Road Materials and Pavement Design

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

Recognition of asphalt pavement crack length using deep convolutional neural networks

Zheng Tong, Jie Gao, Zhenqiang Han & Zhenjun Wang

To cite this article: Zheng Tong, Jie Gao, Zhenqiang Han & Zhenjun Wang (2018) Recognition of asphalt pavement crack length using deep convolutional neural networks, Road Materials and Pavement Design, 19:6, 1334-1349, DOI: 10.1080/14680629.2017.1308265

To link to this article: https://doi.org/10.1080/14680629.2017.1308265

Road Materials and Pavement Design, 2018
Vol. 19, No. 6, 1334–1349, https://doi.org/10.1080/14680629.2017.1308265

Recognition of asphalt pavement crack length using deep convolutional neural networks

Zheng Tong[a], Jie Gao[a,b], Zhenqiang Han[a] and Zhenjun Wang[c]*

[a]*School of Highway, Chang'an University, Xi'an 710061, People's Republic of China;* [b]*National and Local Joint Engineering Materials Laboratory of Traffic Engineering and Civil Engineering, Chongqing Jiaotong University, Chongqing 400074, People's Republic of China;* [c]*School of Materials Science and Engineering, Chang'an University, Xi'an 710061, People's Republic of China*

(*Received 7 July 2016; accepted 9 March 2017*)

Crack length measurement is an important part of asphalt pavement detection. However, some crack measurement techniques cannot satisfy the needs of accuracy and efficiency. This study discusses application of deep convolutional neural networks (DCNN) in automatic recognition of pavement crack length in batches. Original red, green and blue images were transformed to grey-scale images to calculate their threshold and pre-extract cracks' properties by k-means clustering analysis. Then the pre-extracted crack images were used as both training and testing samples. The process of accomplishing DCNN to recognise the crack length included the structure designing, training, and testing of the networks. The output results of well-trained DCNN were compared with those of the actual measurement to verify the accuracy of the networks. The result indicates that the training strategy including two processes overcomes the lack of crack labelled images and improves the accuracy of the network, combining with quadrature encoding and stochastic gradient descent. Recognition accuracy of DCNN is 94.36%, maximum length error is 1 cm and mean squared error is 0.2377. The error rates of length ranges 6–7 cm and 7–8 cm are bigger than other ranges Therefore, the networks can be adopted to measure the crack length accurately, but more 6–8 cm crack images should be used to improve the accuracy of the networks in future.

Keywords: asphalt pavement crack; deep convolutional neural networks (DCNN); deep learning; picture processing; stochastic gradient descent; k-means clustering analysis

1. Introduction

Crack is a type of pavement disease, which is also an important evolution part of pavement surface condition index. The attenuation effect of transverse and longitudinal cracks at pavement condition is evaluated by equivalent area impact of crack length. Accordingly recognising crack length automatically and efficiently is significant for the guidance of pavement maintaining. However, manual measurement and eye measurement having disadvantages of large workloads and low accuracy, do not meet the increasing demand of maintaining pavements. Many scholars, who are devoted to developing the crack detection system, have made great progresses. The Komatsu system realised the smart crack detection preliminarily (Fukuhara, Terada, Nagao, Kasahara, & Ichihashi, 1990). However, this system cannot judge the types of cracks and can only be used in the night. University of Arkansas developed a real-time crack measurement system (digital highway data vehicle) (Luo, Li, & Wang, 2016), collecting data and recognising

*Corresponding author. Email: wangzhenjun029@163.com

crack at a high speed. But this system needs an auxiliary of supercomputer. Bursanescu et al. scanned the pavement by laser sensors and drew 3D images to analyse crack damage information without shadow interference, while the disadvantages of this method are the large amount of data and high demand for hardware equipment (Bursanescu & Blais, 1997). Although the above methods can calculate the crack length, due to accuracy and hardware, all of these have not been used in engineering widely.

With the development of computing technology, the convolutional neural network (Barat & Ducottet, 2016; Leng, Guo, Zhang, & Xiong, 2015; Shi, Bai, & Yao, 2016; Xu, Luo, Wang, Gilmore, & Madabhushi, 2016) and deep learning (Dong, Liu, & Lian, 2016; Shu et al., 2016) have advantages in the field of image recognition. Deep learning is a perception with hidden layers, which groups low-level features to form abstract high-level features to find the characteristics of the distributed data. The convolutional neural network is a type of artificial neural networks, whose network structure of shared weights reduces the complexity of network models and the number of parameters is similar to biological neural networks. Images are used as the input data of the network directly avoiding the complex traditional recognition algorithm in the feature extraction and data reconstruction. The network structure has high invariance in the transformation of translation, scaling, tilting and so on (Ijjina & Chalavadi, 2016; Liu, Zhang, & Liu, 2016; Xu, Zhu, Wong, & Fang, 2016). By introducing deep learning and the network to the field of the pavement health monitoring and road maintenance, instead of manual measurement, the accuracy and the efficiency of length recognition are increased without high hardware and human cost. However, these are little related research.

Therefore, as a technological means, attempts have been made to employ the DCNN method to provide an appropriate model for recognition of the crack length in asphalt pavements. The processes of accomplishing DCNN included the structure designing, training and testing of DCNN. In this work, to increase the robustness of DCNN, crack images of asphalt pavement were preprocessed according to grey-scale characteristics of pavements and cracks. The neural networks model with initialisation parameters was developed in the structure designing of DCNN. The training of DCNN included pre-train processes and fine-tune processes, which were designed based on deep learning to give the DCNN ability of recognising crack length. Meanwhile, stochastic gradient descent (SGD) was adopted to speed up the training rate in the feed-forward algorithm in the training of DCNN. At last, 8000 images were inputted into the DCNN to calculate the crack length. The result was compared with the result of the actual measurement to verify the accuracy of the DCNN. Using the above methods to structure, train and test the DCNN, automatic recognition of pavement crack length in batches had been achieved.

2. Acquisition and preprocessing of crack images

2.1. *Crack datasets*

To ensure that the ratio between the crack length in images and the actual length is 1:2 in this work, the camera lens was kept perpendicular to pavements; and the distance between the camera and the pavements was invariant in acquiring images. Image capture instrument is shown in Figure 1. The camera was installed in a holder, so the operator only controls the switch to use the continuous capture mode in the collection. To get crack images in different light conditions, we collected images on both sunny and cloudy days.

