defining a bounding box around each crack and performing a dimension extraction procedure as explained in [5].

The results obtained through defect detection and dimension extraction algorithms prove that these developed methods provide an opportunity to use laser scanning technology for detecting small defects, such as cracks and spalling regions, effectively. However, it should be noted that since the defect detection is achieved through investigating the local variations at a point of interest on the surface, the developed detection algorithms are sensitive to surface impurities. These impurities and/or any reoccurring patterns result in false positives when the damage detection algorithms are executed. Thus, the obtained results include several falsely detected clusters, which do not represent an actual defect. In order to avoid detecting these false positives, to improve the accuracy of the developed algorithms and to perform automated classification, a neural network classifier is introduced. The details of this classification strategy and the classifier's overall performance on the concrete testing frame are discussed in the following section.

3 Damage Classification with Supervised Learning

The detected false positives for damage results could be an issue due to a repeated pattern, any surface impurity, and etc.; essentially, any surface variation that causes the surface normal to deviate from the computed reference normal and/or results in significant variations in local intensity values. Two significant example cases that are encountered for this dataset are shown in Figure 1(a) and (b). In Figure 1(a), the locations of small surface holes are shown with black circles, and a repetitive pattern for intensity variation, resulted from inefficient texture-mapping, is represented in Figure 1(b). It is required to eliminate the false positives from the detected damage clusters in order to improve the overall efficiency of the proposed defect detection algorithm. To perform this, a trained artificial neural network classifier is used to differentiate the real defects (cracks and/or small spalling regions) from false positives.



(a) (b)

Figure 1. Sample false positive patterns: (a) small holes existing on the surface and (b) lines with significant intensity variation

In [14], an image-based study, which uses 2D images to create 3D surface representations, was performed, in order to eliminate the dependency on the previously listed parameters: constant focal length, resolution, or distance to the object. The first step of this study is segmentation, which is used for isolating the patterns that can be potentially classified as defects. The useful information about the scene objects is extracted by using morphological image processing and then, structuring elements are used to complete the segmentation. Segmentation is followed by feature extraction and finally, the study is completed by classification.

In this paper, a similar methodology for classification is followed. First, the defects are segmented as objects by using the developed clustering methods. Second, a feature set that stores quantitative information on each defect cluster is formed. Finally, this feature set is used to train, validate and test the neural network classifier.

3.1 Feature Extraction

A feature is defined as a set of finite values that represents the quantitative attributes or properties of any segmented object; in our case, clusters. It is crucial to generate an appropriate feature set that includes all the important characteristics that helps identifying similar patterns. In this paper, a feature set, which is similar to the one proposed in [14], is used, with two additions representing the properties of intensity distribution, since both application have a similar nature. The feature set defined in [14] is extracted by using only the 2D properties of the segmented objects; however, since the detected defect clusters represent a 3D dataset, for this research, some of the described features are extracted by using 3D information.

The features that are included in the feature set can be listed as eccentricity, area of the projected cluster divided by the area of the bounding ellipse, solidity, absolute value of the correlation coefficient, compactness, and mean and standard deviation of intensity values. The definitions of each item in the listed feature set could be found in both [14] and [5].

These features listed above are computed for each defect cluster, and the values are stored in a feature matrix. This feature matrix is then used for training, validating and testing the neural network classifier.

3.2 Classification

There are several classifier options that can be used for this application. Some of the possible classifiers can be listed as Bayes classifier [15], k-nearest neighborhood classifier [16], multilayer feed-forward artificial neural networks [17], support vector machines [18], and etc. However, in most of the recent studies performed for crack detection, where several classifiers are compared, the accuracy of the results obtained by using neural network classifiers is shown to be higher than the other listed methods ([8]; [9]; [14]; [19]). Thus, for this research, the neural network classifier is selected to be used for classification.

