Road Materials and Pavement Design

ISSN: 1468-0629 (Print) 2164-7402 (Online) Journal homepage: https://www.tandfonline.com/loi/trmp20

Convolutional neural network for pothole detection in asphalt pavement

Wanli Ye, Wei Jiang, Zheng Tong, Dongdong Yuan & Jingjing Xiao

Road Materials and Pavement Design, 2019
https://doi.org/10.1080/14680629.2019.1615533

Convolutional neural network for pothole detection in asphalt pavement

Wanli Ye[a,b], Wei Jiang[a,b], Zheng Tong[b,c*], Dongdong Yuan[a,b] and Jingjing Xiao[d]

[a]Key Laboratory for Special Area Highway Engineering of Ministry of Education, Chang'an University, Xi'an, PR People's Republic of China; [b]School of Highway, Chang'an University, Xi'an, PR People's Republic of China; [c]Sorbonne Université, Université de Technologie de Compiègne, CNRS, Compiègne cedex, France; [d]School of Civil Engineering, Chang'an University, Xi'an, PR People's Republic of China

(*Received 30 August 2018; accepted 28 April 2019*)

Many image processing techniques (IPTs) were proposed to inspect pavement defects for improving the precision and efficiency of the on-site inspections by humans. However, the various pavement conditions led to the unacceptable stability of IPTs. Therefore, an application of convolutional neural networks (CNNs) is presented for pothole detection using digital images in this study. Two CNNs, called conventional CNN and pre-pooling CNN, were designed, trained, and tested using 96,000 pavement images. Additionally, a stability study and comparative study were conducted based on the testing results. The main difference between the two CNNs was that a pre-pooling layer was used in the pre-pooling CNN to pre-process pavement images before the first convolutional layer. The results demonstrated that the optimised pre-pooling CNN had the 98.95% recognition precision in the testing. The stability study indicated that the optimised CNN model had the robustness in various real-world situations (e.g. light conditions and pavement materials). Compared with the traditional IPT methods, the CNN had a higher precision for extracting pothole features autonomously.

Keywords: pothole; convolutional neural network (CNN); feature extraction; structure optimisation; object recognition

1. Introduction

Pavement condition assessment is essential when developing road network maintenance programmes. Some studies (Chan, Huang, Yan, & Richards, 2010; Jiang et al., 2017; Vijay & Arya, 2006) showed that pavement defects had negative effects on pavement conditions, even increasing the risk of traffic accidents. As a typical pavement defect, potholes always lead to surface uneven and affect the driving comfort and safety (Kwon, Kim, Rhee, & Kim, 2018; Yang, Qian, & Song, 2015). Thus, it is necessary to detect potholes timely. However, pothole detection is very challenging, mostly because they have various shapes under complex real-world conditions.

Several methods for pavement inspection were presented to complete the task, mainly including 3D laser scanning (Chang, Chang, & Liu, 2005), vibration-based approach (De Zoysa, Keppitiyagama, Seneviratne, & Shihan, 2007; Yu & Yu, 2006), autonomous robots (Tseng, Kang, Chang, & Lee, 2011), and image processing techniques (IPTs) (Attard, Debono, Valentino, & Di Castro, 2018; Koch, Georgieva, Kasireddy, Akinci, & Fieguth, 2015). Compared with other methods, IPTs for pavement inspection were the most widely used one because of its flexibility. For example, Zakeri et al. (Zakeri, Nejad, & Fahimifar, 2016) employed a quadcopter-based

*Corresponding author. Email: zheng.tong@hds.utc.fr digital imaging system and support vector machines to collect pavement surface data and locate defect areas. Koch et al. (Koch & Brilakis, 2011) presented a method to detect potholes based on the histogram shape-based threshold of a defect region in pavement images. Zhang et al. (Zhang et al., 2015) applied a high spatial resolution multispectral digital aerial photography to collect data from pavement surface distresses. Li et al. (Li, He, Ju, & Du, 2014) proposed a long-distance crack detection technique based on IPTs. However, there are still certain problems: (1) conventional IPTs need manual assistance, which is time-consuming, labour-intensive, and the results depend on the experience of the inspectors; (2) the equipment is expensive; and (3) the precision is not stable under different conditions (e.g. light conditions and pavement materials). Therefore, there is a need to develop a method to detect potholes automatically with reasonable and stable precision.

Convolutional neural networks (CNNs), first proposed by Lecun and Bengio (1995), have been proved with high precision and robustness in object recognition (Barat & Ducottet, 2016; Leng, Guo, Zhang, & Xiong, 2015; Xu, Zhu, Wong, & Fang, 2016). In recent years, the CNN began to be utilised in civil engineering, especially in damage detection. For example, Abdeljaber et al. (Abdeljaber et al., 2018) proposed a CNN-based approach to detect structural damage. The results showed that the enhanced CNN could evaluate the actual amount of damages with acceptable precision. Lin et al. (Lin, Nie, & Ma, 2017) used a CNN to automatically extract features from low-level sensor data and detect damages based on the features. Cha et al. (Cha, Choi, & Büyüköztürk, 2017; Cha, Choi, Suh, Mahmoudkhani, & Büyüköztürk, 2018) proved that CNNs had the capacity of detecting pavement defects in real-world situations. Tong et al. proposed CNN-based methods to evaluate pavement quality, including surface crack lengths (Tong, Gao, Han, & Wang, 2017), reflection crack sizes (Tong, Gao, & Zhang, 2017), surface texture conditions (Tong, Gao, Sha, Hu, & Li, 2018). In summary, the advantages of the CNN-based method for damage detection are its automation and stability. Automation means that it requires no manual assistance in identifying potholes. Stability means that various conditions (e.g. different light conditions, different pavement materials, and other complex environments) have no observable influence on its precision of recognising pavement defects, such as potholes. Therefore, CNNs can be utilised to recognise and locate potholes using pavement images.

In this study, we used CNN to recognise and locate potholes in asphalt pavements. The main advantages of our method are that the processes of feature extraction are automatic, and its precision is acceptable and stable in the various conditions. The main contribution of this study is that we propose an accurate and robust method for pothole recognition and detection to various real-world conditions (e.g. different light conditions) to replace the manual pavement inspection methods autonomously, whose results can be used as the basis of pavement maintenance. The rest of this paper is organised as follows: the methodology is introduced in section 2, including the structure description and parameter optimisation, the generation of the CNN database, and stability and superiority analysis methods. The results are discussed in section 3, including parameter optimisation results, a stability study, and a comparative study. Finally, the conclusions are summarised in section 4.

2. Methodology

The main research procedure is shown in Figure 1, including acquiring raw images, generating a database, training and testing CNNs and find the optimal ones, and analysing its stability and superiority. The processes of developing our CNNs are introduced in section 2.1, as shown in Figure 2. The generation of the database for our CNNs is presented in section 2.2. The methods to evaluate the stability and superiority of CNNs are shown in section 2.3.