Original images of cracks were divided into 200 $\times$ 200 images and found to be equal to the size of the DCNN input layer. Seven thousand and five hundred divided images with cracks and 500 divided images without crack were selected as the training sample. To ensure the integrity of the sample, all lengths of cracks should be included in the training sample. To evaluate one

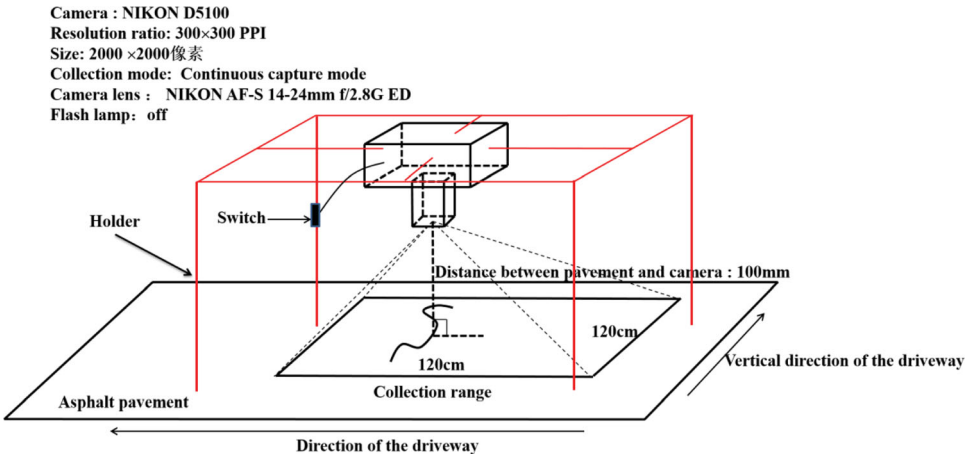

Figure 1. Image capture instrument.

iteration of training, 500 images were randomly selected from the training sample as the testing sample. Notably, 500 images did not take part in this iteration to guarantee the independence between the training sample and the testing sample. Meanwhile, we draw a curve according to crack length manually. The widths of lines were all only 1 pixel. So the lengths of crack in images were the numbers of pixels in these lines. Considering the ratio of 1:2, the actual lengths were acquired. The actual lengths were utilised as target sample for DCNN in the training and the standard results in the testing.

2.2. *Extract crack features*

Collected original images were red, green and blue images in jpg format, including colour information, which increases the difficult of feature detection and recognition of crack characteristics. Considering computer hardware, the colour information was changed and images were saved as grey-scale pictures in bmp format.

Grey values of cracks and pavement in grey-scale map are obviously different, so it is feasible to extract crack features by utilising the difference. One thousand grey histograms based on part of the training sample were drawn by MATLAB. Four histograms are shown in Figure 2, all of which have two obvious peaks, peak 1 and peak 2. It is generally believed that the clustering centre of each peak can be used to distinguish the segmentation value of each composition (Bona, Borba, Benetti, Duan, & Griggs, 2013). One thousand histograms were adopted to do k-means clustering analysis to calculate clustering centres, which were used to determinate the gray-scale ranges of components. The average value of peak 1 was 80.59 and that of peak 2 was 151.81. The grey-scale range of cracks and pavement was adjusted based on the clustering centre and was confirmed in ranges of 50–110 and 110–250,00, respectively.

Grey-scale maps were extracted to ensure that only the information about crack length and shape were saved in images, as the following steps: (1) original images (Figure 3(a)) of cracks were divided into 200 × 200 images as Figure 3(b). (2) The colour information was changed and 200 × 200 images were saved as grey-scale pictures in bmp format as Figure 3(c). (3) Grey-scale range was cut to highlight the information of length and shape. The result of an image is shown in Figure 3(d). (4) Grey-scale range of 50.05–110.14 was inputted to divide the images. The result of an image is shown in Figure 3(e). (5) The result was saved as a grey-scale map in bmp format.

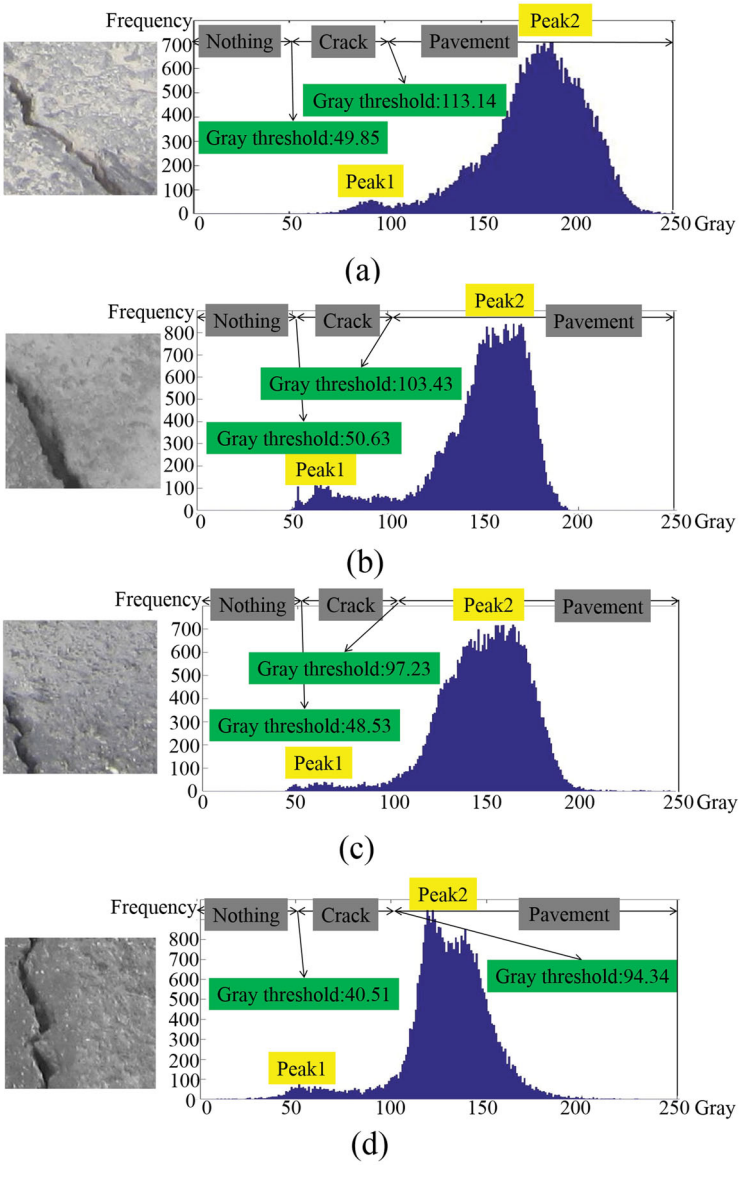

Figure 2. Gray-scale images of crack.

