A CAPSNETS APPROACH TO PAVEMENT CRACK DETECTION USING MOBILE LASER SCANNNING POINT CLOUDS

Wenxuan Zhu[1], Weikan Tan[1], Lingfei Ma[2*], Dedong Zhang[3], Jonathan Li[1, 2, 3], Michael A. Chapman[4]

[1] Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON N2L 3G1, Canada - w78zhu@uwaterloo.ca; weikai.tan@uwaterloo.ca; junli@uwaterloo.ca
[2] Engineering Research Center of State Financial Security, Ministry of Education, Central University of Finance and Economics, Beijing, 102206, China – l53ma@cufe.edu.cn
[3] Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada - dedong.zhang@uwaterloo.ca;
[4] Department of Civil Engineering, Ryerson University, Toronto, ON M5B 2K3, Canada - mchapman@ryerson.ca.

Commission I, WG I/2

KEY WORDS: CapsNets, Crack Detection, Pavement Inspection, Mobile Laser Scanning, Point Cloud

ABSTRACT:

Routine pavement inspection is crucial to keep roads safe and reduce traffic accidents. However, traditional practices in pavement inspection are labour-intensive and time-consuming. Mobile laser scanning (MLS) has proven a rapid way for collecting a large number of highly dense point clouds covering roadway surfaces. Handling a huge amount of unstructured point clouds is still a very challenging task. In this paper, we propose an effective approach for pavement crack detection using MLS point clouds. Road surface points are first converted into intensity images to improve processing efficiency. Then, a Capsule Neural Network (CapsNet) is developed to classify the road points for pavement crack detection. Quantitative evaluation results showed that our method achieved the recall, precision, and F_1-score of 95.3%, 81.1%, and 88.2% in the testing scene, respectively, which demonstrated the proposed CapsNet framework can accurately and robustly detect pavement cracks in complex urban road environments.

1. INTRODUCTION

1.1 Motivation

Pavement cracks are common damages on pavement surfaces, which are signed for potential damages in the supporting structures (Lee, 1991). Road surface defects may cause severe troubles in traffic, such as congestion, delay, and even safety problems. Ragnoli et al. (2018) indicated that there would be an increasing demand for Pavement Maintenance and Rehabilitation programs worldwide. Road cracks tend to deteriorate due to environmental factors (Chen et al., 2004). If the cracks are not sealed in time, water infiltration into the lower layer, especially during the period of snowmelt freezing-thawing mix, will exacerbate the damage of the road and lead to the formation of network cracks. Therefore, it is important to prevent and repair early cracks in the pavement. However, traditional road crack detection usually relies on human inspection, limiting the accuracy and efficiency of the measurement (Li et al., 2019). Most of the common practice in the road is usually time-consuming, dangerous, labour-intensive, and subjective. Thus, it is a trend to replace traditional crack detection methods with automated or semi-automated ones. Semi-automated methods combine human intervention and machine, while automated methods require minimal human assistance. Automated and semi-automated technologies make it possible to develop real-time pavement distress detection.

Mobile laser scanning (MLS) provides high-density data by close-range acquisition, which ensures even the smallest of features are captured in the resulting point clouds. The densities of such point clouds vary significantly depending on several factors, including the driving speed during acquisition, distances from the laser beams to the surfaces reflecting the laser's energy and laser repetition rate. It is common that densities are measured in the hundreds or even thousands of points per square meter. Moreover, comparing with traditional image data, MLS data can provide highly accurate spatial information (Li et al., 2020). The laser scanners used in the high-end MLS systems are typically accurate to a few millimeters. Positioning accuracy of 1 to 2 centimeters is possible with careful planning, quality hardware, favorable GPS conditions, and supplemental ground control. Additionally, MLS systems enable the mobile data collecting of the roads and constructions and provide affordable 3D databases for GIS analysis (Li et al., 2019). To be specific in the crack detection, instead of the color differences in the RGB images, the intensity differences of the generated 2D images of MLS present the crack clearly.

1.2 Objectives of the study

This paper aims to propose an efficient deep learning-based framework to provide pavement crack feature detection, which can be used to maintain and manage road construction. The data used in this paper is 3D point cloud data obtained from an MLS system. The main objectives of this study are as follows: (1) Applying Capsule Neural Network (CapsNet) to pavement inspection and (2) Analyzing the advantages and disadvantages of applying the CapsNet to pavement crack detection.