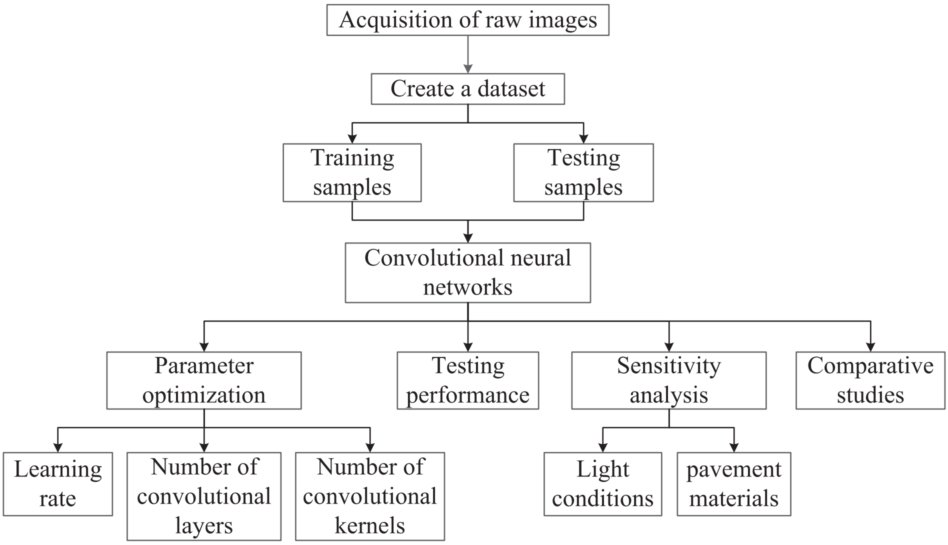


Figure 1. Flow chart of the methodology.

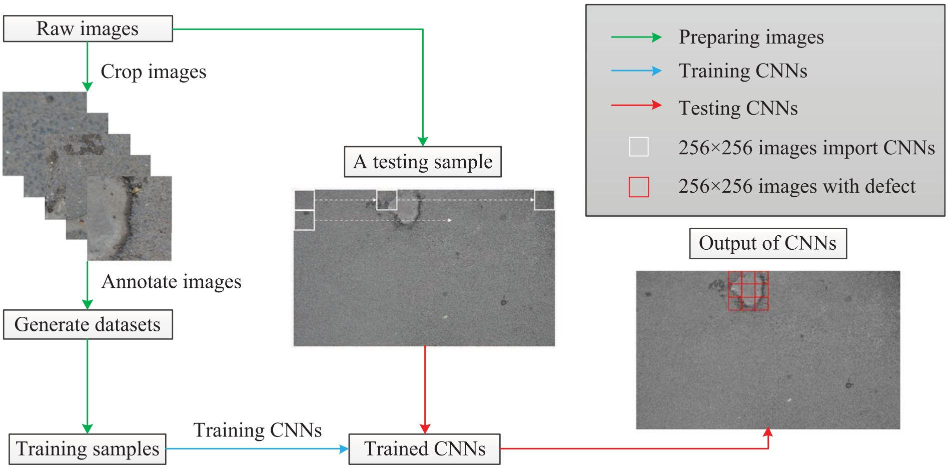


Figure 2. Flow chart of developing CNNs.

2.1. *Deep learning for pothole detection*

2.1.1. *Brief overview of convolutional neural network*

All of our CNNs are composed of an input layer, convolutional layers, pooling layers, fully connected layers, and an output layer. There are several special layers in a CNN model:

(1) Convolutional layer. The function of a convolutional layer is to extract features from the input data by convolution operations (LeCun & Bengio, 1995). A dot product operation was performed between the convolutional kernel and the receptive field in the input data, and the convolutional kernel continued to move at a certain stride to match different receptive fields

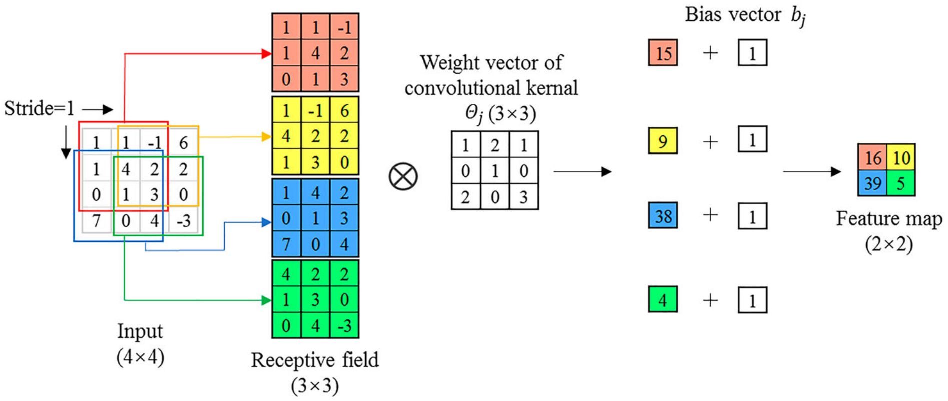


Figure 3. An example of convolution.

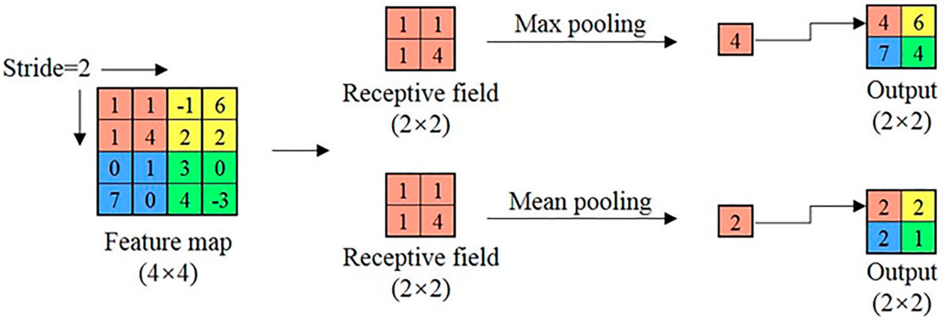


Figure 4. An example of pooling.

(Cha et al., 2017), which was shown in Figure 3 and Equation (1).

$$I_j = \phi\left(\sum_i x_i * \Theta_j + b_j\right) \tag{1}$$

where Θ_j and b_j are the weight vector and bias vector of the convolutional kernel j, respectively.

(2) Pooling layer. The overfitting would occur while there are too many imported data in the CNN (Leinweber, 2007). Thus, a pooling layer is used to reduce to the resolutions of the feature maps acquired by convolutional layers, which reduced the imported data in the CNN to prevent the overfitting. Additionally, by the reduction of the resolutions of the feature maps, the sensitivity of the output to shifts and distortions will also reduce (LeCun & Bengio, 1995). Max pooling and mean pooling are widely used in pooling operation, and their operation processes are shown in Figure 4. A study (Wu & Gu, 2015) showed that max pooling had better precision in pattern recognition using digital images than mean pooling. Therefore, the max pooling was used in this research.

(3) Fully connected layer with dropout. The fully connected layer is used to resize the features extracted by the convolutional and pooling operation to guarantee that features can be mapped regardless of their sizes, which greatly enhanced the robustness of a CNN.