The extracting process of crack images is shown in Figure 3. Comparing Figure 3(b) with (d), the crack features are shown clearly by dividing the images with the grey-scale range. Therefore, the grey-scale range was used as the gist of extraction in batches (some isolate black spots need no't be disposed).

3. Deep convolutional neural (DCNN) network

To handle the cracks' complex shapes and sizes in images, the convolutional neural network with deep learning was adopted in this work. The processes of developing the DCNN to recognise the

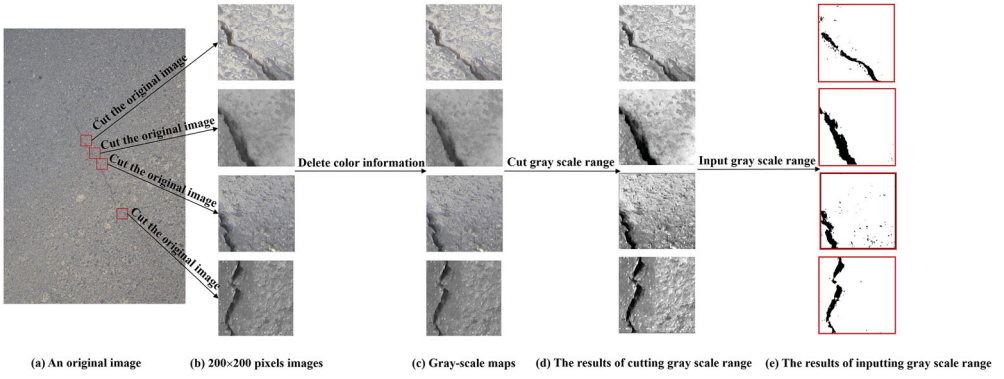

Figure 3. Extracting process of crack images.

crack length are shown in Figure 4. The process mainly includes the structure of DCNN, the training of DCNN and the testing of DCNN. Remarkably, if the result of testing cannot satisfy the demand, we should structure and train the DCNN all over again.

3.1. *Structure designs of DCNN*

The convolutional neural network is a widely used type of deep neural network structure. Fukushima put forward neocognitron which could be considered as one origin of CNN (Fukushima, 1984). Then Lecun et al. developed this structure and utilised it to recognise 2D images and 3D human movement (LeCun, Bengio, & Hinton, 2015). Now CNN is mainly used to recognise 2D and 3D images (Hu, Chang, Nian, Wang, & Li, 2016; Nian, Li, Wang, Xu, & Wu, 2016), which has high invariance in the transformation of translation, tilting and so on. This property is used to solve the problem that crack's displacement and shape change irregularly in the length recognition. Meanwhile, images as the multi-dimensional vectors are inputted into convolutional neural networks, which decreases the complexity of feature extraction and data classification in the process of reconstruction effectively. In addition, due to the same neuron weights in the same feature map, network parallel learning is realised, which is also a big advantage of CNN, compared with fully connect neural network.

Crack length recognition of the DCNN is composed of two convolutional layers (C1 and C2) and two subsampling layers (S1 and S2), followed by two fully connected layers (FC1 and FC2) and the output layer, respectively. The details of DCNN are shown in Figure 5 and Table 1. Notably, the type of pooling used in two subsampling layers was max-pooling. Using

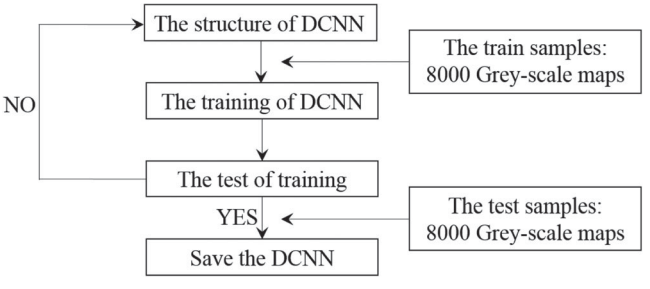

Figure 4. Process of DCNN.

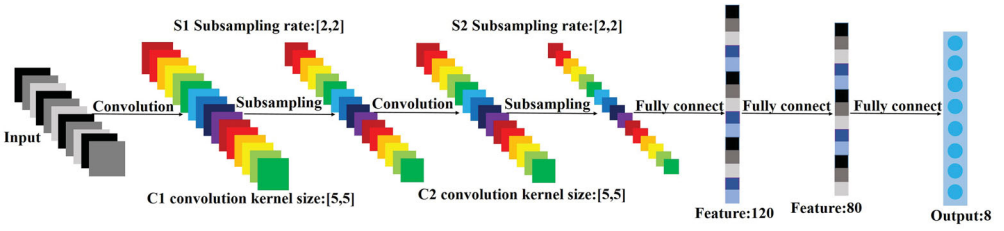

Figure 5. Structure of DCNN.

Table 1. Structure indexes of DCNN.

Layers	Filters	Channels	Size	Stride
C1	16	2	[5,5]	1
S1	N/A	16	[2,2]	2
C2	16	4	[5,5]	1
S2	N/A	16	[2,2]	2

max-pooling, the maximal values in 2 × 2 submatrices of convolutional maps were obtained as a result.

3.2. *Deep learning*

Deep learning has attracted great attention in recent years and has been widely used in the image recognition field with good performance (Guo, Wang, Lei, Tu, & Li, 2016; Zhao & Du, 2016). It is also robust to represent crack information, including length. However, training the DCNN is a challenging task because it largely relies on the parameters across layers (Bengio, Courville, & Vincent, 2013). Hinton et al. proposed a stacked layer-wise training method for training deep belief networks (Hinton & Salakhutdinov, 2006). It is a type of popular deep architecture. In their work, well-trained deep belief networks are used for recognising hand-written digits. It shows strong data representation ability of the deep architecture. Now deep learning methods are effective in practice, including fields of object recognition, natural language processing and speech recognition (Krizhevsky, Sutskever, & Hinton, 2012; Larochelle, Mandel, Pascanu, & Bengio, 2012). It is the regularisation function that guides the learning towards basins of attraction of minima that support better generalisation from training dataset. The greedy strategy does not always get the optimal answers (Goodfellow, Courville & Bengio, 2013). Recent studies focus on training deep hierarchy models jointly, expecting to find a better way to train the deep models.

