SciencePG
Science Publishing Group

Research Article

Automatic Road Crack Detection Using Convolutional Neural Network Based on Semi-Supervised Learning

Jun Yang[1] ⓘ, Xiaoli Sun[1] ⓘ, Shuai Teng[2, *] ⓘ

[1]Technology Research and Development Department, Guangzhou Municipal Engineering Testing Co., Ltd., Guangzhou, The People's Republic of China

[2]Research Centre for Wind Engineering and Engineering Vibration, Guangzhou University, Guangzhou, The People's Republic of China

Abstract

Crack detection in pavements is a critical task for infrastructure maintenance, but it often requires extensive manual labeling of training samples, which is both time-consuming and labor-intensive. To address this challenge, this paper proposes a semi-supervised learning approach based on a DenseNet classification model to detect pavement cracks more efficiently. The primary objective is to leverage a small set of labeled samples to improve the model's performance by incorporating a large number of unlabeled samples through semi-supervised learning. This method enhances the DenseNet model's ability to generalize by iteratively learning from new unlabeled datasets. As a result, the proposed approach not only reduces the need for extensive manual labeling but also mitigates issues related to label inconsistency and errors in the original labels. The experimental results demonstrate that the semi-supervised DenseNet model achieves a prediction precision of 96.77% and a recall of 94.17%, with an F1 score of 95.45% and an Intersectidn over Union (IoU) of 91.30%. These metrics highlight the model's high accuracy and effectiveness in crack detection. The proposed method not only improves label quality and model performance but also offers practical value for engineering applications in the field of pavement maintenance, making it a valuable tool for infrastructure management.

Keywords

Crack Detection, Convolutional Neural Network (CNN), Semi-Supervised Learning, Pseudo-Labelling

1. Introduction

The fatigue stress caused by periodic loading and long-term bad environmental conditions will lead to structural degradation of a highway, and ultimately shorten its service life, and even affect the highway traffic safety [1, 2]. Therefore, highway disease detection, especially crack detection, has become a main goal of road maintenance. In the process of pavement crack detection, the traditional manual measurement and other manual detection methods are time-consuming, subjective and unprecise, which is not conducive to road maintenance. Therefore, some vision-based techniques have been gradually introduced in crack detection to partially replace manual on-site inspection. These techniques include traditional image segmentation methods, such as the threshold-based [3-5], wavelet trans-

form-based [6], edge detection-based [7], seed growth-based [8], and region-based segmentation method [9, 10], etc. The traditional crack segmentation method is widely used in a scene where the road image is simple and does not contain other redundant objects. However, the traditional image processing methods based on specific rules usually encounter noise problems of different severity, such as the road signs, manhole covers, shadows, etc., thus noise elimination and crack extraction are needed, which usually involves complex post-processing procedures and many manually adjustable parameters. They are time-consuming, unstable and impractical; therefore, the scope of their uses is limited. Thus, the machine learning algorithms with knowledge learning as the core are gradually applied to the crack detection.

Many researches on crack detection first use traditional image processing technology to extract key features, and then combine them with the machine learning (ML) algorithms to detect cracks. The ML does not define some strict rules to detect the cracks, but trains the model based on learning experience to detect cracks from the background. A number of road crack detection methods, e.g., support vector machine (SVM) [11], CrackTree [12] and CrackForest [13], have been proposed. The advantage of these ML methods is that the detection model does not need the Ground Truth. However, these methods are not effective in different lighting conditions and crack images with the complex backgrounds (such as the ground shadows and dirt), which make it difficult to detect all kinds of cracks.

With the development of computing technology, the deep learning has shown advantages in the field of image detection [14]. The CNN, as a key algorithm of the deep learning, is an effective automatic crack detection method in the field of pavement health monitoring and pavement maintenance [15]. Some studies in recent literature [16-21] shows that the deep CNN (DCNN) is superior to traditional ML technology in pavement diseases detection. In order to overcome the challenges that the extensively varying real-world situations (e.g., lighting and shadow changes) may bring to the wide adoption of IPT, Cha et al. [22] proposed a vision-based method, which uses the deep architecture of CNN to detect concrete cracks by sliding the window of test image without calculating defect features. Gopalakrishnan et al. [23] adopted transfer learning to automatically detect cracks from images of hot mix asphalt and concrete pavement. Chen et al. [24] proposed a deep learning framework based on a CNN and naive Bayesian data fusion strategy, namely NB-CNN, to analyze each single video frame for crack detection.

Compared with the CNN based method using sliding window, Maeda et al. [25], Wang et al. [26], Arya et al. [27] applied object detection frames to crack detection. To provide quasi real-time simultaneous detection of multiple types of damages, Cha et al. [28] proposed a structural crack detection method based on the Faster R-CNN. However, the target detection model can only roughly locate the crack position, and unable to calculate the crack length and other re-

lated information. The CNN model that assigned category labels to a single pixel according to the local context around the pixel can accurately mark the crack pixels in the pavement image. Yang et al. [29] proposed a new deep learning technology using a fully convolutional network (FCN) to detect and measure different cracks at the pixel level. Nevertheless, the class label assigned to a single pixel is still based on the local context around the pixel, which leads to an overestimation of the crack width. Though the image detection technology based on deep learning has made great progress in crack detection, however, the application of neural networks to crack detection is limited by various complex situations. The CNN algorithm relies on a large number of labeled samples, but it is very expensive to obtain labeled samples. The cracks are generally divided into transverse cracks, longitudinal cracks and oblique cracks according to their shapes [30], but due to the variety of crack shapes and rich features in the actual situations, the real labels may deviate from the manual labels, thus it is difficult to improve the detection precision. At the same time, due to the unclear boundaries between cracks and the background, coupled with the influence of image noise (such as the changes of light and shadow) and the subjectivity of human beings, the manual marking is prone to deviations, which leads to the inevitable noise in these samples. Because semi-supervised learning [31-34] can improve model learning performance by combining a small number of labeled samples and a large number of unlabeled samples, it has attracted extensive attention in the field of deep learning.

In view of the above problems, this paper proposes a road crack detection method using the DenseNet classification model based on the semi-supervised learning method, which improves the accuracy, efficiency and generalization ability of the road crack detection, and reduces laborious manual labeling workload. This method can detect cracks in complex road environment and improve the applicability of the CNN detection model.