1.3 Related Work

1.3.1 Rule-Based Methods: Rule-based crack detection was one of the earliest functions presented for the semi-automated pavement crack (Mei et al., 2020). Tsao et al. (1994) proposed a rule-based system containing facts and variable rules created by prominent features of different types of distress. In addition, the major processes were gathering information on the input image and then deciding the efficient pattern. The results achieved accuracy at about 85% to 90%. Moreover, Amhaz et al.

* Corresponding Author

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLIII-B1-2021-39-2021 | © Author(s) 2021. CC BY 4.0 License.

(2016) stated a rule based on the minimal path selection algorithms with a redefined artifact filtering step to estimate the thickness of the crack pattern. Gavilán et al. (2011) proposed an approach combining a series of image processing techniques.

In this approach, the image was preprocessed to enhance the linear features, and confusing area, like joints on pavements, will be eliminated. Moreover, a seed-based approach combining multiple directional non-minimum suppression with symmetry check was proposed. In general, rule-based crack detection was easy to verify the pavement crack, as it did not require an annotation and training process (Mei et al., 2020). However, in this kind of methods, most of the features were created artificially in some original datasets, in which not all the variation in real-life images could be considered, especially illumination changes or irregular shape of cracks. Thus, these rule-based methods could not work well in changeable situations.

1.3.2 Learning-based methods: As stated above, the rule-based methods for pavement crack could be improved into more efficient and accurate ways. The deep learning-based algorithms were studied by many researchers in the last decade. Tensor voting included two major steps, representing data using tensor calculus, and data combining by nonlinear voting, including sparse tensor voting and dense tensor voting (Guan et al., 2014). In addition, the iterative tensor voting (ITV) contained three steps, preprocessing of the MLS data, GRF image generation and ITV-based crack detection, in which the third step included the core algorithm of the whole process. The ITV method achieved much more accurate results. However, this method required intensive computation (Guan et al., 2014). The major processed of tensor voting and ITV were shown.

CNN had become state-of-the-art for various image analysis tasks (Mei et al., 2020). A basic neural network normally consisted of three layers: an input layer, hidden layers, and an output layer (LeCun et al., 1999). In a typical CNN model, the hidden layer constituted a group of convolutional layers. The convolution layer would be multiplied with other layers (LeCun et al., 1999). The activation function was commonly a RELU layer following additional convolutions such as pooling layers, fully connected layers and normalization layers. These layers referred to hidden layers because the activation function and final convolution masked inputs and outputs.

A geometric graph CNN was based on the MLS data established by Li et al. (2019). To learn the major features from the MLS point sets, they combined the Taylor Gaussian mixture model network. This algorithm could reduce the computation cost with guaranteed segmentation performance. However, the multiobject connected area labeling was limited because of the limited receptive field for TGConv. Furthermore, real-time road crack mapping was introduced based on MLS technology (Naddaf-Sh et al., 2019), which combined the CNN and Bayesian optimization algorithm to improve the precision and decrease processing time. As a result, they achieved over 90% accuracy with real-time images and videos.

In conclusion, there were two main types of data for crack detection, i.e., images and MLS data. The rule-based and deep learning-based methods could be used for-based crack detection (Mei et al., 2020). Moreover, for MLS data-based crack detection, there were two main methods, 2D georeferenced feature (GRF) image-driven detection and 3D point-driven detection functions (Ma et al., 2018).

2. METHOD

This experiment used MLS point clouds and converted them into intensity images. The generated images were then divided into training, validation, and testing datasets. Finally, the CapsNet was proposed for pavement crack detection. The CapsNet model has three stages, which are the Rectified Linear Unit (ReLU) convolution, primary capsules, and convolution capsules. Additionally, the evaluation metrics used in the experiment are recall, precision, and F_1-score.