Deep learning was used combined with the convolutional neural network to recognise the crack length of input images. The bounding box of crack and the key points (continuous black spots) were detected first. Crack images were aligned by similarity transformation with key points. Current available crack estimation datasets did not contain enough labelled images to train the DCNN. To overcome this problem, two steps (fine-tune process and pre-train process) were taken to train the DCNN as shown in Figure 6. In the pre-train process, the randomly initialised network was first trained by a related task using images owning enough labels, which were blue boxes as shown in Figure 6. In the fine-tune process, parameters learnt in the pre-train process were used as initialisation for the new task. In the DCNN, networks for a crack identification task were tainted first using images owning enough labels, which were blue boxes

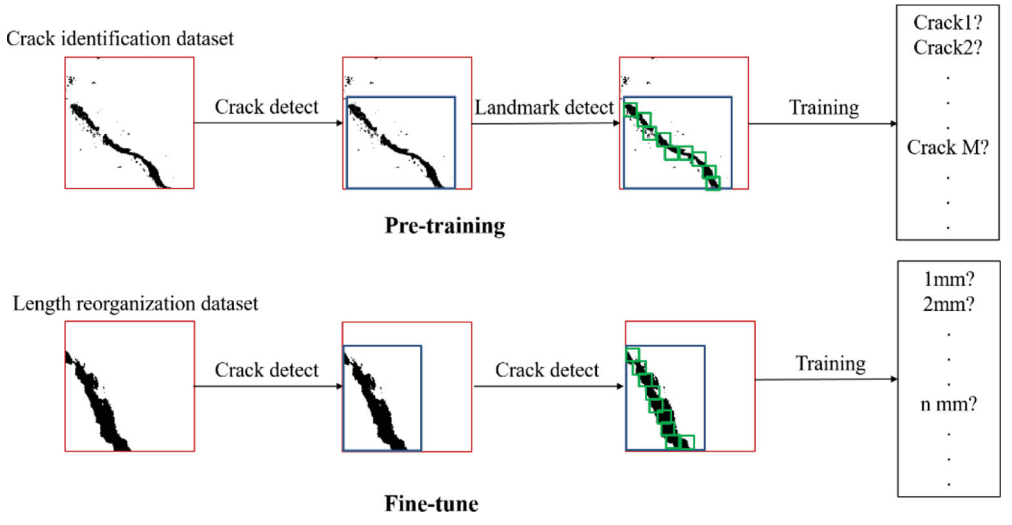

Figure 6. Diagram of deep learning.

Table 2. Classes of images.

Length range (cm)	number	Length range (cm)	number
0–1	800	4–5	1100
1–2	500	5–6	1500
2–3	1000	6–7	1300
3–4	1200	7–8	600

as shown in Figure 6. DCNN trained by the crack identification task was able to extract features. Later the parameters learnt in the pre-train process were fine tuned for the task of length reorganisation.

Due to the deep learning strategy, the DCNN was trained for the two crack identification tasks and length classification task. Thus, two datasets for the two different tasks were used. In the crack identification task (the pre-train process), a dataset was generated containing 1200 images, which was randomly selected from 8000 images. In the length identification task (the fine-tune process), the DCNN was trained based on all of the 8000 images. Crack images were assigned to eight classes as shown in Table 2. The above-mentioned processes were realised based on Caffe in the condition of Inter(R) Core(TM) i7-6700 CPU, 8.00GB Random Access Memory (RAM) and NVIDA GeForce GTX 1060 6GB GPU.

3.3. *Training of DCNN*

An important property of DCNN is its learning ability used to find the mapping between input data (8000 grey-scale maps) and output data (the crack length corresponding with images). The mapping was realised by the training of DCNN. The main job was to confirm the weights and bias. Before the training of DCNN, we quadrature encoded the target data to improve the rate of response speed and the training convergence. Then the feed-forward algorithm was used to train the DCNN, while SGD was adopted to reduce the calculation amount and accelerate the convergence of training.

3.3.1. *Quadrature encoding*

Quadrature encoding has advantages of high precision and fast response speed. The target data are transformed into the form of quadrature encoding, making count convenient and the training convergence rate faster.

The target data of the DCNN are the crack length corresponding with images, whose measured accuracy of length is centimetre. Generally, the crack length in a 200 × 200 pixels' image acquired by the photography method described above was less than 7 cm. Therefore, the target data matrixes in both pre-train and fine-tune process were transformed from 1 × 1000 to 8 × 1000. The sketch map is shown in Figure 7.

3.3.2. *Feed-forward algorithm*

The training in both the fine-tune process and pre-train process is the adjustment of weights and biases to obtain output through applying a proper method. The supervised methods are the most general methods for training. The mechanism of the supervised methods is to use the algorithm, the least mean square method and its simplification to multilayer networks, which is the feed-forward algorithm (Shi, Schillings, & Boyd, 2004; Zhang, Fu, Jiang, Liu, & Lv, 2015). Training of the DCNN takes place in the feed-forward algorithm in order to decrease the error between the DCNN output and the target data. The feed-forward algorithm begins with adjusting initial random values for weights and biases in the pre-train process, and getting values of weights and biases from DCNN that had accomplished the pre-train process in the fine-tune process. After importing input images, the feed-forward algorithm of the intermediate results led to producing the output vectors. Then the errors were calculated by the difference between the target data and the DCNN output. By the feed-forward of the error in the DCNN, the weights and biases were modified to decrease the error in the subsequent cycle of prediction (Prasad, Eskandari, & Reddy, 2009; Shafabakhsh, Ani, & Talebsafa, 2015; Xiao, Amirkhanian, & Juang, 2009). The flowchart describing the development of training using feed-forward is shown in Figure 8.