Dropout was employed in fully connected layers in our CNNs as a trick to prevent overfitting and data contamination. Its validity has been proved in the study of Srivastava et al. (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014). In each iteration, neurons in the

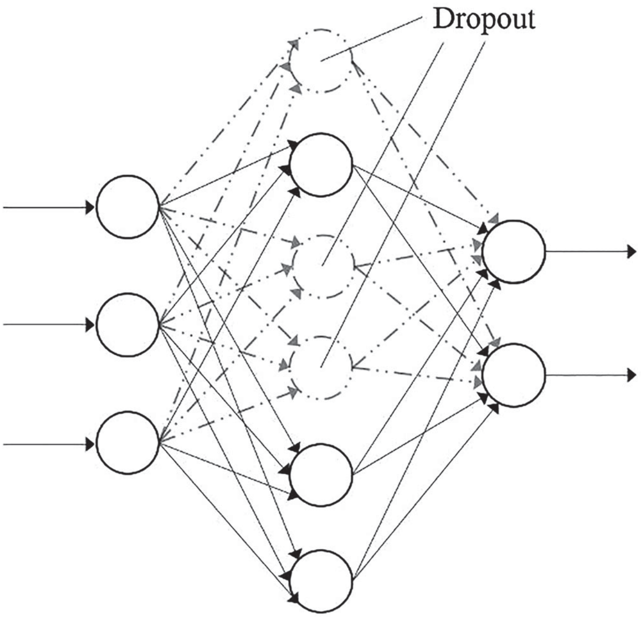


Figure 5. The process of dropout.

fully connected layers were dropout temporarily according to a certain probability as shown in Figure 5. The precision will not be affected by the insufficient data resulted in the dropout operation, such as a small part of images with inaccurate labels (Krizhevsky, Sutskever, & Hinton, 2012). The parameter of the dropout rate in our research was 0.5, referred to a previous study (Srivastava et al., 2014).

(4) Softmax layer. The Softmax layer can map the output of fully connected layers to (0,1) for a classification task, which is implemented by Equation (2),

$$P(y^{(i)} = n|x^{(i)}; W) = \begin{bmatrix} P(y^{(i)} = 1|x^{(i)}; W) \\ P(y^{(i)} = 2|x^{(i)}; W) \\ \cdots \\ P(y^{(i)} = n|x^{(i)}; W) \end{bmatrix} = \frac{1}{\sum_{j=1}^{n} e^{W^T x^{(i)}}} \begin{bmatrix} e^{W_1{}^T x^{(i)}} \\ e^{W_2{}^T x^{(i)}} \\ \cdots \\ e^{W_n{}^T x^{(i)}} \end{bmatrix} \tag{2}$$

where $P(y^{(i)} = n|x^{(i)}; W)$ is the possibility of the i^{th} image being the n^{th} disease, and the sum of all possibilities is guaranteed to be 1. $n = 2$ in this research. W is a weight matrix. The loss function of Softmax is Equation (3),

$$L = \frac{1}{m}\left[\sum_{i=1}^{m}\sum_{j=1}^{n} 1\{y_i = j\} \log \frac{e^{W_j^T x_i}}{\sum_{l=1}^{n} e^{W_i^T x_i}}\right] + \frac{\lambda}{2}\sum_{j=1}^{n} W_j^2 \tag{3}$$

where $1\{y_i = j\}$ is a logical expression that returns either 0 or 1, $x^{(i)}$ is the matrix associated with the pavement image, W is a parameter matrix obtained by training, and $\frac{\lambda}{2}\sum_{j=1}^{n} W_j^2$ is used to prevent overfitting (Bengio, 2012).

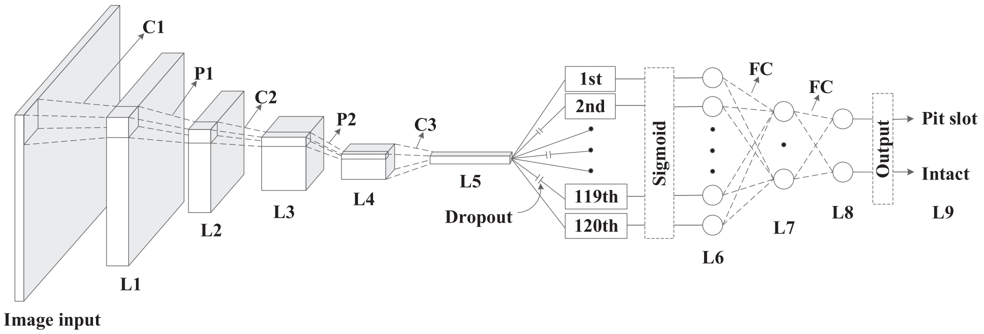


Figure 6. A conventional convolutional neural network.

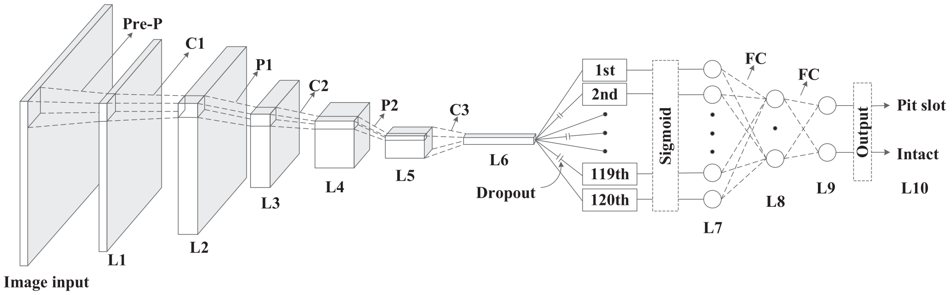


Figure 7. A Pre-pooling convolutional neural network.

2.1.2. *Different CNN structures*

For a conventional CNN, as shown in Figure 6 (Cha et al., 2017), the input image first passes through the 1st convolution layer, then the 1st pooling layer and other layers. Finally, the Softmax layer predicts whether the image includes a pothole or not. However, pothole areas are only a small part of pavement surfaces, which means the efficiency of extracting features was low in the model. Thus, we added a pre-pooling layer in the CNN model as shown in Figure 7 to solve the problem. The pre-pooling layer reduces the resolution of the imported data by a max-pooling operation before the first convolutional operation. After the pre-pooling, the characteristics irrelevant to potholes were reduced, such as pavement materials and image noises. However, the characteristics relevant to the classification task, such as the pothole outlines, were still reserved. Therefore, a pre-pooling layer was used in CNN before the 1st convolution layer to improve the efficiency and precision of pothole detection, as shown in Figure 7. Therefore, two structures of CNN were employed to recognise potholes and their precisions were compared to verify the superiority of the pre-pooling CNN.

2.1.3. *Parameter optimisation*

Parameter optimisation is one of the most important steps of training the CNN. For example, a too large learning rate will lead to the programme to fail to converge, while a too small learning rate will lead to a waste of time during the training. In general classification algorithms, the learning rate was kept from 0.01–1, the number of convolutional layers was kept from 1 to 5, and the number of convolutional kernels was kept from 10 to 50 (Cha et al., 2017; Tong et al., 2017). In order to optimise the structural parameters of CNNs, different parameters of CNNs were combined to find the optimal one, as shown in Table 1.

Table 1. Orthogonal test combinations.

No.	Learning rate	Number of convolutional layers	Number of convolutional kernels
1	0.01	1	10
2	0.01	2	20
3	0.01	3	30
4	0.01	4	40
5	0.01	5	50
6	0.05	1	20
7	0.05	2	30
8	0.05	3	40
9	0.05	4	50
10	0.05	5	10
11	0.20	1	30
12	0.20	2	40
13	0.20	3	50
14	0.20	4	10
15	0.20	5	20
16	0.50	1	40
17	0.50	2	50
18	0.50	3	10
19	0.50	4	20
20	0.50	5	30
21	1.00	1	50
22	1.00	2	10
23	1.00	3	20
24	1.00	4	30
25	1.00	5	40

Note: the number of convolution kernels in the 2st convolution was twice that of the 1st convolution layer empirically.