2.1 Data

The MLS data, which was collected in April 2012 by a RIEGL VMX-450 MLS system, was smoothly integrated with two RIEGL VQ-450 scanners. The scanners' laser pulse repetition rates are up to 550 kHz (Guan et al., 2014). The average speed of the traveling vehicle was 50 km/h. The two laser scanners were symmetrically settled on the left and right sides, which orient the rear of the vehicle at a heading angle of approximately 145 ° in the "Butterfly" configuration pattern, which can also be called the "X" configuration pattern. With this pattern, RIEGL VQ-450 can scan in a 360° circle owing to the motorized mirror scanning mechanism.

According to the RIEGL website (2013), the system can provide a measurement rate of 1.1 million pts/sec and a scan frequency of 400 lines/sec. The density sharply drops perpendicular to the travel lines, which means the closer to the vehicle trajectory will lead to a higher point density level. The average point density on the road is about 3,300 points/m^2. The dataset with 8.4 million points in the length of 105 m is selected from the whole survey. In addition, it covers a two-lane cement-paved road segment of about 3,105 m. All the MLS data used in this experiment has been pre-processed with registration operations. Figure 1 shows the generated georeferenced intensity image of the road and Figure 2 shows the labeled pavement cracks highlighted in yellow. A comparison of part of the georeferenced intensity image and classified image are shown in Figure 3.

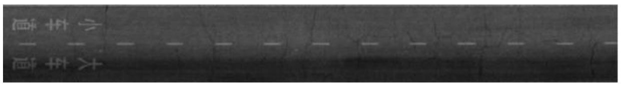

Figure 1. Georeferenced intensity image of pavement surface.

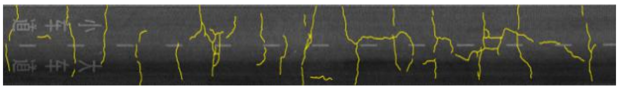

Figure 2. Labeled pavement cracks (yellow).

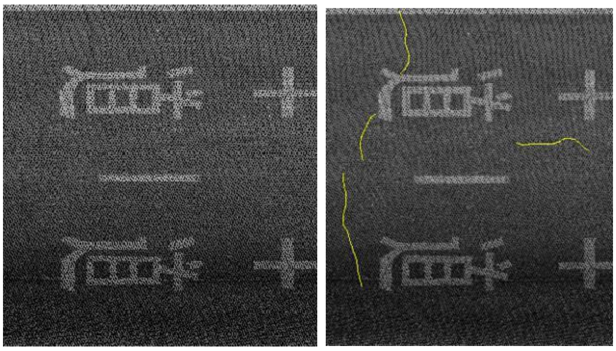

Figure 3. Compare of georeferenced intensity image (left) and classified images (right).

2.2 Pavement Crack Detection

2.2.1 Overview of workflow: A basic neural network usually consists of three layers: an input layer, hidden layers, and an output layer (Hinton, 2017). Specifically, in the CNN algorithms proposed for pavement crack, the input layer's input data are tensors, while the hidden layer contains convolutional layers, RELU layers, and pooling layers. In the CNN algorithms, the most important part is convolution, in which the input tensor data and convolution kernel are multiplied to obtain the improved results. In this paper, the proposed CapsNet uses the capsules to replace the simple layers in the traditional neuron. It mainly includes two stages, linear combination and dynamic routing (Jiménez-Sánchez et al., 2018). Furthermore, the CapsNet builds the multi-dimensional squashing function and creates tensors by grouping multiple feature channels. CapsNet includes dynamic routing mechanisms (Hinton et al., 2018).

Figure 4 shows the processing workflow. The proposed method can be divided into three steps, which are road segmentation using intensity image generation, data preprocessing, and CapsNet model. The CapsNet contains four parts, they are the convolution layer, the primary capsule layer, the convolution capsule layers, and the full capsule layer.

2.2.2 Road Segmentation and Intensity Image Generation: As stated above, the collected MLS data contain a large volume of 3D points with which the entire road scene is covered. In order to narrow the searching region and improve efficiency, we only focus on the processing of road surface points. The cracks are on the road surfaces, and these road surfaces can be regarded as planes. Therefore, we projected MLS point clouds onto the road surface to generate georeferenced intensity images without height information. The computational efficiency can be improved effectively.