3.3.3. *Stochastic gradient descent*

Gradient descent (GD) is a common method to confirm the neural network's weights and bias (Erdal & Khanesar, 2016). GD is a common method to minimise the risk function and the damage function (Senov, 2015), including batch gradient descent (BGD) and SGD (Clemencon, Bellet, Jelassi, & Papa, 2015; Shi, Si, Feng, & Zhang, 2016; Sopyla & Drozda, 2015). BGD is to minimise the damage function of the whole training sample, so the ultimate solution is the global optimal solution. In the DCNN of this paper, 8000 images were inputted together to calculate, each of whose iterative calculation amount was 8000 × 2^2. SGD is used to minimise the damage function of a part of training samples. Although one iteration does not always make the damage

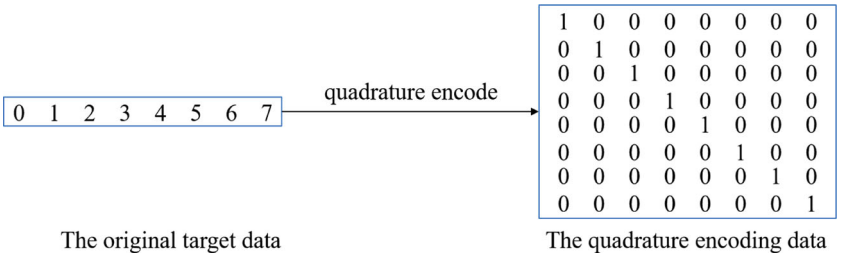

Figure 7. Sketch of quadrature encoding.

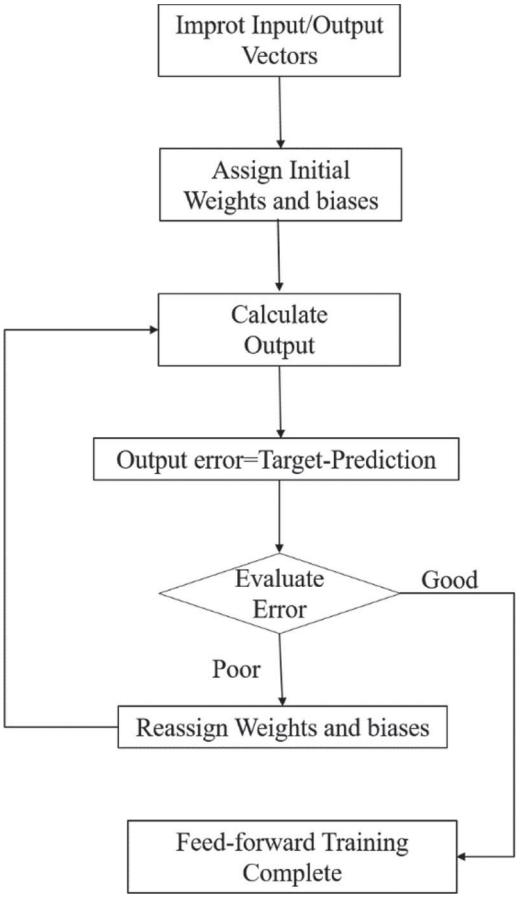

Figure 8. Flowchart of feed-forward algorithm.

function towards the global optimal solution, the overall direction is to the global optimal solution and the result is the near global optimal solution. In the DCNN, only 100 images were inputted to calculate in an integration, whose calculation amount is 100×2^2. Obviously, SGD was adopted under the condition of satisfying the calculation accuracy, reducing the calculation amount and accelerating the convergence of training.

SGD was adopted to calculate the damage function of training sample $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(m)}, y^{(m)})\}$. $(x^{(m)}, y^{(m)})$ stood a grey-scale map gotten from Section 2.2 as shown in Figure 3(e), following formula (1). The first term is mean square error reflecting the gap between predicted values and real sample values, while the second is the regularisation term used to reduce the range ability of weights and bias to prevent over fitting, and ε is tradeoff control parameter.

$$J(W|b) = \left[\frac{1}{m}\sum_{i=1}^{m} J(W|b;\, x^{(i)}|y^{(i)})\right] + \frac{\varepsilon}{2}\sum_{l=1}^{n_{l+1}}\sum_{i=1}^{S_l}\sum_{j=1}^{S_{l+1}}(W_{ji}^{l})^2. \tag{1}$$

The aim at training is to minimise the damage function. The gradient of the $J(W, b)$ function about W and b is iterated to minimise the damage function of 100 images. Function W_{ij} and function b_i (α is learning rate) can be calculated by formulas (2) and (3). Notably, we used the

values of α 0.1, 1.0, 2.0 and 3.0 in the training of length identification task to find an appropriate learning rate. And the range of α was 0.1–3.0.

$$W_{ij}^{l} = W_{ij}^{l} - \alpha \frac{\partial}{\partial W_{ij}^{l}} J(W|b), \tag{2}$$

$$b_{i}^{(l)} = b_{i}^{(l)} - \alpha \frac{\partial}{\partial b_{i}^{(l)}} J(W|b). \tag{3}$$

3.4. *Results of training*

3.4.1. *Performance of crack identification task*

DCNN was first trained by the crack identification dataset to transfer the weights and the bias for the length classification task. The crack identification task was a multi-class classification task. It was evaluated by error rate of classification. Figure 9 shows the error rate of DCNN in the pre-train process. The horizontal axis represents the number of the iteration. The vertical axis represents the DCNN's error rate in corresponding iteration.

The trained DCNN in the crack identification task was able to extract crack features from images. And parameters learnt in this task were used as initialisation of the length estimation task.

3.4.2. *Performance of length identification task*

In the length classification task, the DCNN trained in the crack identification task was fine tuned using the crack length database. The length identification error rate of the DCNN is presented in Figure 10. Obviously, DCNN using learning rate 0.1 and 1.0 showed a higher error rate than 2.0. And the learning rate of 3.0 was a high rate which led to the error rate increasing and DCNN over fitting. So learning rate 2 was appropriate for the training in the length identification task; and Table 3 shows the DCNN performance through the confusion matrix using learning rate 2. The total number of each column is corresponding to Table 2. Each row shows the result of length recognition of 8000 images. In the length recognition and test based on the well-trained DCNN, one image uses about 3 ms in the PC, which has Intel(R) Core(TM) i5-2520M CPU, 4.00GB

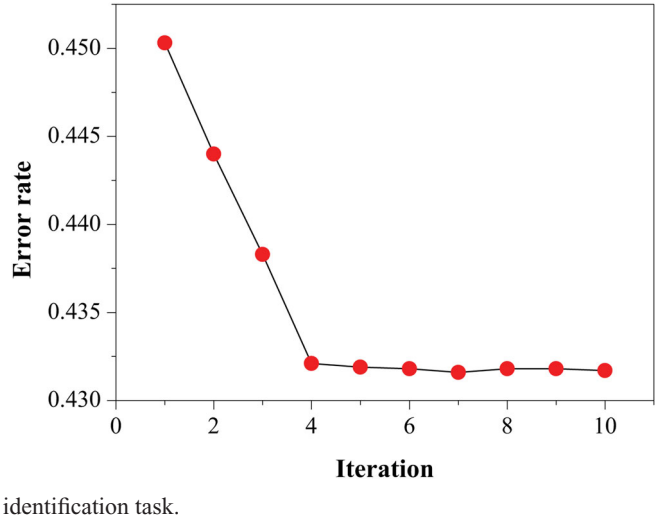

Figure 9. Crack identification task.