2.2. *Database and implementation details*

To train and test the two types of CNNs, a database including 400 raw potholes images of 5,120 pixels × 3,072 pixels resolution was generated. Notably, the 400 raw images were collected from different pavements (AC-13, AC-16, and SMA-16) under different light conditions. All these images were cropped into 96,000 small images of 256 pixels × 256 pixels resolution. Then, 72,000 small images were selected randomly as a training dataset and the rest was used as a testing dataset. In each iteration of training, 5,000 cropped images from the training dataset were selected randomly for validation.

After the two datasets were generated, they were labelled respectively. The smaller images were manually annotated as "*Intact*" or "*Pothole*", which were quadrature-encoded as $[1\ 0]'$ and $[0\ 1]'$, respectively. The two CNNs were trained and tested using Caffe (Jia et al., 2014) on a computer equipped with an Intel(R) Core(TM) i7-8750H CPU, 32.00 GB RAM, and an NVIDIA GeForce GTX 1080 8 GB GPU.

2.3. *Stability and superiority analysis methods*

2.3.1. *Stability analysis*

The average precision in the testing can only reflect the overall precision of the model but not indicate the stability of the model to various conditions. For example, the model may detect all the potholes in the AC-13 pavement under the bright light condition but only detect a part of potholes in the SMA-16 pavement under shadowy light conditions. Therefore, a stability analysis of the CNN was necessary and the mean squared error (MSE) was applied to evaluate the stability of CNN, which was shown in Equation (4).

$$MSE = \frac{1}{m}\left[\sum_{i=1}^{m}\sum_{j=1}^{n} 1\{y_i = j\} \log \frac{e^{W_j^T x_i}}{\sum_{l=1}^{n} e^{W_i^T x_i}}\right] \tag{4}$$

where m and n are the number of images and classification in the stability analysis; W_j^T is the weights in the Softmax layer; x_i is the data imported into the Softmax layer.

The stability analysis consisted of two parts:

(1) Different light conditions. In the testing dataset, 60 raw images under 3 different light conditions (bright, shadow and dark) were selected and there were 20 images for each one. In this part, the illuminance ranges of the bright and dark condition were $500 - 1000$ *lx* and $5 - 300$ *lx*, respectively. The shadow condition meant there was partial occlusion on the pavement surface. The detection results of the CNN were used to verify its stability to the light conditions.

(2) Different pavement materials. In the testing dataset, 60 raw images from 3 different highway were selected, whose pavement materials were AC-13, AC-16, and SMA-16, respectively. There were 20 images for each one. The detection results of the CNN were used to verify its stability to the pavement materials.

2.3.2. *Superiority analysis*

The CNN-based method was compared with two state-of-the-art methods (Sobel edge detection and K-means clustering analysis) for detecting potholes. Sobel edge detection detected potholes by extract the edges of the objects. K-means clustering detected potholes based on the differences in gray value ranges between potholes and pavements. In the testing dataset, 3 raw images under different situations were selected and the detection results of the three methods were compared to verify the CNN-based method's superiority.

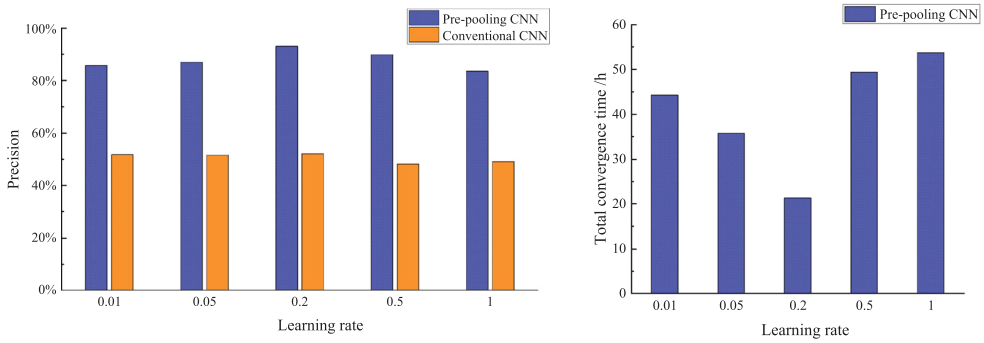
3. Results and discussion

3.1. *Structure optimisation*

By comparing the precision of CNNs under different parameter combinations, the optimised structure was selected, mainly including a comparison of two CNN models, and optimisation of parameters based on precision and efficiency, where the precision meant the average accuracy of each trained CNN to detect potholes, and the efficiency was measured by the total convergence time during each training process.

(1) Pre-pooling optimisation. The detection precisions of two structures after 1500 iterations were shown in Figure 8(a). The results showed the pre-pooling CNNs detected potholes with acceptable precisions, while the conventional CNN could not recognise the pothole in the images precisely. It indicated that the precision of the pre-pooling CNNs was better than the conventional ones. Therefore, the utilisation of the pre-pooling layer had the capacity of improving the precision.

(2) Optimisation for learning rate. Figure 8(a) presented that the test precision increased first and then decreased as the learning rate increased. The test precision of the pre-pooling CNN was maximum when the learning rate was 0.2, which was 93.07%. The test precisions were 85.74%



(a) Precision

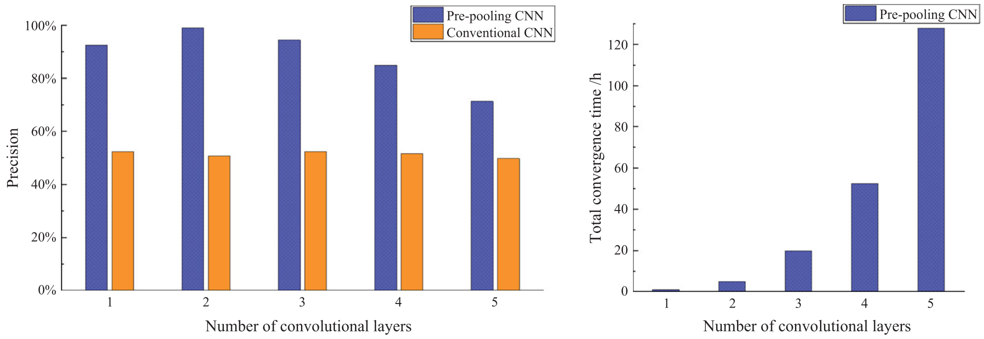
(b) Efficiency

Figure 8. The optimisation of the two CNNs with different learning rates.

and 83.49% when the learning rates were 0.01 and 1. It was because the pre-pooling CNNs with low or high learning rate learned little or too many local features, which always led overfitting. Figure 8(b) presented that the total convergence time increased first and then decreased as the learning rate increased. The total convergence time of the pre-pooling CNN was minimum when the learning rate was 0.2, which was 21.28 h. The total convergence time was 44.27 and 53.73 h when the learning rate were 0.01 and 1. This was because a small learning rate led to too many steps of feature extraction, which not only increased the total convergence time but also led to overfitting easily. However, the CNN ignored a large part of useful features in the process of the training when the learning rate was too large. It indicated that the 0.2 learning rate was suitable for our pre-pooling CNN.