An artificial neural network, which is composed of processing elements that are interconnected via synaptic or weighted connections, is a parallel processing optimization system. In a neural network, weighted interconnections are used to process inputs received at a processing element in a layer, and then the resulting outputs are transmitted to the following set of processing elements. For nonparametric pattern classification, feed-forward neural networks, which are trained by using a back-propagation algorithm, are the most commonly used neural approaches [20]. This type of neural network can be composed of three or more layers of processing elements: the input layer, hidden layer/layers, and output layer. Through weighted interconnections, each processing element in a layer is connected with all processing elements in the preceding and following layers. For pattern classification applications, the input layer consists of a set of feature vectors; thus, the size of the input layer is always equal to the number of features. However, it should be noted that the number of processing units in hidden layers is completely dependent on the complexity of the pattern recognition problem. The number of neurons in the hidden layer is generally empirically determined by trial and error [21]. The number of layers and the number of neurons in the hidden layers of neural networks have to be chosen carefully in order to obtain good classification results. Since the computed feature matrix is composed of similar entities, in this research, the neural network configuration given in [14] is used. The classifier used for this application is a three-layer feed forward neural network, which has 2 output neurons and 10 neurons in the hidden layer.

4 Results

The developed damage detection algorithms are used to process 106 randomly selected surface patches from the concrete testing frame. As a result, 201 candidate defect clusters are detected and separated for further analysis. These defect clusters are plotted, and the actual defects and the false positives are manually separated for every patch.

The generated feature set is composed of 201 damage and non-damage feature vectors. Out of 201 feature vectors, 74 vectors represent the properties of actual defects, whereas 127 vectors are composed of features extracted from false positives. For training, % 70 of the complete feature set is used; 15% is used for validation and finally, 15% is used for testing.

The performance of the selected classifier is shown by using four items: accuracy, precision, sensitivity, and specificity. Accuracy shows the proportion of true classifications in the test set (15% of the entire feature set). Precision is defined as the proportion of the true positive classifications against all positive classifications. Sensitivity is the proportion of actual positives that were correctly classified, and specificity is the proportion of negatives that were correctly classified. The results are shown in Table 1.

Figure 2(a) shows a portion of the concrete testing frame, and Figure 2(b) presents the post-classification damage detection results on the corresponding region of the point cloud.

Table 1 Performance results for the neural network classifier

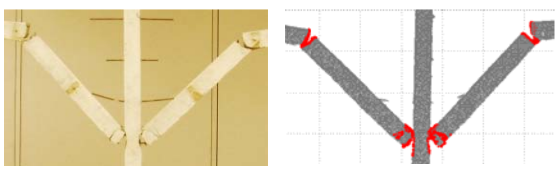
	Percentage (%)
Accuracy	93.51
Precision	94.45
Sensitivity	95.89
Specificity	90.38

5 Conclusions

The results show that the accuracy of the neural network classification is high for this specific application.

However, it should be noted that the detected defect clusters sometimes fail to represent all the surface damage that is present on an investigated patch. Some of the defects may not be detected depending on the properties of the point cloud. For these cases, even though the accuracy of the classifier is high for the generated feature set, the obtained results may not represent the overall efficiency of the developed damage detection algorithm. At the same time, it should be mentioned that the performance of the trained neural network on another structure cannot be estimated from the obtained results; however, in literature, there are several examples that show the accuracy of a trained neural network reduces when the classifier is tested on different structures.

Further study on the subject, which includes the application of the learning algorithms on larger datasets with varying surface properties, needs to be performed, in order to build a comprehensive understanding on the use of supervised learning algorithms for laser scanning-based damage detection and classification applications.



(a) (b)

Figure 2. (a) Image of a portion of the concrete testing frame, and (b) defect detection results shown on corresponding portion of the point cloud

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. IIS-1328816 and Northeastern University. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] Chang, P. C., Flatau, A. and Liu, S. (2003). “Review Paper: Health Monitoring of Civil Infrastructure.” *Structural Health Monitoring*, 2(3), 257-267.

[2] Anil, E. B., Akinci, B., Kurc, O. and Garrett, J. H. (2015). “Building-Information-Modeling–Based Earthquake Damage Assessment for Reinforced Concrete Walls.” *Journal of Computing in Civil Engineering*, 04015076.

[3] Li, M., Cheng, L., Gong, J., Liu, Y., Chen, Z., Li, F., Chen, G., Chen, D. and Song, X. (2008). “Post-earthquake Assessment of Building Damage Degree using LiDAR Data and Imagery.” *Science in China Series E: Technological Sciences*, 51(2), 133-143.