The curb-line-based road segmentation method was adopted to separate road surface points from the entire point cloud data (Guan et al., 2014). Instead of processing the discrete, unordered road surface points in 3D space, we rasterized them into a 2D georeferenced intensity image using the inverse distance weighted (IDW) interpolation method. In this experiment, road surface points were vertically partitioned into a series of grids with a specific spatial resolution. The spatial resolution was determined according to the point cloud density. Then, the grid points were interpolated into a single pixel. The gray value was determined by the distances to the grid center and intensities of these points. If a grid contains no points, the associated pixel value is set to be zero.

To divide and classify the data into suitable sizes, the cracks were marked in yellow, as shown in Figure 2. Then, it was divided into pieces using the same method as Figure 1 is divided. With the labeled dataset, the unlabeled one can be easily classified. The numbers of the images with a crack after clipping can be recorded based on the labeled one's information, and they will be selected from the labeled data set. The labelled dataset in this study is called Pavement Crack (PC) data.

The training, validation, and testing data were fed into the model in the Incremental Design Exchange (IDX) data format. The format of the PC dataset is similar to the MNIST dataset, which is a handwritten digit classification dataset used for machine-learning training. The total image size of the generated intensity image is 377 × 3770. Additionally, the segmented pieces were labeled as pavements and cracks. Furthermore, 1,000 pieces of segmented pavement were selected as the training data, while 316 pieces were selected as validation data, and 176 pieces were used as testing data. The training, testing, and validation data were all randomly selected. Moreover, the data ratio is 67%, 21%, 12% of the total dataset.

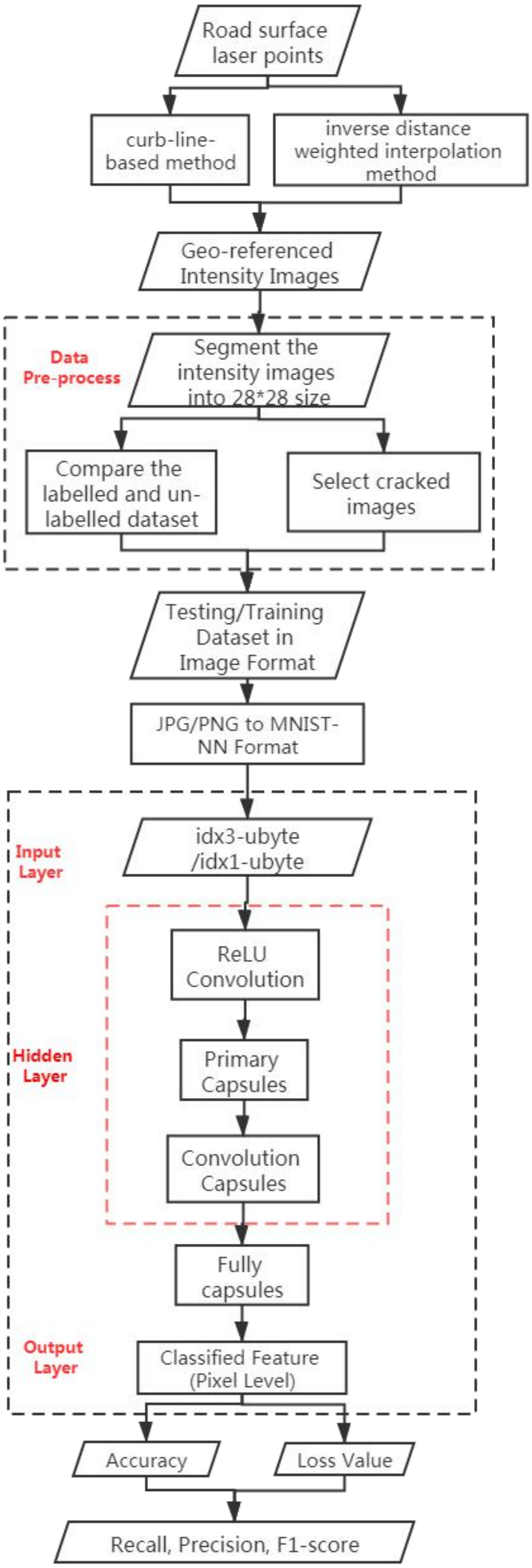
Road surface laser points
curb-line-based method
inverse distance weighted interpolation method
Geo-referenced Intensity Images
Data Pre-process
Segment the intensity images into 28*28 size
Compare the labelled and un-labelled dataset
Select cracked images
Testing/Training Dataset in Image Format
JPG/PNG to MNIST-NN Format
Input Layer
idx3-ubyte /idx1-ubyte
Hidden Layer
ReLU Convolution
Primary Capsules
Convolution Capsules
Fully capsules
Output Layer
Classified Feature (Pixel Level)
Accuracy
Loss Value
Recall, Precision, F1-score

Figure 4. Flowchart of the proposed method.