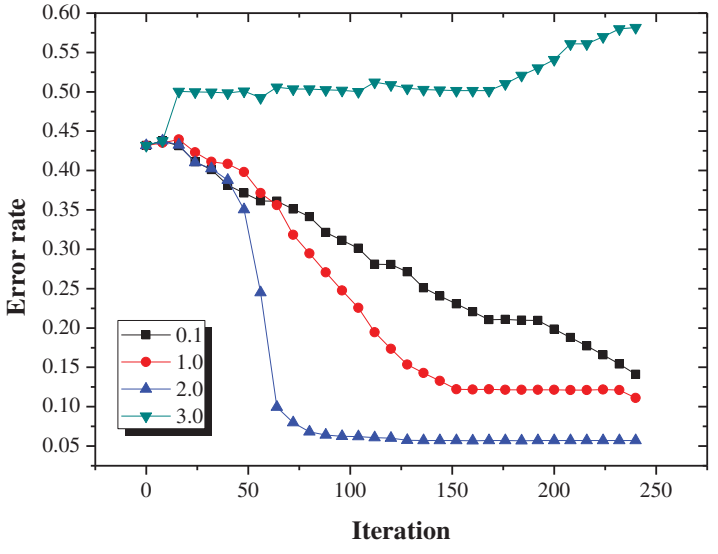

Figure 10. Length identification task.

Table 3. Confusion matrixes of testing.

	0–1 cm	1–2 cm	2–3 cm	3–4 cm	4–5 cm	5–6 cm	6–7 cm	7–8 cm
0–1 cm	799							
1–2 cm	1	490	23					
2–3 cm		10	949	19				
3–4 cm			28	1138	13			
4–5 cm				43	1033	17		
5–6 cm					54	1450	41	
6–7 cm						33	1190	101
7–8 cm							69	499
Total	800	500	1000	1200	1100	1500	1300	600
Error	1	10	51	62	67	50	110	101
Error rate (%)	0.12	2.00	5.10	5.17	6.09	3.33	8.46	16.83

RAM and NVIDIA GeForce GT 630. Obviously, the length recognition based on DCNN is efficient. So the developed DCNN showed the improvement that it has lower requirement for hardware to analyse crack analysis with high efficiency.

As Table 3 and Figure 10 show, the accuracy of DCNN is 94.35% and mean squared error (MSE) is 0.2377 following formulas (4) and (5), while the max error of fault length recognition is controlled in the context of 1 cm meeting the demand of pavement crack length detection. However, the length ranges of 6–7 cm and 7–8 cm have a bigger error rate than other ranges. So more crack images whose length is between 6–8 cm should be used to improve the accuracy of the DCNN in future

$$\text{The accuracy of DCNN} = \frac{\text{The number of the correct results}}{\text{The total number of images}}, \tag{4}$$

$$\text{MSA} = \frac{1}{n}\sum_{i=1}^{n}(\text{target} - \text{predicted})^2. \tag{5}$$

3.5. *Reliability analysis of the developed DCNN*

3.5.1. *Resolution analysis of the developed DCNN*

The image resolution should be considered after developing DCNN. Five different pixels per inch (PPI) 36 PPI, 54 PPI, 72 PPI, 180 PPI, 300 PPI were used to evaluate the sensitivity to image resolution. Notably, the resolution of images acquired from the instruction in Figure 1 was 300 PPI. Then the PPI of 8000 images was changed to 36 PPI, 54 PPI, 72 PPI, 180 PPI, respectively, to realise the aim of sensitivity analysis to image resolution and ensure the rationality of comparison among the five datasets.

The five datasets were imported to develop DCNN. The MSE of the five datasets 36 PPI, 54 PPI, 72 PPI, 180 PPI, 300 PPI were 0.3128, 0.2443, 0.2431, 0.2412, 0.2377, respectively. Therefore, image resolution had a non-significant influence on the accuracy of the developed DCNN in the range of 54–300 PPI. However, low image resolution, such as 36 PPI, had a bad effect on DCNN. So image resolution should be guaranteed higher than 54 PPI in service of CNN.

3.5.2. *Light analysis of the developed DCNN*

Light conditions should be considered after developing DCNN. Eight thousand images were collected in different light conditions. Three images with different light conditions are shown in Figure 11(a). The 8000 images were divided by the conditions under which they were collected either sunny or cloudy day. The results of the two different light conditions are shown in Tables 4 and 5. The results showed that there was no significant influence on the accuracy of the developed DCNN. So in our method using DCNN, crack analysis can be done both in the day and night without having to consider weather situations while essential illumination should be guaranteed in the night for digital imaging. The reason why the light condition had no significant influence on the work of DCNN is that the influence of lights was removed primarily in the process of extracting crack features by k-means. The main influence of lights was on the widths of cracks as shown in Figure 11(b) and some isolate black spots. However, the width changes of cracks had little influence on the DCNN's work. And these isolate black spots were wiped off by the convolutional layers.

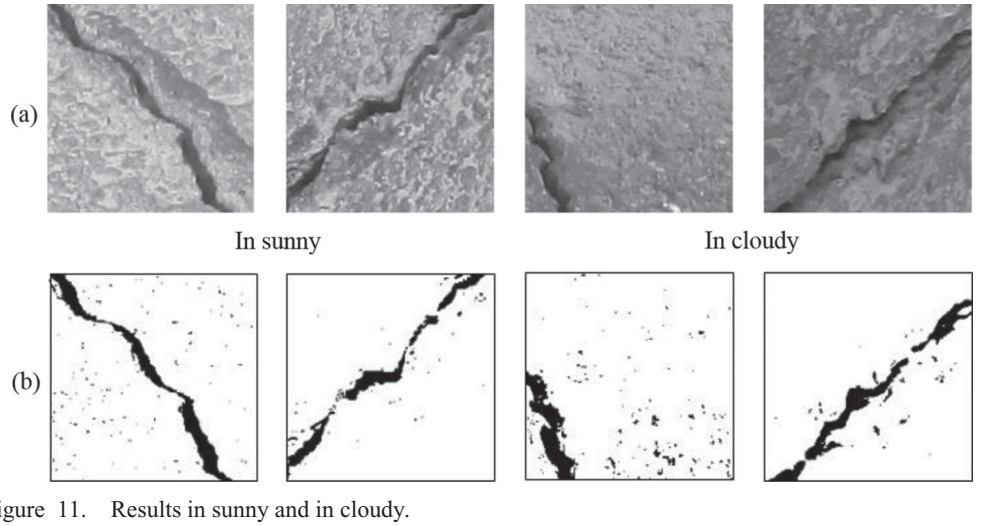

Figure 11. Results in sunny and in cloudy.