2. Methodology

2.1. Overview of Model

The semi-supervised learning method proposed in this paper consists of two parts: pseudo labelling and self-learning. Pseudo labelling is a process of using the model trained on the labeled data to predict the unlabeled data, screening the samples according to the prediction results, and then inputting them into the model for training. The label removed from the training set is regarded as unlabeled dataset, then use the model pre-trained by the training set to relabel the dataset. The difference between the original label and the new label of each sample in the training set is compared. Samples with the same class label remain unchanged and discard samples with different class labels. This process is called self-learning. Table 1 shows the symbols commonly

used in the proposed scheme. The overall framework of the proposed scheme is shown in Figure 1.

| Symbol	Description
$\mathcal{I}$	Initial training set
$f_i^{(n)}$	The network model used, n denotes the number of pseudo labelling iterations, i represents the iteration number s of self-learning in each round of pseudo labelling
$\mathcal{U}^{(n)}$	Unlabeled dataset
$\mathcal{P}^{(n)}$	Pseudo labeled dataset
$\mathcal{C}^{(n)}$	The dataset obtained by replacing the label with the wrongly labeled sample
$\mathcal{E}_i^{(n)}$	The enhanced training set consist of initial training set and wrongly labeled sample dataset
$\mathcal{D}_i^{(n)}$	Unlabeled dataset generated after the training set is unlabeled
$\mathcal{S}_i^{(n)}$	The training set obtained by re-labeling the training set
$\mathcal{R}_i^{(n)}$	Filter the labels of the training set
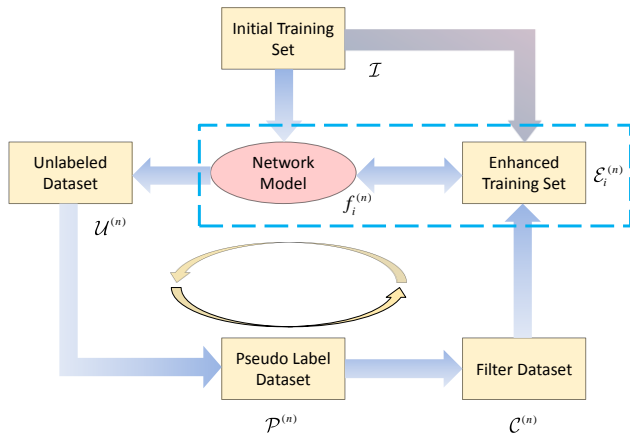
Figure 1. Semi-supervised learning framework diagram; outside the dotted box is pseudo labelling, inside the dotted box is self-learning.

The basic principle of self-learning: for the training set which is obtained either by the manual labelling or pseudo labelling, it is necessary to detect the consistency of the labels through model self-labeling, that is, whether the labels of the samples truly represent their features. The pre-trained model is used to label the unlabeled samples ($\mathcal{U}^{(n)}$), and then the labeled dataset is used as the training set to make the model learn. The trained model is used to label the originally unlabeled samples ($\mathcal{U}^{(n)}$) again (Figure 2). The results before and after labeling are not necessarily the same. How to

correct these deviations is important to improve the performance of the model. It is necessary to replace or delete the inconsistent labels in the training set, so that the model can achieve consistent stability for the training set.
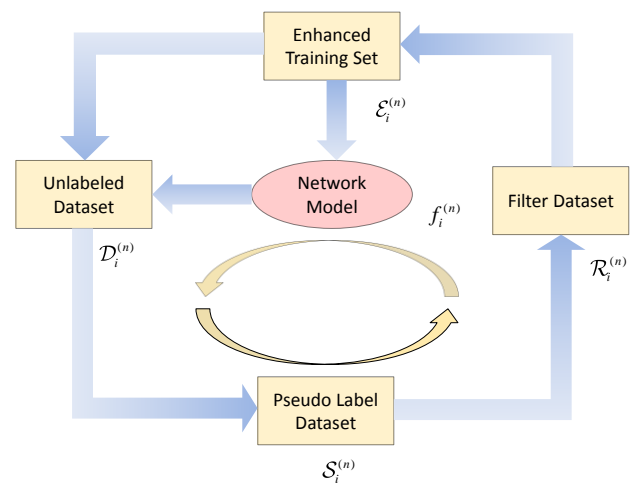

Figure 2. Schematic diagram of self-learning framework.

2.2. Semi-Supervised Learning

The initial training set containing a small number of crack and background images are used to pretrain the CNN model, which is used to classify unlabeled road images to obtain the pseudo-labelled set. The samples in the pseudo-labelled set

with wrong labels need to be manually screened out, and replaced with the new correct class label sequence and combined with the training set to form an enhanced training set. The model is trained again, and this process is repeated until the convergence conditions are satisfied. In this paper, the crack detection task is regarded as the modification of the "two-category classification" problem of the background and crack in the road images. The samples are divided into two main categories: background and crack. Meanwhile, according to the specific features of the object, the two categories are further divided into multiple sequences. For example, the background is divided into road, sign line and so on. The trained model detects the unlabeled dataset, then replaces the samples wrongly labeled as cracks in the recognition results with new sequence background labels, and adds them to the training set. The background is the same. At the same time, in order to balance the number of samples of each class in the training set, and to ensure that the model has enough samples to learn the corresponding features for a single class in the training process, it is generally specified as a new class when the number of wrongly labeled samples under a single label reaches a specific threshold. Pseudo label learning enhances the number and category of training sets.

Let $\mathcal{I} = (x_i, y_i), \forall i = 1, \cdots, a$ be the initial training sets of road crack images, $x_i \in \mathbb{R}^d$, $y_i \in \{1, \ldots, K\}$. There are two kinds of labels: background and crack. Assuming that the background label has two sequences and the crack label has only one sequence, thus there are three labels: background 1, background 2 and crack 1 ($K = 3$), and each label contains a small number of samples.

The unlabeled sample set $\mathcal{U}^{(n)} = (z_i)_{i=1}^{b}, b < a$ is given (n is the batch of the sample set), which belongs to the same group as $\mathcal{I}$, and all of them are related datasets containing pavement crack images, and the number of samples

of $\mathcal{U}$ is significantly greater than $\mathcal{I}$.

Given a neural network model $f(x) : \mathbb{R}^d \to \mathbb{R}$, for a certain point x, the prediction probability distribution is Soft-Max:

$$\hat{p}_f(s|x) = \frac{e^{f_s(x)}}{\sum_{l=1}^{K} e^{f_l(x)}} \tag{1}$$

The iterative scheme can be described as follows:

1. Use the initial training set $\mathcal{I}$ to train the model $f(x)$ to get the initial model $f_0^{(1)}(x)$. The initial model $f_0^{(1)}$ is used to pseudo label all the samples in the first batch of the originally unlabeled sample set $\mathcal{U}^{(1)}$ to obtain the labeled sample set $\mathcal{P}^{(1)}$;

2. According to the results of pseudo labels, the samples with wrong labels in $\mathcal{P}^{(1)}$ are screened out, and are replaced with new correct class sequence labels to create a subset $\mathcal{C}^{(1)} = (o_i, v_i)_{i=1}^{r}$, ($\mathcal{C} \subset \mathcal{P}$), $v_i \in \{1, \ldots, H\}$. For example, suppose that the sample labels of the initial training set corresponding to the model include: background 1, background 2, and crack 1, then in subset $\mathcal{C}$, all crack samples that were wrongly labeled as background 1 are labeled as crack 2, all crack samples that were wrongly labeled as background 2 are labeled as crack 3, and all background samples that were wrongly labeled as crack 1 are labeled as background 3, otherwise, no new class label will be added.