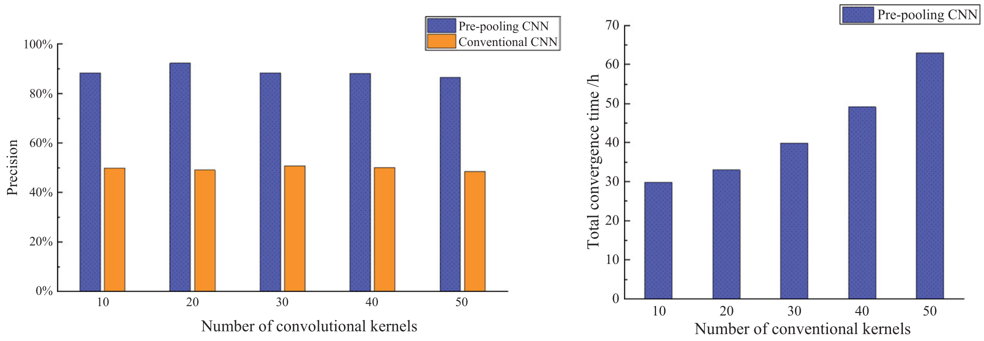
(3) Optimisation for the number of convolutional layers. Figure 9(a) presented that the test precision increased first and then decreased as the number of convolutional layers increased. The number of convolutional layers had a more significant impact on precision than the learning rate. The test precision of the pre-pooling CNN was maximum when the number of convolutional layers was 2, which was 98.95%. The test precisions were 92.46% and 71.31% when the number of convolutional layers was 1 and 5. It was because the convolutional layers only extracted the limited low-dimensional features of potholes when the number of convolutional layers was small, which led the Softmax layer could not classify the defects precisely. However, too many features were extracted during the convolution process when the number of convolutional layers was too large. Too many features, especially some unnecessary features, led to overfitting and reduced the precision. Figure 9(b) presented that the total convergence time increased as the number of convolutional layers increased. This was because the parameters involved in the training process increased correspondingly as the number of convolutional layers increased. Therefore, considering both the precision and efficiency, the 2 convolution layers was suitable for our pre-pooling CNN.

(4) Optimisation for convolutional kernels. Figure 10(a) presented that the test precision increased first and then decreased as the number of convolutional kernels increased. The test precision of the pre-pooling CNN was maximum, which was 92.24% when the number of convolution kernels was 20, which was slightly higher than other cases. This was because the number of convolution kernels determined the number of features extracted by CNN. The features of the potholes were limited, and a certain number of kernels could meet the demand of extracting features. However, too many features from kernels, especially some unnecessary features, led overfitting and reduced the precision. Figure 10(b) presented that the total convergence time increased as the number of convolutional kernels increased. This was because the number of



(a) Precision (b) Efficiency

Figure 9. The optimisation of the two CNNs with different convolutional layers.



(a) Precision (b) Efficiency

Figure 10. The optimisation of the two CNNs with different convolutional kernels.

convolution kernels determined the number of features extracted by CNN, which led more convolution operations and more parameters and thereby increased the amount of computation. Therefore, considering both the precision and efficiency, 20 convolutional kernels in the first convolution layer was suitable for our pre-pooling CNN.

(5) CNN with the best performance. The results turned out that the optimal learning rate, number of convolution layers, and number of convolutional kernels in 1st convolutional layer were 0.2, 2, and 20. Empirically, the number of convolution kernels in the 2nd convolution should be about twice that of the 1st convolution layer. Thus, and the number of convolutional kernels in the 2nd convolutional layer should be 40. Besides, the structure parameters of the CNN were shown in Table 2.

3.2. *Testing for different light conditions*

We selected 60 images from the testing dataset based on their light conditions, including bright, shadow, and dark. Each light condition had 20 images. These images were cropped to obtain small images of 256 pixels × 256 pixels resolution and then were imported into the optimised pre-pooling CNN to verify its stability of different light conditions.

Figure 11 presented the convergence processes of the pre-pooling CNN to the 60 images during the training. For the bright condition, the pre-pooling CNN converged quickly, and the fluctuation of MSE during iteration was small. It indicated that the pre-pooling CNN detected

Table 2. Detailed parameters of the pre-pooling CNN.

Layer number	Type	Input size	Filter size	Number	Stride
L0	Input	256 × 256 × 1	–	–	–
L1	Pre-pooling	256 × 256 × 1	–	–	–
L2	Convolution	128 × 128 × 1	5 × 5 × 1	20	2
L3	Pooling	62 × 62 × 20	-	-	2
L4	Convolution	31 × 31 × 20	5 × 5 × 20	40	2
L5	Pooling	13 × 13 × 40	–	–	–
L6	Convolution	6 × 6 × 40	6 × 6 × 40	120	1
L7	Sigmoid	1 × 1 × 120	–	–	–
L8	Fully connected	1 × 1 × 120	–	84	1
L9	Fully connected	1 × 1 × 84	–	2	1
L10	Output	1 × 1 × 2	–	–	–

Notes: The function of the 3rd convolution layer was to reshape the output of L5 to a vector in 1 × 1 × 120, which was different from L2 and L4. Since this layer did not require the optimisation of the number of the kernels, the optimisation for the number of convolutional layers mentioned in this paper did not include this layer.

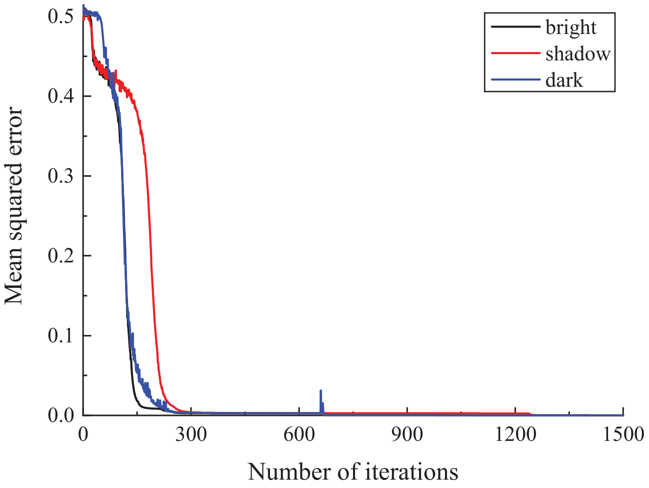
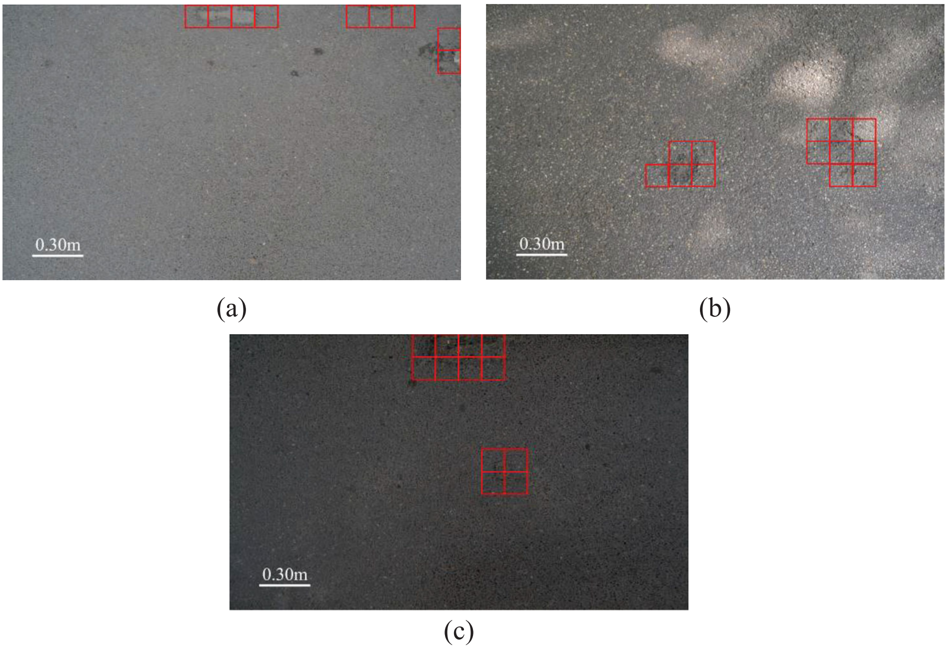