Table 4. Confusion matrixes of crack images in sunny.

	0–1 cm	1–2 cm	2–3 cm	3–4 cm	4–5 cm	5–6 cm	6–7 cm	7–8 cm
0–1 cm	356							
1–2 cm		280	9					
2–3 cm		3	595	5				
3–4 cm			20	701	7			
4–5 cm				30	450	6		
5–6 cm					27	742	21	
6–7 cm						16	635	72
7–8 cm							38	364
Total	356	280	624	736	484	764	694	436
Error	0	3	29	35	34	22	59	72
Error rate (%)	0	1.07	4.65	4.76	7.02	2.88	8.50	16.51

Table 5. Confusion matrixes of crack images in cloudy.

	0–1 cm	1–2 cm	2–3 cm	3–4 cm	4–5 cm	5–6 cm	6–7 cm	7–8 cm
0–1 cm	443							
1–2 cm	1	220	14					
2–3 cm		4	354	14				
3–4 cm			8	437	6			
4–5 cm				13	583	11		
5–6 cm					27	708	20	
6–7 cm						17	553	29
7–8 cm							31	135
Total	444	220	376	464	616	736	604	164
Error	1	4	22	27	33	28	51	29
Error rate (%)	0.22	1.82	5.85	5.82	5.36	3.80	8.44	17.68

4. Application in pavement

Developed DCNN was used to measure cracks in practice after the above processes. Four highways in central China were detected. The pavement materials were SBS Asphalt Concrete (AC-16C), Asphalt Concrete (AC-16C), Asphalt Concrete(AC-16) and Stone mastic asphalt (SMA-13) in four highways separately. Forty images from each road were acquired using the above-mentioned collection method and image capture instrument. Forty images were transformed into grey-scale maps and divided into 200 × 200 pixels. Grey-scale maps of 487,200 × 200 pixels with cracks and 125 grey-scale maps without cracks were imported into the DCNN and the lengths of cracks were calculated.

The detection results are shown in Table 6. The results showed that the accuracies of DCNN were 0.2318 following formulas (5), while the max error of fault length recognition was controlled in the context of 1 cm meeting the demand of pavement crack length detection. However, the length ranges 7–8 cm have a bigger error rate than other ranges. All of the results showed that the developed DCNN can be transferrable to other asphalt pavements with similar accuracy of the results in the length identification task, and pavement materials' textures had little influence on the developed DCNN.

5. Conclusions

In this work, recognition of asphalt pavement crack length using DCNN networks was conducted. The following conclusions can be drawn:

Table 6. MSE results of DCNN application in pavement.

	Highway1	Highway2	Highway3	Highway4
0–1 cm	0.2315	0.2218	0.2104	0.2004
1–2 cm	0.2291	0.2301	0.2298	0.2135
2–3 cm	0.2234	0.2279	0.2301	0.2238
3–4 cm	0.2126	0.2341	0.2334	0.2358
4–5 cm	0.2198	0.2316	0.2297	0.2324
5–6 cm	0.2137	0.2356	0.2241	0.2335
6–7 cm	0.2310	0.2407	0.2218	0.2211
7–8 cm	0.2613	0.2830	0.2735	0.2771
Average	0.2278	0.2381	0.2316	0.2297

(a) Networks trained in crack task are able to extract discriminative crack features. The trained network was further fine tuned in the length identification task by crack labelled images to recognise crack length. The training strategy based on the two processes overcomes the lack of crack labelled images and improves the accuracy of DCNN, combining with quadrature encoding and SGD.
(b) One image uses about 3 ms in the PC, the accuracy of DCNN is 94.35% and MSE is 0.2377 in the length recognition and test based on the well-trained DCNN. Therefore, the networks can measure the crack length accurately and efficiently.
(c) The length ranges 6–7 cm and 7–8 cm have a bigger error rate than the other ranges. So more crack images whose length is between 6 and 8 cm should be used to improve the accuracy of the DCNN in future.
(d) Image resolution in the range of 54–300 PPI had a non-significant influence on the accuracy of developed DCNN. And light conditions showed little significant influence on the accuracy of developed DCNN for using the k-means and the convolutional layers.
(e) The application in central China showed that the developed DCNN could be transferrable to other asphalt pavements with similar accuracy of the results in length identification.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

The authors thank the supports from the Fundamental Research Funds for the Central Universities of China [No. 310831153504 and 310831163113] and National and Local Joint Engineering Materials Laboratory of Traffic Engineering and Civil Engineering, Chongqing Jiaotong University [No: LHSYS-2016-002].

References

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., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(8), 1798–1828. doi:10.1109/TPAMI.2013.50

Bona, A. D., Borba, M., Benetti, P., Duan, Y., & Griggs, J. A. (2013). Three-dimensional finite element modelling of all-ceramic restorations based on micro-CT. *Journal of Dentistry*, *41*(5), 412–419. doi:10.1016/j.jdent.2013.02.014

Bursanescu, L., & Blais, F. (1997). Automated pavement distress data collection and analysis: A 3-D approach. *Digital Identity Management*, 311–317. doi:10.1109/IM.1997.603881

Clemencon, S., Bellet, A., Jelassi, O., & Papa, G. (2015). Scalability of stochastic gradient descent based on "smart" sampling techniques. *Procedia Computer Science*, *53*, 308–315. doi:10.1016/j.procs.2015.07.308

Dong, Y., Liu, Y., & Lian, S. (2016). Automatic age estimation based on deep learning algorithm. *Neurocomputing*, *187*, 4–10. doi:10.1016/j.neucom.2015.09.115

Erdal, K., & Khanesar, M. A. (2016). Chapter 5 – gradient descent methods for type-2 fuzzy neural networks. *Fuzzy Neural Networks for Real Time Control Applications, Butterworth-Heinemann*, 45–70. doi:10.1016/B978-0-12-802687-8.00005-0

Fukuhara, T., Terada, K., Nagao, M., Kasahara, A., & Ichihashi, S. (1990). Automatic pavement – distress – survey system. *Journal of Transportation Engineering-ASCE*, *116*(3), 280–286. doi:10.1061/(ASCE)0733-947X(1990)116:3(280)

Fukushima, K. (1984). Pattern recognition and the neocognitron. *Journal of the Institute of Television Engineers of Japan*, *38*(3), 212–218. Retrieved from https://www.jstage.jst.go.jp/article/itej1978/38/3/38_3_212/_article

Goodfellow, I. J., Courville, A., & Bengio, Y. (2013). *Joint training deep Boltzmann machines for classification*. Computer Science, May 1, 1–11.