3. The initial training set $\mathcal{I}$ and the sample set $\mathcal{C}^{(1)}$ of the replacement label are combined to form a new sample set $\mathcal{E}_1^{(1)} = (e_i, g_i)_{i=1}^{w}$, $w = a + r$, $g_i \in \{1, \ldots, N\}, N = H + K$. The specific process is shown in Figure 2.
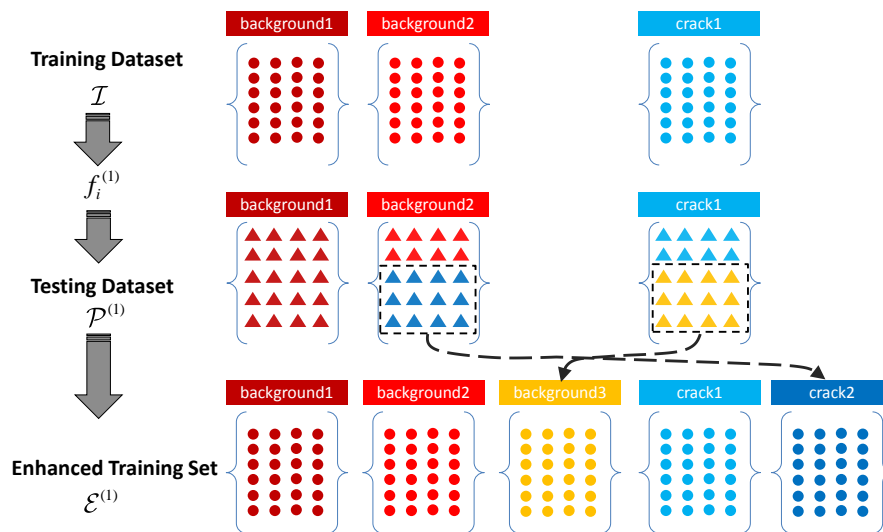

Figure 3. The training set is expanded by pseudo labelling. Each color dot represents a training set sample and its label, and each color triangle represents a pseudo labeled sample and its label.

4. The new model $f_1^{(1)}$ is trained by minimizing the cross-entropy loss of the enhanced training set $\mathcal{E}_1^{(1)}$. Discard the labels in the enhanced training set $\mathcal{E}_1^{(1)}$ to generate the unlabeled sample set $\mathcal{D}_1^{(1)}$; all the samples in the $\mathcal{D}_1^{(1)}$ are predicted by the model $f_1^{(1)}$, and the results are used as the label of samples to obtain a new training set $\mathcal{S}_1^{(1)}$;

5. Compare $\mathcal{E}_1^{(1)}$ with $\mathcal{S}_1^{(1)}$, filter the samples of different labels, delete the samples with the original label as the background and the new label as the background, and get the updated training set $\mathcal{R}_1^{(1)}$. The specific process is shown in Figure 5;
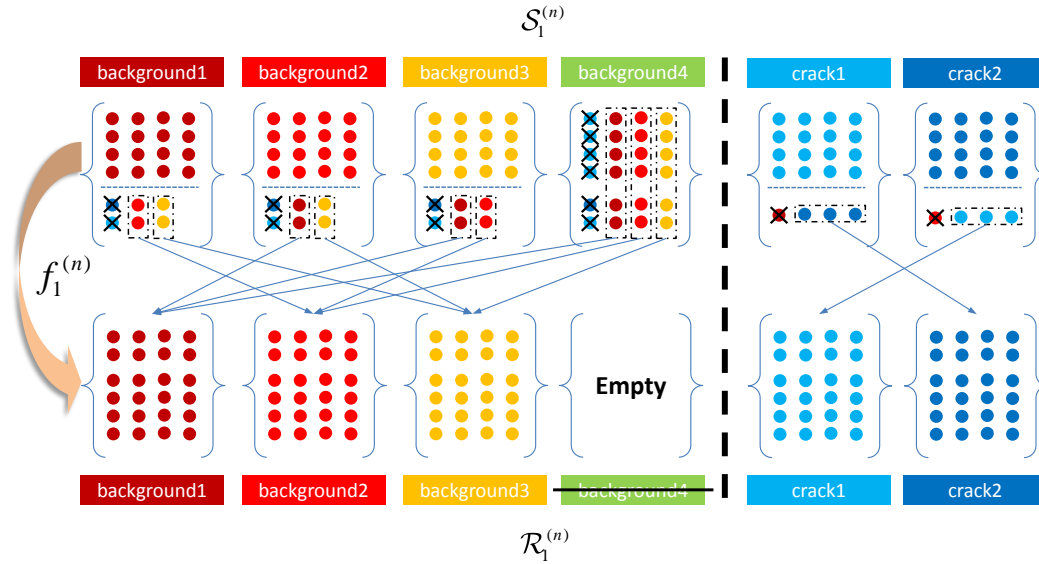

Figure 4. The training set is contracted by self-learning, and various color dots represent the samples and their labels.

6. Take $\mathcal{R}_1^{(1)}$ as the enhanced training set, replace $i+1$ with i, and return to step 4. The new model $f_0^{(2)}$ is trained by minimizing the cross-entropy loss of labeled sample $\mathcal{E}^{(1)}$:

$$\mathcal{L}(\boldsymbol{g}, f(\boldsymbol{e};\theta)) = -\sum_{j=1}^{N} g_j \log f_j(\boldsymbol{e};\theta) \qquad (2)$$

where θ is the parameters in the model.

7. Replace n with $n+1$ and return to step 1.

2.3. Evaluation Indicators

In order to quantitatively evaluate the influence of the proposed method on the crack detection, the commonly used indexes, the P (precision), R (recall), harmonic mean F1 and IoU, in the ML are introduced:

$$P = \frac{TP}{TP+FP} \qquad (3)$$

$$R = \frac{TP}{TP+FN} \qquad (4)$$

$$F1 = \frac{2 \times P \times R}{P+R} \qquad (5)$$

$$IoU = \frac{TP}{TP+FP+FN} \qquad (6)$$

where, TP (true positive) is the number of images correctly classified as cracks, TN (true negative) is the number of images correctly classified as background, FP (false positive) is the number of images mistakenly classified as cracks, and FN (false negative) is the number of images mistakenly classified as background, as shown in Figure 5.