Figure 11. Performance of the pre-pooling CNN with different light conditions.

potholes accurately under the bright conditions as shown in Figure 12(a). For the shadow condition, the bright spot had a negative effect on the convergence of the training as shown in the red curve in Figure 11, but it had no influence on the final results as shown in Figure 12(b). It indicated that the well-trained pre-pooling CNN could recognise the light spots and potholes well. Similarly, the deep colour of the pavement had a negative effect on the convergence of the training as shown in the blue curve in Figure 11. For example, a large fluctuation occurred in the 660th iterations. However, it had no influence on the final results as shown in Figure 12(c).

3.3. *Testing for different pavement materials*

We selected 60 images from the testing dataset based on their pavement materials, including AC-13, AC-16, and SMA-16. Each pavement material had 20 images. These images were cropped to obtain small images of 256 pixels × 256 pixels resolution and then were imported into the pre-pooling CNN.



(a) (b)

(c)

Figure 12. Examples of the Pre-pooling CNN detecting results with different light conditions. (a) The pavement under bright light condition (500 - 1000 *lx*), (b) The pavement under shadow light condition and (c) The pavement under dark light condition (5 - 300 *lx*).

Figure 13 presented the convergence processes of the pre-pooling CNN to the 60 images during the training. For AC-13 pavements, the pre-pooling CNN converged quickly, and the fluctuation of MSE during iteration was small. It indicated that the pre-pooling CNN detected potholes accurately in AC-13 pavements as shown in Figure 14(a). However, the texture of AC-16 and SMA-16 pavements had a negative effect on the convergence of the training as shown in the red and blue curves in Figure 13, but it had no influence on the final results as shown in Figure 14(b) and Figure 14(c). It indicated that the well-trained pre-pooling CNN could recognise the potholes in different pavements.

3.4. *Comparative study*

The comparison of precision between two well-known methods (Sobel edge detection and K-means clustering analysis) and the CNN-based method was used to verify the CNN-based method's superiority. In order to reflect the complexity of the real-world situations, 3 images with different light conditions and pavement materials in the test set were selected.

Figure 15(a) presents an AC-13 pavement under a bright light condition. The output of the CNN-based method was accurate as shown in Figure 15(b). Sobel edge detection could not extract edge features of potholes as shown in Figure 15(c), while the results of K-means clustering included much noise as shown in Figure 15(d).

Figure 16(a) presents an AC-16 pavement under a shadow light condition. The output of the CNN-based method was accurate as shown in Figure 16(b). However, the two traditional methods obtained little features about the pothole, as shown in Figure 16(c and d).

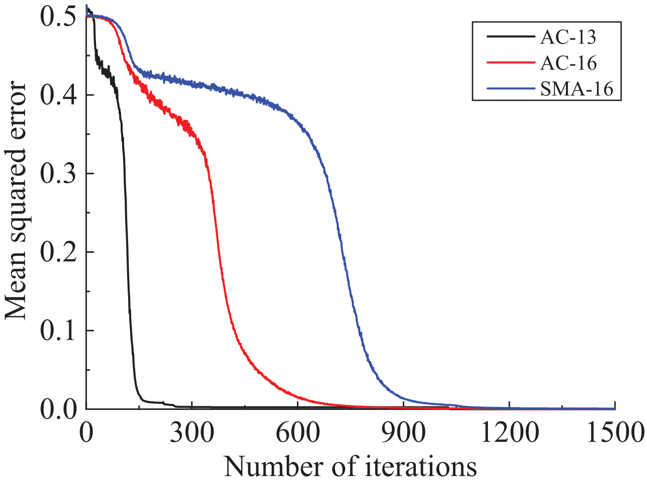


Figure 13. Performance of the pre-pooling CNN in different pavements.

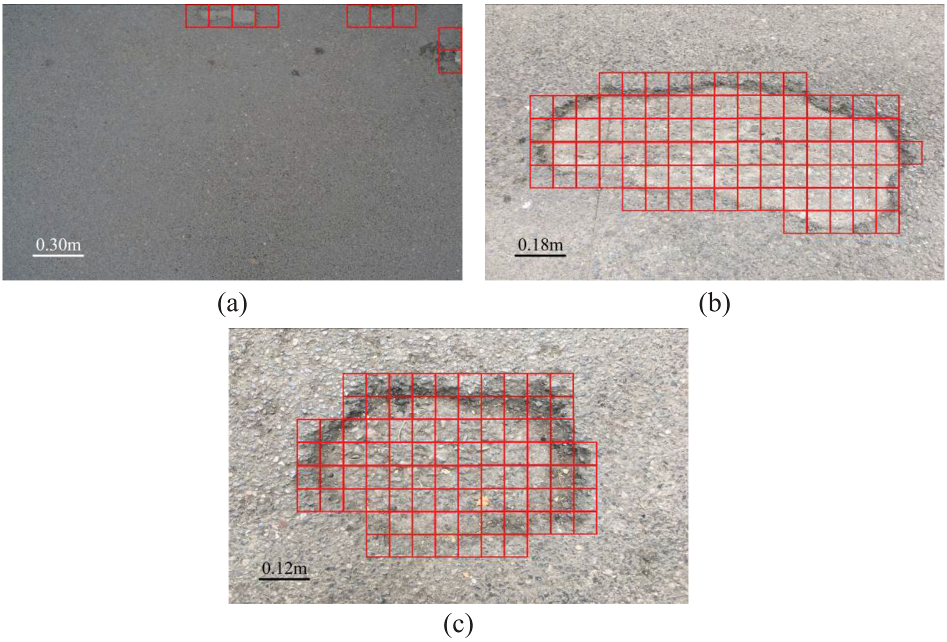
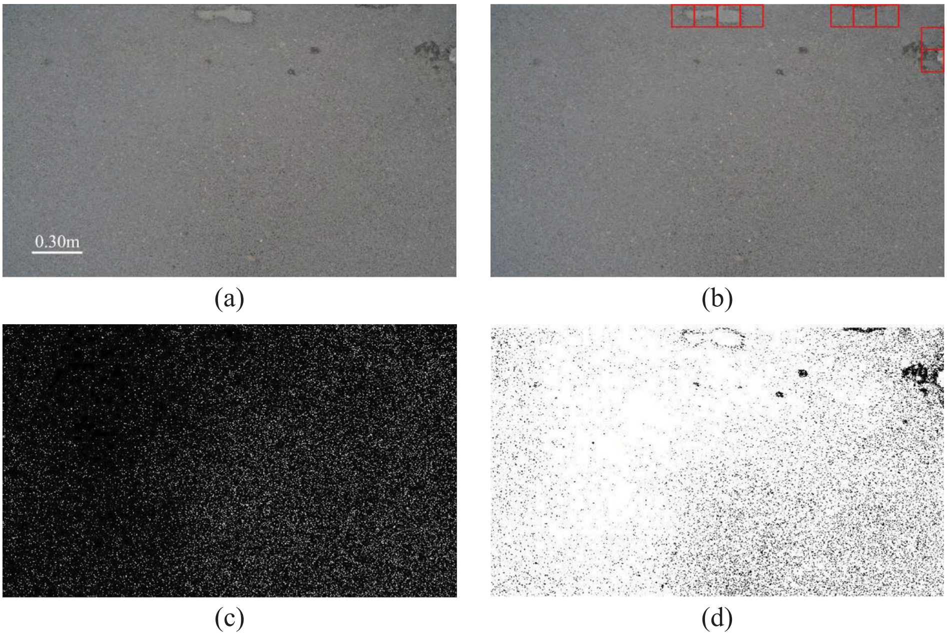


Figure 14. Examples of the Pre-pooling CNN detecting results in different pavements. (a) AC-13, (b) AC-16 and (c) SMA-16.