Guo, Q., Wang, F., Lei, J., Tu, D., & Li, G. (2016). Convolutional feature learning and hybrid CNN-HMM for scene number recognition. *Neurocomputing*, *184*(4), 78–90. doi:10.1016/j.neucom.2015.07.135

Hinton, G. E., & Salakhutdinov, R. (2006). Reducing the dimensionality of data with neural networks. *Science*, *313*(5786), 504–507. doi:10.1126/science.1127647

Hu, Y., Chang, H., Nian, F., Wang, Y., & Li, T. (2016). Dense crowd counting from still images with convolutional neural networks. *Journal of Visual Communication and Image Representation*, *38*, 530–539. doi:10.1016/j.jvcir.2016.03.021

Ijjina, E. P., & Chalavadi, K. M. (2016). Human action recognition using genetic algorithms and convolutional neural networks. *Pattern Recognition*, *59*, 199–212. doi:10.1016/j.patcog.2016.01.012

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). *Imagenet classification with deep convolutional neural networks*. 26th annual conference on neural information processing systems (pp. 1097–1105). Lake Tahoe, NV.

Larochelle, H., Mandel, M. I., Pascanu, R., & Bengio, Y. (2012). Learning algorithms for the classification restricted Boltzmann machine. *Journal of Machine Learning Research*, *13*(1), 643–669.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. doi:10.1038/nature14539

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

Liu, Z., Zhang, J., & Liu, L. (2016). Upright orientation of 3D shapes with convolutional networks. *Graphical Models Graphical Models Vgraphical Models and Image Processing Vcomputer Vision, Graphics, and Image Processing*, *85*, 22–29. doi:10.1016/j.gmod.2016.03.001

Luo, W., Li, L., & Wang, K. C. P. (2016). Automated pavement horizontal curve measurement methods based on inertial measurement unit and 3D profiling data. *Journal of Traffic and Transportation Engineering (English Edition)*, *3*(2), 137–145. doi:16/j.jtte.2016.03.004

Nian, F., Li, T., Wang, Y., Xu, M., & Wu, J. (2016). Pornographic image detection utilizing deep convolutional neural networks. *Neurocomputing*, *210*, 283–293. doi:10.1016/j.neucom.2015.09.135

Prasad, B. K., Eskandari, H., & Reddy, B. V. (2009). Prediction of compressive strength of SCC and HPC with high volume Fly ash using ANN. *Construction and Building Materials*, *23*(1), 117–128. doi:10.1016/j.conbuildmat.2008.01.014

Senov, A. A. (2015). Improving distributed stochastic gradient descent estimate via loss function approximation. *IFAC-PapersOnLine*, *48*(25), 292–297. doi:10.1016/j.ifacol.2015.11.103

Shafabakhsh, G. H., Ani, O. J., & Talebsafa, M. (2015). Artificial neural network modeling (ANN) for predicting rutting performance of nano-modified hot-mix asphalt mixtures containing steel slag aggregates. *Construction and Building Materials*, *85*, 136–143. doi:10.1016/j.conbuildmat.2015.03.060

Shi, B., Bai, X., & Yao, C. (2016). Script identification in the wild via discriminative convolutional neural network. *Pattern Recognition*, *52*, 448–458. doi:10.1016/j.patcog.2015.11.005

Shi, X. L., Schillings, P., & Boyd, D. W. (2004). Applying artificial neural networks and virtual experimental design to quality improvement of two industrial processes. *International Journal of Production Research*, *42*(1), 101–118. doi:10.1080/00207540310001602937

Shi, Y., Si, L., Feng, G., & Zhang, J. (2016). Numerical and experimental study on liquid crystal optical phased array beam steering combined with stochastic parallel gradient descent algorithm. *Optik – International Journal for Light and Electron Optics*, *127*(3), 1450–1454. doi:10.1016/j.ijleo.2015.11.023

Shu, Z., Qi, C., Xin, S., Hu, C., Wang, L., Zhang, Y., & Liu, L. (2016). Unsupervised 3D shape segmentation and co-segmentation via deep learning. *Computer Aided Geometric Desing*, *43*(3), 39–52. doi:10.1016/j.cagd.2016.02.015

Sopyla, K., & Drozda, P. (2015). Stochastic gradient descent with Barzilai-Borwein update step for SVM. *Information Sciences*, *316*, 218–233. doi:10.1016/j.ins.2015.03.073

Xiao, F., Amirkhanian, S. N., & Juang, C. H. (2009). Prediction of fatigue life of rubberized asphalt concrete mixtures containing reclaimed asphalt pavement using artificial neural networks. *Journal of Materials in Civil Engineering*, *21*(6), 253–261. doi:10.1061/(ASCE)0899-1561(2009)21:6(253)

Xu, J., Luo, X., Wang, G., Gilmore, H., & Madabhushi, A. (2016). A deep convolutional neural network for segmenting and classifying epithelial and stromal regions in histopathological images. *Neurocomputing*, *191*(5), 214–223. doi:10.1016/j.neucom.2016.01.034

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

Zhang, H., Fu, X., Jiang, H., Liu, X., & Lv, L. (2015). The relationships between asphalt ageing in lab and field based on the neural network. *Road Materials and Pavement Design*, *16*(2), 493–504. doi:10.1080/14680629.2015.1020846

Zhao, W., & Du, S. (2016). Learning multiscale and deep representations for classifying remotely sensed imagery. *Isprs Journal of Photogrammetry and Remote Sensing*, *113*, 155–165. doi:10.1016/j.isprsjprs.2016.01.004