Because the Precision does not take into account the situation of missing detection of cracks, and the recall does not take into account the situation of detecting the background as cracks, thus, the F1 value and IoU are used as the main evaluation indicators in this paper.
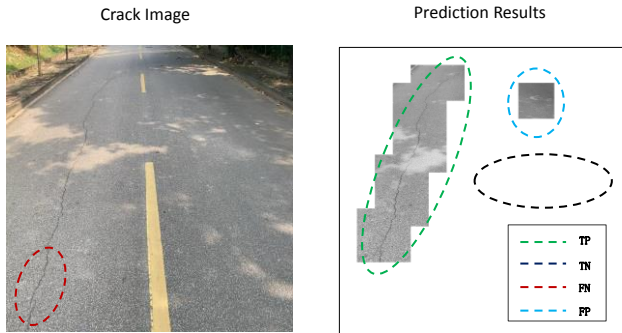
Figure 5. *Example of classification of model test results.*

3. Experiment

3.1. Data Sources

The 480 road images used in this paper were taken under natural light conditions by an iPhone se, whose focus is 3 mm, f / 1.8 aperture. The original images collected in this paper have the resolution of $1,920 \times 1,080$ pixels, which is adjusted to $2,048 \times 2,048$ pixels. Each image is clipped into 64 blocks of 256×256 pixels, and the position information of each block image in the original image is retained. After clipping, the block images of 256×256 pixels are processed through the data augmentation techniques (rotation, mirroring, etc.) to get the training sample set.

The proposed method divides the dataset into 8 experimental batches, each batch contains 60 images. In each batch, 6 images are randomly selected as the batch validation set; 4 images in each batch and totally 32 images are selected as the testing set, and the rest as the batch training set, which are used for the training of the crack detection model.

Table 2. *Details of training, validation and test sets, with block images in brackets.*

Training set (single batch)	Validation set (single batch)	Test set
54 (3,456)	6 (384)	32 (2,048)

After a round of iteration for a batch of training samples, the obtained model is used to detect the testing set, and the evaluation index is calculated for the detection results of each image in the testing set, and the training batch with the worst evaluation index is selected as the input of the next round of unlabeled samples.

Totally 75 block images with cracks are selected and labeled as Crack 1, and then 250 block images without cracks are randomly selected and labeled as Background 1, which are used as the initial training set. Some images in the initial training sets, for example, are shown in Figure 6.

Figure 6. *Examples of initial training set.*

3.2. Model Training Configuration

All the experiments were performed on an Intel Desktop with a quad core i7-4790 CPU@3.6 GHz processor, 8GB memory and NVIDIA GeForce GTX 1650 4GB GPU. The proposed semi-supervised learning method based on a DenseNet classification model. The realization of the DenseNet architecture is based on the deep learning toolbox of MATLAB. The minimum batch size of training is 16 and the learning rate is 3×10^{-4}, all data were rotated fiveepochs. The DenseNet was trained with the dataset and used to detect every batch of images and relabel them according to the detection results. The noise samples were added randomly in the training set, and the self-learning was performed for 34 iterations.

4. Results and Discussions

4.1. Training Set Obtained

In this paper, the training set of multiple sequences including the cracks and backgrounds is obtained by the pseudo labelling. Figure 7 shows that the proposed method detects 9 kinds of road cracks and 12 kinds of road background in the road image dataset. The sample number in the background sequence will change obviously with the increase of the number of iterations, while the sample number in the crack sequence remains relatively stable. It can be seen that the features of cracks are relatively simple, and the background will include a variety of categories with the change of road conditions. With pseudo labelling, the background of different features can be divided from a single label, and the cracks hidden in similar background features can be separated.
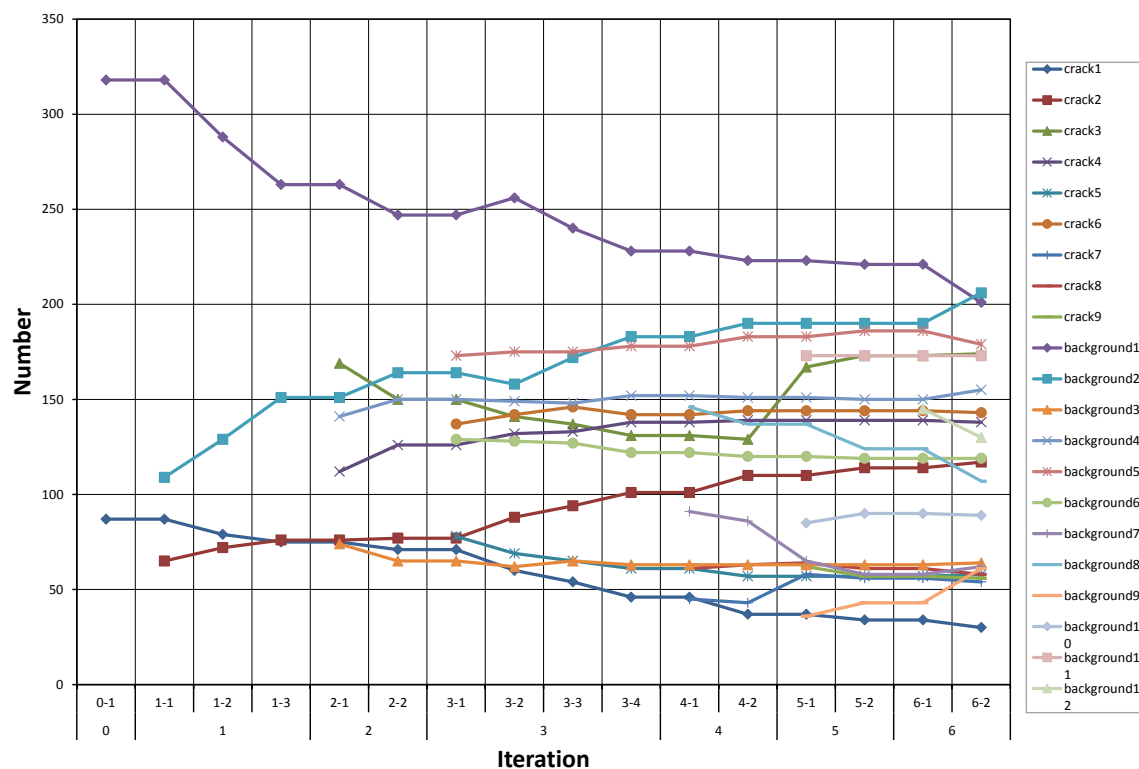
Figure 7. The change of the sample number of each label in the training set during the semi-supervised learning process of the model.