2.2.3 Proposed Model: The first layer, called ReLU Conv1, is a convolution layer, by which the simple features can be detected and inputted to the primary capsules. In addition, there are 256 kernels, each of which is in the size of 9*9 (Sabour, Frosst & Hinton, 2017). With this convolution, the parameter is in the size of K2*K2*256 (Sabour et al., 2017). Moreover, the second layer, called primary capsules, combines the convolutional layer features (Yu et al., 2019). Each of the primary capsules converts the data with eight convolutional kernels in the size of K3*K3*8. Furthermore, with the convolution capsules and the full capsule, the 8-D vector is converted into 16-dimensional.

In the capsule layer, the transformation is shown and calculated by the following equation:

$$S_j = \sum_i C_{ij} \hat{u}_{j|i} \tag{1}$$

In Eq. (1), S_j shows the entire input to the *j-th* capsule. In addition, the C_{ij} is the weight for the connection of the *i-th* and the *j-th* capsules, and $\hat{u}_{j|i}$ is the transformed input to the *j-th* capsule. Moreover, $\hat{u}_{j|i}$ can be calculated by:

$$\hat{u}_{j|i} = W_{ij} u_i \tag{2}$$

As is mentioned above, $\hat{u}_{j|i}$ is the input of the *j-th* capsule. In addition, W_{ij} is the weights of the transformation from the *i-th* to *j-th* capsule, and u_i is the output of the *i-th* capsule. These transformations allow learning the whole relationship, instead of detecting independent features by filtering at different scales portions of the image (Hinton et al., 2011).

In routing by agreement, the outputs from one capsule are routed to capsules in the next layer according to the child capsule's ability to predict the parent capsule's outputs (Hinton et al., 2011). The squashing function, which is a non-linearity function, is combined by an additional scaling and a unit scaling (Ma et al., 2020):

$$u_j = \frac{\|S_j\|^2}{1 + \|S_j\|^2} \frac{S_j}{\|S_j\|^2} \tag{3}$$

where u_j is the vector output of *j-th* capsule, and S_j is its input.

2.2.4 Performance Assessment: In this experiment, the performance assessment on training and validation date will be based on the loss equation. During the training, the loss function is:

$$L_k = T_k \max(0, m^+ - \|v_k\|)^2 + \alpha(1 - T_k) \max(0, \|v_k\| - m^-)^2 \tag{4}$$

In the loss equation, k means the number of classes of the digit, and α is set to 0.5 for the balance of loss. Additionally, only if m^+=0.9 and m^-=0.1, it meets T_k=1 (Hinton et al., 2018).

Moreover, using the testing data to do the evaluation is also part of the performance assessments. The main difference between the evaluation and the training is the processing dataset, as the evaluation is based on the testing data while the training is based on the other two datasets. Additionally, both of them use the capsule steps as provided above during the test.

Furthermore, the accuracy is also assessed by comparing the extracted road crack markings with the manually ground-truth. The results are quantitatively evaluating by the following three measures: recall, precision, and F_1 score. Recall describes if the pavement crack markings are completely extracted, while precision indicates the percentage of the valid markings. The recall and precision are defined as (Guan et al., 2014):

$$\begin{cases} recall = Cp/Rf \\ precision = Cp/Ep \end{cases} \tag{5}$$

where *Cp* means the number of pixels belonging to the actual pavement cracks, *Rf* shows the amount of ground-truth collected by the manual interpretation, and *Ep* represents the number of pixels extracted by the proposed algorithm. F_1-score shows an overall score, which is defined as (Guan et al., 2014):

$$F_1\text{-score} = 2 \times \frac{(recall - precision)}{(recall + precision)} \tag{6}$$

3. RESULTS AND DISCUSSION

3.1 Experiment Results

Figure 5 shows the overall accurate of the classification . To compare and visualize the result, several scatter diagrams with trend lines are created.