[4] Guldur, B. and Hajjar, J. F. (2013). “Laser-based Automatic Cross-Sectional Change Detection for Steel Frames.” *Proceedings of the 9th International Workshop on Structural Health Monitoring 2013*, Stanford University, Stanford, CA.

[5] Guldur, B. (2014). “Laser-based Structural Sensing and Surface Damage Detection.” Ph.D. Dissertation, Northeastern University, Boston, Massachusetts.

[6] Guldur, B. and Hajjar, J. F. (2014). “Damage Detection on Structures Using Texture Mapped Laser Point Clouds.” *Proceedings of the ASCE/SEI Structures Congress 2014*, Boston, MA.

[7] Guldur, B. and Hajjar, J. F. (2014). “Automated Damage Assessment from 3-D Laser Scans.” *Proceedings of the Istanbul Bridge Conference*, Istanbul, Turkey.

[8] Kaseko, M. S. and Ritchie, S. G. (1993). “A Neural Network-based Methodology for Pavement Crack Detection and Classification.” *Transportation Research Part C: Emerging Technologies*, 1(4), 275-291.

[9] Cheng, H., Chen, J.-R., Glazier, C. and Hu, Y. (1999). “Novel Approach to Pavement Cracking Detection based on Fuzzy Set Theory.” *Journal of Computing in Civil Engineering*, 13(4), 270-280.

[10] Abdel-Qader, I., Abudayyeh, O. and Kelly, M. E. (2003). “Analysis of Edge-detection Techniques for Crack Identification in Bridges.” *Journal of Computing in Civil Engineering*, 17(4), 255-263.

[11] Abdel-Qader, I., Pashaie-Rad, S., Abudayyeh, O. and Yehia, S. (2006). “PCA-based Algorithm for Unsupervised Bridge Crack Detection.” *Advances in Engineering Software*, 37(12), 771-778.

[12] Choudhary, G. K. and Dey, S. (2012). “Crack Detection in Concrete Surfaces using Image Processing, Fuzzy Logic, and Neural Networks.” *IEEE Fifth International Conference on Advanced Computational Intelligence*, Nanjing, Jiangsu, China.

[13] Rabah, M., Elhattab, A. and Fayad, A. (2013). “Automatic Concrete Cracks Detection and Mapping of Terrestrial Laser Scan Data.” *NRIAG Journal of Astronomy and Geophysics*, 2(2), 250-255.

[14] Jahanshahi, M. R. and Masri, S. F. (2012). “Adaptive Vision-based Crack Detection using 3D Scene Reconstruction for Condition Assessment of Structures.” *Automation in Construction*, 22, 567-576.

[15] Rish, I. (2001). "An Empirical Study of the Naive Bayes Classifier." *IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence*, Seattle, WA.

[16] Keller, J. M., Gray, M. R. and Givens, J. A. (1985). "A Fuzzy k-nearest Neighbor Algorithm." *IEEE Transactions on Systems, Man and Cybernetics*, 4, 580-585.

[17] Hornik, K., Stinchcombe, M. and White, H. (1989). "Multilayer Feed Forward Networks are Universal Approximators." *Neural Networks*, 2(5), 359-366.

[18] Hearst, M. A., Dumais, S., Osman, E., Platt, J. and Scholkopf, B. (1998). "Support Vector Machines." *Intelligent Systems and their Applications, IEEE*, 13(4), 18-28.

[19] Adhikari, R., Moselhi, O. and Bagchi, A. (2014). "Image-based Retrieval of Concrete Crack Properties for Bridge Inspection." *Automation in construction*, 39, 180-194.

[20] Rumelhart, D. E., McClelland, J. L. and Group, P. R. (1995). *Parallel Distributed Processing*, Cambridge, MA, MIT Press.

[21] Kaseko, M. S., Lo, Z.-P. and Ritchie, S. G. (1994). "Comparison of Traditional and Neural Classifiers for Pavement-crack Detection." *Journal of Transportation Engineering*, 120(4), 552-569.