Figure 17(a) presents an SMA-16 pavement under a dark light condition. The output of the CNN-based method was acceptable as shown in Figure 17(b), even in some areas without potholes was included. However, the two traditional methods obtained little features about the pothole, as shown in Figure 17(c and d).

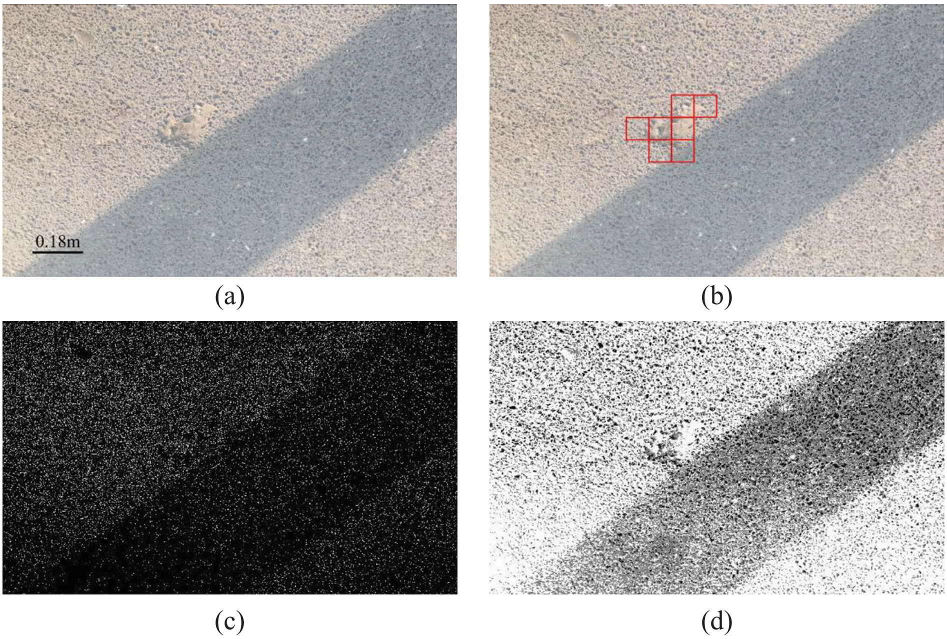
It indicated that the CNN-based method recognised and located potholes accurately under various conditions. Although K-means cluster analysis and Sobel edge detection extracted some features about potholes, it could not accurately recognise and locate all of them. Therefore, the CNN-based method had superiority in pothole detection under various conditions.



(a) (b)

(c) (d)

Figure 15. Pothole with the bright light condition from highway 1. (a) Raw image, (b) Pre-pooling CNN, (c) Sobel edge detection and (d) K-means clustering analysis.



(a) (b)

(c) (d)

Figure 16. Pothole with the shadow light condition from highway 2. (a) Raw image, (b) Pre-pooling CNN, (c) Sobel edge detection and (d) K-means clustering analysis.

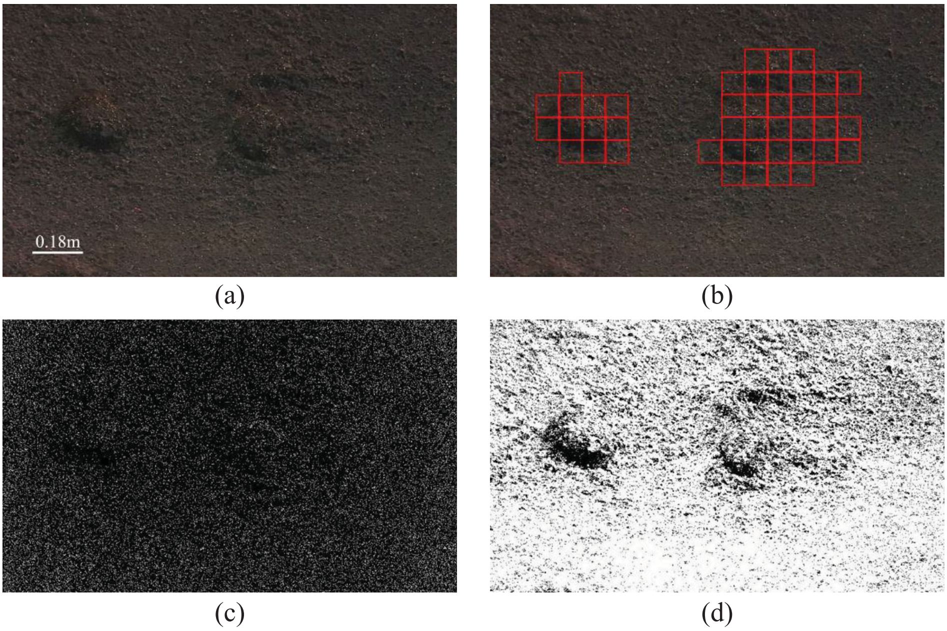


Figure 17. Pothole with the dark light condition from highway 3. (a) Raw image, (b) Pre-pooling CNN, (c) Sobel edge detection and (d) K-means clustering analysis.

4. Conclusions

The application of CNN using pavement images for pothole detection was presented in this study, and the following conclusions can be drawn:

(1) The pre-pooling CNN had the average precision higher than 80% during the training, which meets the demands of the pavement inspection. The precision of the optimised pre-pooling CNN was 98.95%, whose learning rate was 0.2, the number of convolution layers was 2, and the number of convolutional kernels was 20.

(2) Compared with the average precision of the pre-pooling CNN and the conventional CNN, it indicated that the pre-pooling layer could improve the precision for pothole detection using the same database.

(3) The results of the stability analysis indicated that the outputs of the pre-pooling CNN were little affected by different light conditions and pavement materials. It indicated that the CNN-based method could be applied under various real-world conditions.

(4) The comparative study results demonstrated that the CNN-based method recognised and located potholes accurately under various conditions. Although K-means cluster analysis and Sobel edge detection extracted some information about potholes, it could not accurately recognise and locate all of them. Therefore, compared with other conventional methods, the CNN-based method had superiority in pothole detection.

Funding

This project was jointly supported by the National Natural Science Foundation of China (grant number 51608043), the Youth Top-notch Talent Support Program of Shaanxi Province, the Fok Ying-Tong Education Foundation (grant number 161072), and the Fundamental Research Funds for the Central Universities (grant number 300102219317).