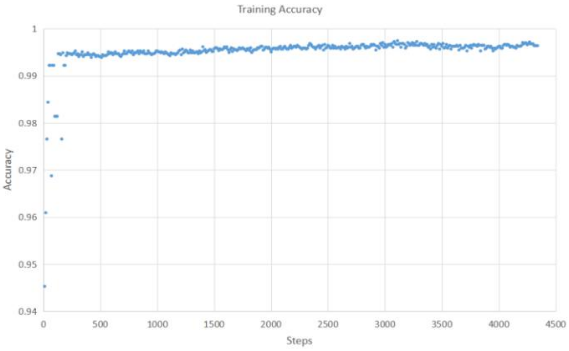

Figure 5. Training accuracy of the proposed model.

Figures 5 and 6 show the accuracy of the corrected experiment and the iteration is set as 3. To clearly present the trends of accuracy, the accuracy of the first step is not shown in Figures 5 and 6, as the accuracy of them close to 0. They both present upward trends during the training. In Figure 5, the initial accuracy is relatively dispersed and unstable, but the accuracy gradually increases with the increase of training. In Figure 6, the validation accuracy is increasing stably between 0.980 and 0.995. In addition, when the epoch is set to 50 and the iteration is set to 3, it spends about two hours training the dataset.

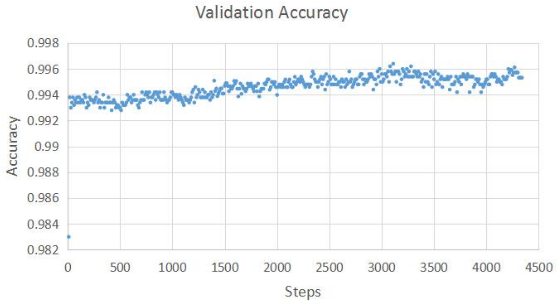

Figure 6. Validation accuracy of the proposed model.

3.2 Loss Value

Figure 7 presents the best loss value trend, in which the iteration is set as 3 rather than 1 or 2. It shows a relatively concentrated of data. Figures 7 and 8 show the loss value of the experiment and the trend during the training. The loss function in this experiment is used to optimize the CapsNet. The loss accuracy provides the sum of errors made for each example in training or validation sets. If the model's prediction is perfect, the loss will be zero, while the greater value indicates a worse prediction.

Figure 8 shows a trend line graph that presents the comparison of the loss value with different iterations. All three lines show the overall decreasing trend. The orange line is the trend line of loss when the iteration is equal to three. The green line shows the loss value with three times iteration, while the blue one is with two times iterations. The loss values of these three experiments are in the trend from 0.0005 to 0.0003. Comparing these three lines, the orange line keeps lower than the other two.

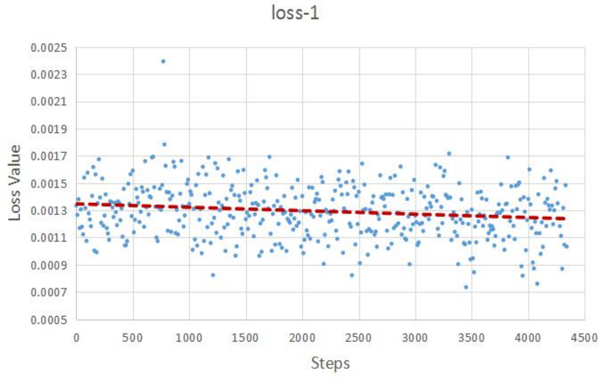

Figure 7. Best loss value of the proposed model.

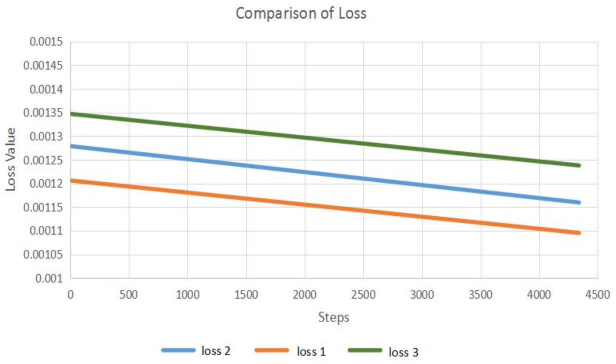

Figure 8. Comparison of loss.