As shown in Figure 8, the number of the background and crack samples in the training set is increasing with the increase of training rounds, and the performance of the model was also improved. It shows that the more samples the model has, the smaller the generalization error was.
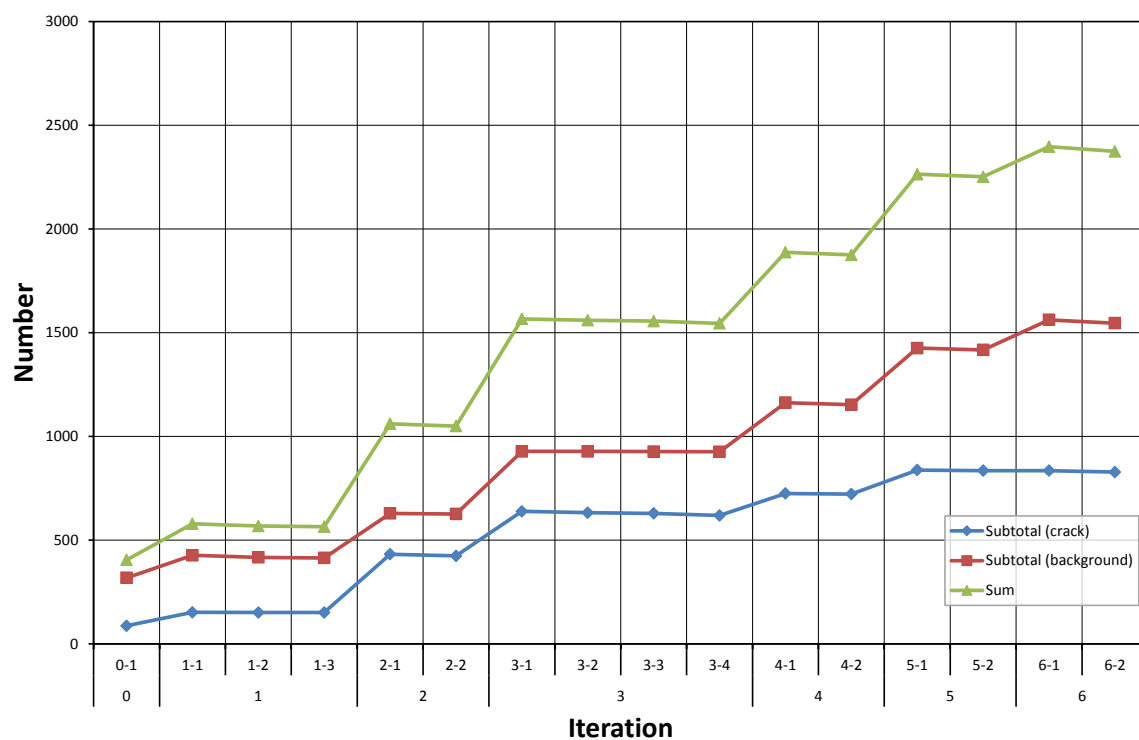

Figure 8. The change of the total number of samples of two types of labels in the training set during the iterative learning process of the model.

Through six rounds of the semi-supervised learning, nine series of crack categories were obtained, and six of them were shown in Figure 9. The figure shows that the model automatically separates the cracks with different features, which was similar to the manual classification of cracks into the transverse cracks, longitudinal cracks and oblique cracks.
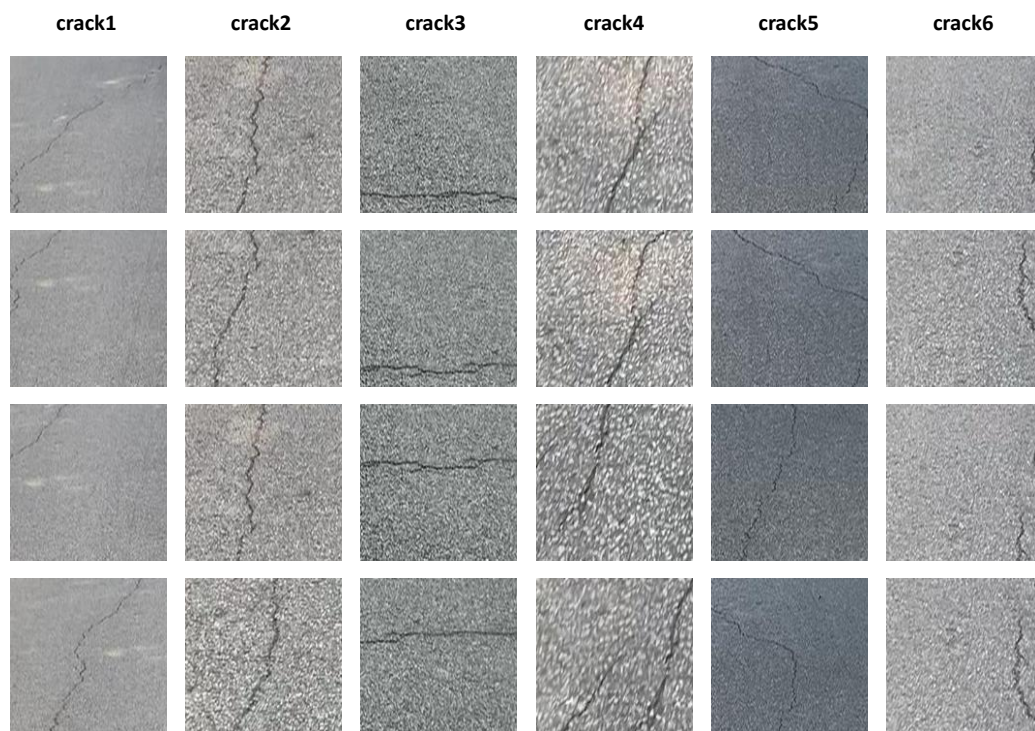

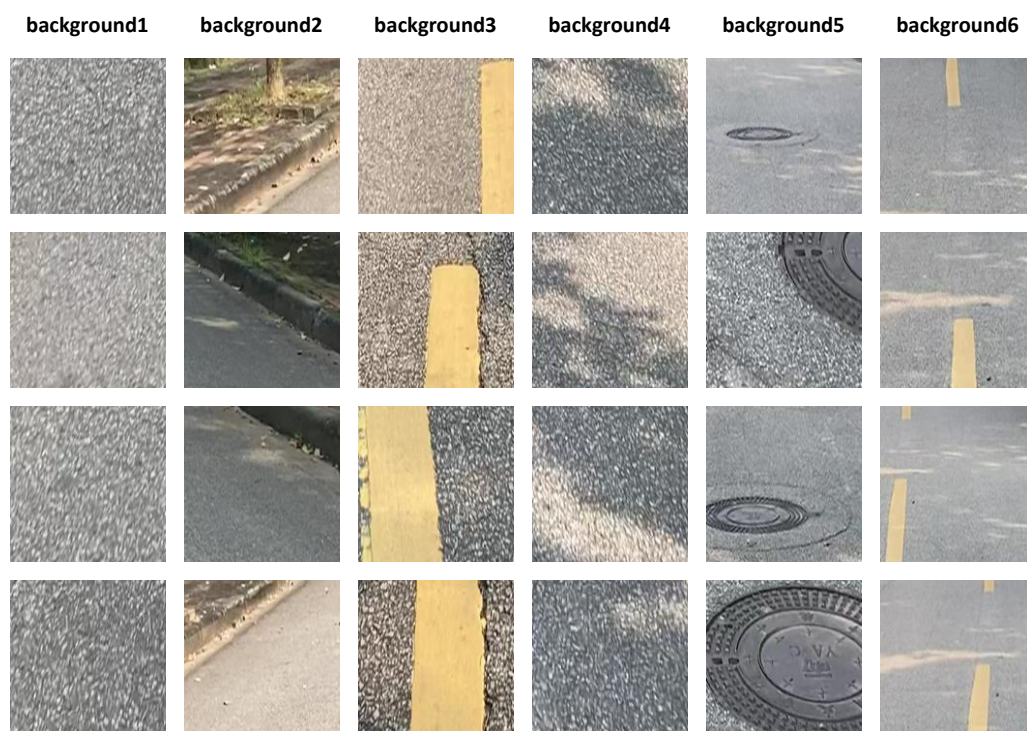*Figure 9. Example of crack sequence detected by the DenseNet.*

Figure 10. Background sequences detected by the DenseNet.