ORCID

Wanli Ye http://orcid.org/0000-0001-8164-0517

References

Abdeljaber, O., Avci, O., Kiranyaz, M. S., Boashash, B., Sodano, H., & Inman, D. J. (2018). 1-D CNNs for structural damage detection: Verification on a structural health monitoring benchmark data. *Neurocomputing*, *275*, 1308–1317. doi:10.1016/j.neucom.2017.09.069

Attard, L., Debono, C. J., Valentino, G., & Di Castro, M. (2018). Tunnel inspection using photogrammetric techniques and image processing: A review. *ISPRS Journal of Photogrammetry and Remote Sensing*, *144*, 180–188. doi:10.1016/j.isprsjprs.2018.07.010

Barat, C., & Ducottet, C. (2016). String representations and distances in deep convolutional neural networks for image classification. *Pattern Recognition*, *54*, 104–115. doi:10.1016/j.patcog.2016.01.007

Bengio, Y. (2012). Practical recommendations for gradient-based training of deep architectures. *Neural Networks: Tricks of the Trade*, *7700*, 437–478. doi:10.1007/978-3-642-35289-8_26

Cha, Y. J., Choi, W., & Büyüköztürk, O. (2017). Deep learning-based crack damage detection using convolutional neural networks. *Computer-Aided Civil and Infrastructure Engineering*, *32*(5), 361–378. doi:10.1111/mice.12263

Cha, Y. J., Choi, W., Suh, G., Mahmoudkhani, S., & Büyüköztürk, O. (2018). Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. *Computer-Aided Civil and Infrastructure Engineering*, *33*(9), 731–747. doi:10.1111/mice.12334

Chan, C. Y., Huang, B., Yan, X., & Richards, S. (2010). Investigating effects of asphalt pavement conditions on traffic accidents in Tennessee based on the pavement management system (PMS). *Journal of Advanced Transportation*, *44*(3), 150–161. doi:10.1002/atr.129

Chang, K. T., Chang, J. R., & Liu, J. K. (2005). Detection of pavement distresses using 3D laser scanning technology. *Computing in Civil Engineering*, 1–11. doi:10.1061/40794(179)103

De Zoysa, K., Keppitiyagama, C., Seneviratne, G. P., & Shihan, W. W. A. T. (2007, August). A public transport system based sensor network for road surface condition monitoring. In *Proceedings of the 2007 workshop on Networked systems for developing regions*, 9.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., . . . Darrell, T. (2014, November). Caffe: Convolutional architecture for fast feature embedding. *Proceedings of the 22nd ACM International Conference on Multimedia*, 675–678. doi:10.1145/2647868.2654889

Jiang, W., Yuan, D., Xu, S., Hu, H., Xiao, J., Sha, A., & Huang, Y. (2017). Energy harvesting from asphalt pavement using thermoelectric technology. *Applied Energy*, *205*, 941–950. doi:10.1016/j.apenergy.2017.08.091

Koch, C., & Brilakis, I. (2011). Pothole detection in asphalt pavement images. *Advanced Engineering Informatics*, *25*(3), 507–515. doi:10.1016/j.aei.2011.01.002

Koch, C., Georgieva, K., Kasireddy, V., Akinci, B., & Fieguth, P. (2015). A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure. *Advanced Engineering Informatics*, *29*(2), 196–210. doi:10.1016/j.aei.2015.01.008

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 1097–1105. doi:10.1145/3065386

Kwon, B. J., Kim, D., Rhee, S. K., & Kim, Y. R. (2018). Spray injection patching for pothole repair using 100 percent reclaimed asphalt pavement. *Construction and Building Materials*, *166*), doi:10.1016/j.conbuildmat.2018.01.145

LeCun, Y., & Bengio, Y. (1995). Convolutional networks for images, speech, and time series. In M. A. Arbib (Ed.), *The handbook of brain theory and neural networks* (pp. 255–258). Cambridge: MIT Press.

Leinweber, D. J. (2007). Stupid data miner tricks: Overfitting the S&P 500. *Journal of Investing*, *16*(1), 15. doi:10.3905/joi.2007.681820

Leng, B., Guo, S., Zhang, X., & Xiong, Z. (2015). 3D object retrieval with stacked local convolutional autoencoder. *Signal Processing*, *112*, 119–128. doi:10.1016/j.sigpro.2014.09.005

Li, G., He, S., Ju, Y., & Du, K. (2014). Long-distance precision inspection method for bridge cracks with image processing. *Automation in Construction*, *41*, 83–95. doi:10.1016/j.autcon.2013.10.021

Lin, Y. Z., Nie, Z. H., & Ma, H. W. (2017). Structural damage detection with automatic feature-extraction through deep learning. *Computer-Aided Civil and Infrastructure Engineering*, *32*(12), 1025–1046. doi:10.1111/mice.12313

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, *15*(1), 1929–1958. doi:10.1214/12-AOS1000

Tong, Z., Gao, J., Han, Z., & Wang, Z. (2017). Recognition of asphalt pavement crack length using deep convolutional neural networks. *Road Materials and Pavement Design*, 1–16. doi:10.1080/14680629.2017.1308265

Tong, Z., Gao, J., Sha, A., Hu, L., & Li, S. (2018). Convolutional neural network for asphalt pavement surface texture analysis. *Computer-Aided Civil and Infrastructure Engineering*, *33*(12), 1056–1072. doi:10.1111/mice.12406

Tong, Z., Gao, J., & Zhang, H. (2017). Recognition, location, measurement, and 3D reconstruction of concealed cracks using convolutional neural networks. *Construction and Building Materials*, *146*, 775–787. doi:10.1016/j.conbuildmat.2017.04.097

Tseng, Y. H., Kang, S. C., Chang, J. R., & Lee, C. H. (2011). Strategies for autonomous robots to inspect pavement distresses. *Automation in Construction*, *20*(8), 1156–1172. doi:10.1016/j.autcon.2011.04.018

Vijay, S., & Arya, K. (2006). Low Cost–FPGA based system for pothole detection on Indian Roads. *M-Tech Thesis, Indian Institute of Technology Bombay*.

Wu, H., & Gu, X. (2015). Towards dropout training for convolutional neural networks. *Neural Networks*, *71*, 1–10. doi:10.1016/j.neunet.2015.07.007

Xu, T., Zhu, F., Wong, E. K., & Fang, Y. (2016). Dual many-to-one-encoder-based transfer learning for cross-dataset human action recognition. *Image and Vision Computing*, *55*, 127–137. doi:10.1016/j.imavis.2016.01.001

Yang, Y., Qian, Z., & Song, X. (2015). A pothole patching material for epoxy asphalt pavement on steel bridges: Fatigue test and numerical analysis. *Construction and Building Materials*, *94*, 299–305. doi:10.1016/j.conbuildmat.2015.07.017

Yu, B. X., & Yu, X. (2006). Vibration-based system for pavement condition evaluation. *Applications of Advanced Technology in Transportation*, 183–189. doi:10.1061/40799(213)31

Zakeri, H., Nejad, F. M., & Fahimifar, A. (2016). Rahbin: A quadcopter unmanned aerial vehicle based on a systematic image processing approach toward an automated asphalt pavement inspection. *Automation in Construction*, *72*, 211–235. doi:10.1016/j.autcon.2016.09.002

Zhang, S., Bogus, S. M., Lippitt, C. D., Neville, P. R., Zhang, G., Chen, C., & Valentin, V. (2015). Extracting pavement surface distress conditions based on high spatial resolution multispectral digital aerial photography. *Photogrammetric Engineering & Remote Sensing*, *81*(9), 709–720. doi:10.14358/PERS.81.9.709