3.3 Training,Validation and Test Accuracy

The training and validation accuracy is the accuracy of a model that it was constructed on. Figures 5, 6, and 9 illustrate the training and validation accuracy. Figure 9, a sample training accuracy of the overfitting model, shows a trend from 0 to 100%, as after step 530, the accuracy is 100%. As the training accuracy meets 100% in training, this shows that the model is overfitting. Overfitting refers to a model that has learned the training dataset too well, including the statistical noise or random fluctuations in the training dataset. However, the problem with overfitting is that the more specialized the model becomes to training data, the less well it can generalize to new data, resulting in an increasing generalization error. This increase in generalization error can be measured by the performance of the model on the validation dataset. In conclusion, the three main reasons for the model's overfitting are the high complexity of the model, the insufficient training data, and the big data noise. Specifically, in this experiment, the overfitting is caused by the inaccuracy of the training data. After re-selecting the crack datasets, the overfitting problem can be solved.

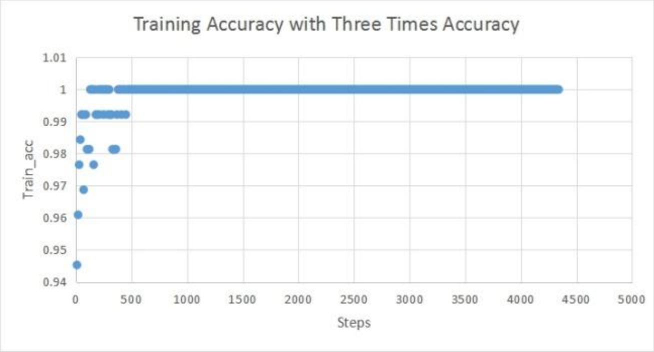

Figure 9. Train Accuracy of the overfitting model.

3.4 Accuracy Assessment

Table 1. Results of performance assessment.

Iteration Times	Recall (%)	Precision (%)	F_1 score (%)
1	88.4	80.3	81.4
2	93.2	80.9	87.7
3	95.3	81.1	88.2

The performance of our proposed method was evaluated by the following three measures: recall, precision, and F_1-score. As is shown in Table 1, the recall, precision, and F_1-score values are all greater than 80.0%. Moreover, all of them increase with the increasing of the iterations. The highest values of the accuracy assessment in this experiment are 95.3%, 81.1%, and 88.2%.

4. CONCLUSIONS AND RECOMMENDATIONS

In this paper, we proposed a CapsNet-based model for pavement crack segmentation for applications of the pavement management system. The proposed methods contain three main steps. They are road segmentation using intensity image generation, data preprocessing, and crack detection using the proposed CapsNet model. Then, the accuracy assessment is presented based on recall, precision, and F_1 -score, which achieved an average score of 95.3%, 81.1%, and 88.2%, respectively. Moreover, the comparison result shows that the higher iteration value can provide a better proposed model, but consume more computational costs and spend more time. The experimental results show an overall high accuracy through the testing process.

In conclusion, the CapsNet is efficient to encode inherent features from MLS point clouds, contributing to effective and accurate pavement crack detection, especially in urban road scenarios. This experiment can be improved by optimizing the methods of inputting variable types of data, especially general remote sensing types. For example, if the BMP format data can be extracted and tested directly, it will be possible to achieve real-time crack detection.

REFERENCES

Alavi, A. H., Hasni, H., Lajnef, N., & Chatti, K., 2016. Continuous health monitoring of pavement systems using smart sensing technology. *Construction and Building Materials*, 114, 719-736.

Amhaz, R., Chambon, S., Idier, J., & Baltazart, V., 2016. Automatic crack detection on two-dimensional pavement images: An algorithm based on minimal path selection. *IEEE Transactions on Intelligent Transportation Systems*, 17(10), 2718-2729.

Gavilán, M., Balcones, D., Marcos, O., Llorca, D. F., Sotelo, M. A., Parra, I., . . . Amírola, A. 2011. Adaptive road crack detection system by pavement classification. *Sensors*, 11(10), 9628-9657.