As shown in Figure 10, the proposed method divides the background into pavement, curb, manhole cover, sign line, etc. Background 1 and Background 4 were classified to the different category as Background 4 has shadow. Cracks and road markings were mixed in Background 3, which was difficult to separate from the background because the cracks coincide with the edges of road noise (road markings, shadows, etc.).

The validation accuracy of each iteration was shown in Figure 11. The results show that the performance of the model is greatly improved after the self-learning of the training set. The method can get high precision in the automatic segmentation of pavement cracks, while the model of minimizing the error of training samples tends to overfit specific training samples. The model with strong fitting ability will be more complex and prone to overfitting.
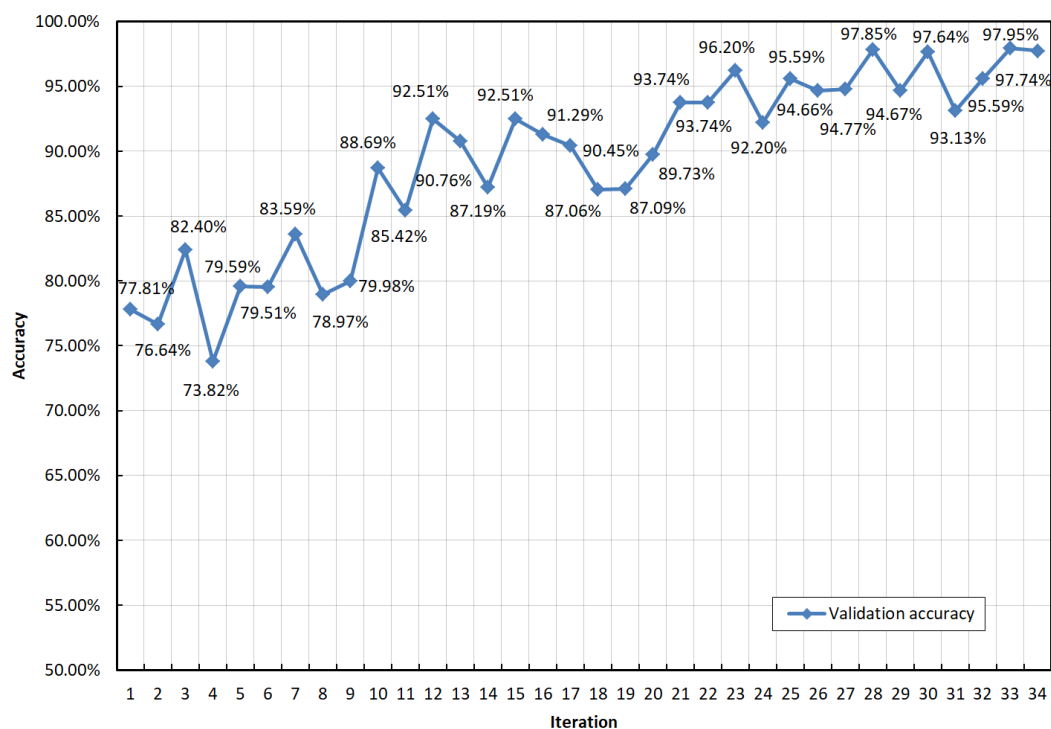

Figure 11. Validation accuracy of the training process.

4.2. Test Set Detection Results

After each round of training, the DenseNet classification model was tested with the testing set. It can be seen from Table 3 that the error rate of detecting the crack as the background is low, and most of the error rates are concentrated on detecting the background as crack. In the first five iterations, the error rate of Crack 3 was higher, which was always more than 50%.

As shown in Figure 12, with the increase of the number of

rounds, the value of the evaluation index shows an overall upward trend. When the number of rounds reached 6, the precision was 89.20%, the recall was 99.48%, the F1 value was 94.06%, and the IoU was 88.79%. Among the four indicators, the recall has been on the rise, which indicates that the situation under which the model mistakenly detects the cracks as the background was gradually decreasing, while the precision value was relatively low, which indicates that the situation that the model mistakenly detects the background as cracks was more likely.

Table 3. Error rate of single label in each iteration test set.

Label	1	2	3	4	5	6
background 1	3.85%	0.52%	0.67%	0.00%	0.00%	0.00%
background 2	10.75%	2.92%	1.56%	1.78%	0.70%	0.00%

| Label	1	2	3	4	5	6
background 3	—	6.12%	6.90%	0.00%	0.00%	0.00%
background 4	—	7.39%	3.63%	0.00%	0.00%	0.00%
background 5	—	—	0.00%	0.00%	0.00%	0.00%
background 6	—	—	1.05%	0.00%	0.00%	0.00%
background 7	—	—	—	12.73%	0.00%	0.00%
background 8	—	—	—	0.00%	0.00%	0.00%
background 9	—	—	—	—	0.00%	0.00%
background 10	—	—	—	—	0.00%	0.00%
background 11	—	—	—	—	0.00%	0.00%
background 12	—	—	—	—	—	0.32%
crack 1	16.67%	6.45%	0.00%	14.29%	0.00%	0.00%
crack 2	42.20%	24.32%	19.35%	14.71%	23.40%	15.56%
crack 3	—	82.42%	57.45%	52.78%	54.81%	7.32%
crack 4	—	46.09%	11.94%	30.26%	48.84%	4.76%
crack 5	—	—	61.76%	42.31%	48.15%	30.00%
crack 6	—	—	54.93%	40.43%	24.14%	17.24%
crack 7	—	—	—	34.78%	0.00%	0.00%
crack 8	—	—	—	17.65%	31.25%	0.00%
crack 9	—	—	—	—	0.00%	0.00%
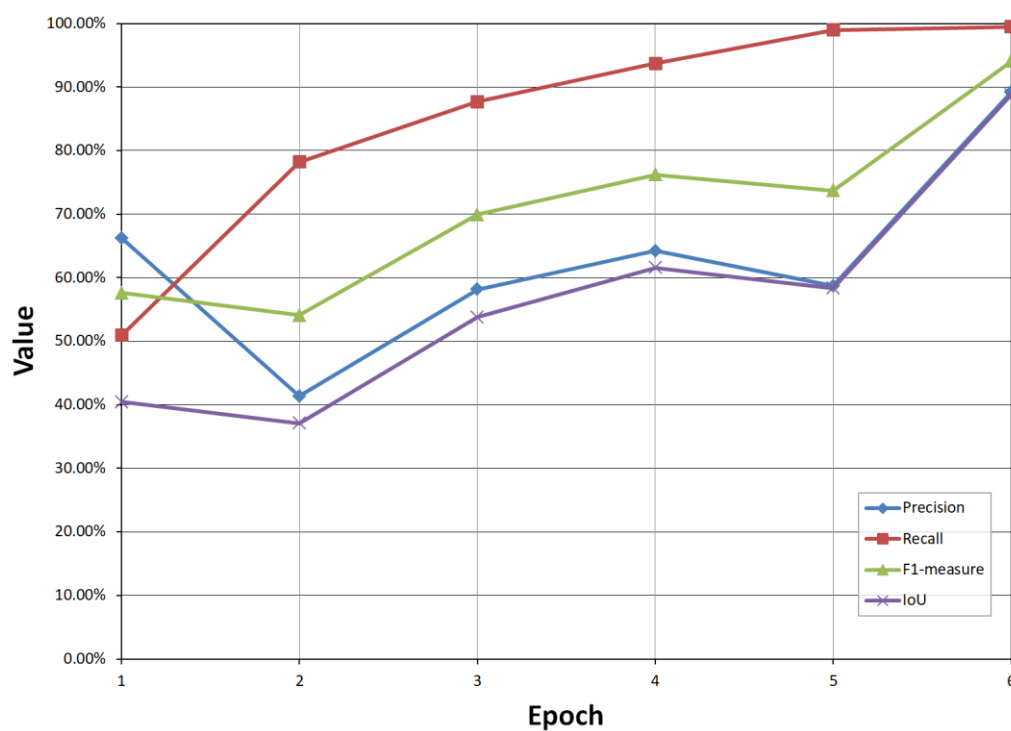
Figure 12. Changes of evaluation indexes of test set with the iteration of the training model.

As shown in Figure 13, the method detects the testing set and uses the anchor box to locate and label the crack. After the first round of semi-supervised learning, only Crack 1 in the upper left corner of the model can correctly detect the cracks. After the second round of semi-supervised learning, all the cracks on the road surface are detected, and the corresponding labels were Crack 3 and Crack 4, respectively. However, a large number of cases of mistakenly detecting the background as cracks appear. After the third round of semi-supervised learning, the false-positive results were re-duced on the premise of retaining the detected cracks completely. After the fourth round of semi-supervised learning, the false-positive results were further reduced, but some cracks were not detected. After the fifth round of semi-supervised learning, all the cracks were detected, while there were still false positive cases. After the sixth round of semi-supervised learning, the false-positive results were completely eliminated, and all the cracks were correctly identified. The crack labels identified were Crack 2, Crack 3, Crack 4 and Crack 9.

Figure 13. *Testing results of different test round.*

4.3. Comparative Test Results of Different Models

Besides the DenseNet classification model, other deep learning models, AlexNet [35], GoogleNet [36], VGG16 [37], VGG19 [37], ResNet18 [38], ResNet50 [38], ResNet101 [38] were also tested in this paper. In the training process, the high-quality training set come from the DenseNet classification model after the semi-supervised learning, and the validation set was also from the campus road crack dataset. In addition, the training parameters of these models are the optimal parameters, and the comparative evaluation indexes were shown in Table 4.

Table 4. *The detection performance index of the training set obtained by semi-supervised learning using different network models for the same test set.*

Model	depth	Number of layers	parameter	Precision (%)	Recall (%)	F1 (%)	IoU (%)
AlexNet	8	25	61 M	85.15%	86.28%	85.71%	75.00%
GoogleNet	22	144	7 M	90.87%	83.63%	87.10%	77.14%
VGG16	16	41	138 M	96.77%	94.17%	95.45%	91.30%

Model	depth	Number of layers	parameter	Precision (%)	Recall (%)	F1 (%)	IoU (%)
VGG19	19	47	144 M	91.63%	95.44%	93.50%	87.79%
ResNet18	18	71	11.7 M	84.39%	76.89%	80.47%	67.32%
ResNet50	50	177	25.6 M	83.09%	74.14%	78.36%	64.42%
ResNet101	101	347	44.6 M	92.75%	84.21%	88.28%	79.01%
DenseNet	201	708	20 M	83.80%	69.44%	75.95%	61.22%

The prediction Precision of VGG16 was 96.77%, the recall rate was 94.17%, the F1 value was 95.45%, and the IoU was 91.30%. Although the recall rate was slightly lower than that of VGG19, the Precision was better than other models, and the F1 value was more than 7% higher than the third, which shows that the model structure has a great impact on the test results. It can be seen from the table that the more layers the model has, the better the model performs. The experimental results show that the more parameters, the better the generalization effect.

5. Conclusion

This paper proposes a method combining the pseudo labelling and self-learning to enhance the performance of the model from the perspective of improving the quality of the training set. The pseudo labelling can realize the diversity of labels and improve the generalization ability of models, while the self-learning can improve the detection precision of models.

This method can accurately detect the road crack area in the image and reduce the influence of water stains, shadows, road markings and other noises in the image after the semi-supervised learning. Through a large number of experiments, it was proved that the proposed method can continuously improve the detection precision and recall, and it was proved that the proposed method can identify inconsistent labels and recommend consistent labels.

This paper proves that the pseudo labelling can improve the precision of the benchmark model and reduce the rate of disease false detection.

While the proposed method shows significant improvements, there are several avenues for future research. One potential area is the exploration of advanced techniques for handling more complex noise types in road images, such as seasonal variations and lighting conditions. Additionally, integrating other semi-supervised learning techniques, such as consistency regularization or adversarial training, could further refine the model's robustness. Another interesting direction would be to apply the proposed method to other types of infrastructure defects or extend it to three-dimensional data sources, like LiDAR, to enhance the detection of surface irregularities. Lastly, investigating the method's scalability and performance in real-time applications could provide insights into its practical deployment in large-scale infrastructure monitoring systems.

Abbreviations

IoU	Intersection over Union
CNN	Convolutional Neural Network
ML	Machine Learning
SVM	Support Vector Machine
FCN	Fully Convolutional Network
P	Precision
R	Recall
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative

Author Contributions

Jun Yang: Conceptualization, Data curation, Formal Analysis, Methodology, Writing – original draft

Xiaoli Sun: Conceptualization, Data curation, Formal Analysis, Funding acquisition, Investigation, Resources, Visualization, Writing – review & editing

Shuai Teng: Conceptualization, Data curation, Formal Analysis, Funding acquisition, Investigation, Methodology, Visualization, Writing – original draft

Funding

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] Zhang, J.; Yang, X.; Wang, W.; Brilakis, I.; Davletshina, D.; Wang, H.; Cao, M. Segment-to-track for pavement crack with light-weight neural network on unmanned wheeled robot. *Automation in Construction* 2024, *161*, 105346, https://doi.org/10.1016/j.autcon.2024.105346

[2] Yu, Z.; Shen, Y.; Zhang, Y.; Xiang, Y. Automatic crack detection and 3D reconstruction of structural appearance using underwater wall-climbing robot. *Automation in Construction* 2024, *160*, 105322, https://doi.org/10.1016/j.autcon.2024.105322

[3] Kirschke, K. R.; Velinsky, S. A. Histogram-Based Approach for Automated Pavement-Crack Sensing. *Journal of Transportation Engineering* 1992, *118*, 700-710, https://doi.org/10.1061/(ASCE)0733-947X(1992)118:5(700)

[4] Bhutani, K. R.; Battou, A. An application of fuzzy relations to image enhancement. *Pattern Recognition Letters* 1995, *16*, 901-909, https://doi.org/10.1016/0167-8655(95)00035-F

[5] Oliveira, H.; Correia, P. L. Automatic road crack segmentation using entropy and image dynamic thresholding. In Proceedings of 2009 17th European Signal Processing Conference.