Girshick, R., Donahue, J., Darrell, T., & Malik, J. 2016. Region-based convolutional networks for accurate object detection and segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(1), 142-158.

Guan, H., Li, J., Yu, Y., Wang, C., Chapman, M., & Yang, B. 2014. Using mobile laser scanning data for automated extraction of road markings. *ISPRS Journal of Photogrammetry and Remote Sensing*, 87, 93-107.

Guan, H., Li, J., Yu, Y., Chapman, M., Wang, H., Wang, C., & Zhai, R. 2015. Iterative tensor voting for pavement crack extraction using mobile laser scanning data. *IEEE Transactions on Geoscience and Remote Sensing*, 53(3), 1527-1537.

Hinton, G. E., Krizhevsky, A., & Wang, S. D. 2011. Transforming auto-encoders. *In International Conference on Artificial Neural Networks*, pp. 44-51.

Hinton, G. E., Sabour, S., & Frosst, N. 2018. Matrix capsules with EM routing. *In 6th International Conference on Learning Representations (ICLR)*, 1-15.

Jiménez-Sánchez, A., Albarqouni, S., & Mateus, D. 2018. Capsule networks against medical imaging data challenges. *In Intravascular Imaging and Computer Assisted Stenting and Large-Scale Annotation of Biomedical Data and Expert Label Synthesis,* pp. 150-160.

Kussul, N., Lavreniuk, M., Skakun, S., & Shelestov, A. 2017. Deep learning classification of land cover and crop types using remote sensing data. *IEEE Geoscience and Remote Sensing Letters*, 14(5), 778-782.

Li, Y., Ma, L., Zhong, Z., Liu, F., Chapman, M. A., Cao, D., & Li, J. 2020. Deep learning for LiDAR point clouds in autonomous driving: a review. *IEEE Transactions on Neural Networks and Learning Systems*. doi: 10.1109/TNNLS.2020.3015992.

Li, Y., Ma, L., Zhong, Z., Cao, D., & Li, J. 2020. TGNet: Geometric graph CNN on 3-D point cloud segmentation. *IEEE Transactions on Geoscience and Remote Sensing*, 58(5), 3588-3600.

Lee, H., 1991. Standardization of distress measurements for the network-level pavement management system. *Pavement Management Implementation*. ASTM International. doi: 10.1520/STP17817S.

Ma, L., Li, Y., Li, J., Wang, C., Wang, R., & Chapman, M. 2018. Mobile laser scanned point-clouds for road object detection and extraction: A review. *Remote Sensing*, 10(10), 1531.

Ma, L., Li, Y., Li, J., Yu, Y., Junior, J. M., Gonçalves, W. N., & Chapman, M. A. 2020. Capsule-based networks for road marking extraction and classification from mobile LiDAR point clouds. *IEEE Transactions on Intelligent Transportation Systems*, 22(4), 1981-1995.

McGhee, K.H., 2004. *Automated pavement distress collection techniques*, vol. 334. Transportation Research Board. doi:10.17226/23348.

Mei, Q., & Gül, M. 2020. A cost-effective solution for pavement crack inspection using cameras and deep neural networks. *Construction and Building Materials*, 256, 119397.

Naddaf-Sh, M., Hosseini, S., Zhang, J., Brake, N. A., & Zargarzadeh, H. 2019. Real-time road crack mapping using an optimized convolutional neural network. *Complexity*, 1-17.

Ragnoli, A., De Blasiis, M. R., & Di Benedetto, A. 2018. Pavement distress detection methods: A review. *Infrastructures*, 3(4), 58-76.

Sabour, S., Frosst, N., & Hinton, G. E. 2017. Dynamic routing between capsules. *arXiv preprint*, arXiv:1710.09829.

Tsao, S., Kehtarnavaz, N., Chan, P., & Lytton, R. 1994. Image-based expert ‐ system approach to distress detection on CRC pavement. *Journal of Transportation Engineering*, 120(1), 52-64.

Yu, Y., Guan, H., Li, D., Gu, T., Wang, L., Ma, L., & Li, J. 2019. A hybrid capsule network for land cover classification using multispectral LiDAR data. *IEEE Geoscience and Remote Sensing Letters*, 17(7), 1263-1267.