[6] Ying, L.; Salari, E. Beamlet Transform-Based Technique for Pavement Crack Detection and Classification. *Computer-Aided Civil and Infrastructure Engineering* 2010, *25*, 572-580, https://doi.org/10.1111/j.1467-8667.2010.00674.x

[7] Ikhlas, A.; Osama, A.; E., K. M. Analysis of Edge-Detection Techniques for Crack Identification in Bridges. *Journal of Computing in Civil Engineering* 2003, *17*, 255-263, https://doi.org/10.1061/(ASCE)0887-3801(2003)17:4(255)

[8] Li, Q.; Zou, Q.; Zhang, D.; Mao, Q. FoSA: F* Seed-growing Approach for crack-line detection from pavement images. *Image and Vision Computing* 2011, https://doi.org/10.1016/j.imavis.2011.10.003

[9] Cheng, H. D.; Shi, X. J.; Glazier, C. Real-Time Image Thresholding Based on Sample Space Reduction and Interpolation Approach. *Journal of Computing in Civil Engineering* 2003, *17*, 264-272, https://doi.org/10.1061/(ASCE)0887-3801(2003)17:4(264)

[10] Huang, Y.; Xu, B. Automatic inspection of pavement cracking distress. *Journal of Electronic imaging* 2006, *15*, 013017, https://doi.org/10.1117/1.2177650

[11] al, M. G. e. Adaptive Road Crack Detection System by Pavement Classification. *Sensors* 2011, *11*, 9628-9657, https://doi.org/10.3390/s111009628

[12] Zou, Q.; Cao, Y.; Li, Q.; Mao, Q.; Wang, S. CrackTree: Automatic crack detection from pavement images. *Pattern Recognition Letters* 2011, *33*, 227-238, https://doi.org/10.1016/j.patrec.2011.11.004

[13] Shi, Y.; Cui, L.; Qi, Z.; Meng, F.; Chen, Z. Automatic Road Crack Detection Using Random Structured Forests. *IEEE Transactions on Intelligent Transportation Systems* 2016, *17*, 3434-3445, https://doi.org/10.1109/TITS.2016.2552248

[14] Teng, S.; Liu, A.; Wu, Z.; Chen, B.; Ye, X.; Fu, J.; Kitiporncha, S.; Yang, J. Automated detection of underwater cracks based on fusion of optical and texture information. *Engineering Structures* 2024, *315*, 118515, https://doi.org/10.1016/j.engstruct.2024.118515

[15] Bai, S.; Ma, M.; Yang, L.; Liu, Y. Pixel-wise crack defect segmentation with dual-encoder fusion network. *Construction and Building Materials* 2024, *426*, 136179, https://doi.org/10.1016/j.conbuildmat.2024.136179

[16] Zhang, L.; Yang, F.; Zhang, Y. D.; Zhu, Y. J. Road crack detection using deep convolutional neural network. In Proceedings of 2016 IEEE International Conference on Image Processing (ICIP), 25-28 Sept. 2016; pp. 3708-3712.

[17] Wang, K. C. P.; Zhang, A.; Li, J. Q.; Fei, Y.; Chen, C.; Li, B. *Deep Learning for Asphalt Pavement Cracking Recognition Using Convolutional Neural Network*; 2017; https://doi.org/10.1061/9780784480922.015 pp. 166-177

[18] Alfarrarjeh, A.; Trivedi, D.; Kim, S. H.; Shahabi, C. A Deep Learning Approach for Road Damage Detection from Smartphone Images. In Proceedings of 2018 IEEE International Conference on Big Data (Big Data), 10-13 Dec. 2018; pp. 5201-5204.

[19] Hoang, N.; Nguyen, Q.; Tran, V. Automatic recognition of asphalt pavement cracks using metaheuristic optimized edge detection algorithms and convolution neural network. *Automation in Construction* 2018, *94*, 203-213, https://doi.org/10.1016/j.autcon.2018.07.008

[20] Tong, Z.; Gao, J.; Han, Z.; Wang, Z. Recognition of asphalt pavement crack length using deep convolutional neural networks. *Road Materials and Pavement Design* 2018, *19*, 1334-1349, https://doi.org/10.1080/14680629.2017.1308265

[21] Zhang, A.; Wang, K. C. P.; Li, B.; Yang, E.; Dai, X.; Peng, Y.; Fei, Y.; Liu, Y.; Li, J. Q.; Chen, C. Automated Pixel-Level Pavement Crack Detection on 3D Asphalt Surfaces Using a Deep-Learning Network. *Computer-Aided Civil and Infrastructure Engineering* 2017, *32*, 805-819, https://doi.org/10.1111/mice.12297

[22] Cha, Y.; Choi, W.; Büyüköztürk, O. Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks. *Computer-Aided Civil and Infrastructure Engineering* 2017, *32*, 361-378, https://doi.org/10.1111/mice.12263

[23] Gopalakrishnan, K.; Khaitan, S. K.; Choudhary, A.; Agrawal, A. Deep Convolutional Neural Networks with transfer learning for computer vision-based data-driven pavement distress detection. *Construction and Building Materials* 2017, *157*, 322-330, https://doi.org/10.1016/j.conbuildmat.2017.09.110

[24] Chen, F.; Jahanshahi, M. R. NB-CNN: Deep Learning-Based Crack Detection Using Convolutional Neural Network and Naïve Bayes Data Fusion. *IEEE Transactions on Industrial Electronics* 2018, *65*, 4392-4400, https://doi.org/10.1109/TIE.2017.2764844

[25] Maeda, H.; Sekimoto, Y.; Seto, T.; Kashiyama, T.; H. Omata. Road Damage Detection and Classification Using Deep Neural Networks with Smartphone Images. *Computer-Aided Civil and Infrastructure Engineering* 2018, *33*, 1127-1141, https://doi.org/10.1111/mice.12387

[26] Wang, W.; Wu, B.; Yang, S.; Wang, Z. Road Damage Detection and Classification with Faster R-CNN. In Proceedings of 2018 IEEE International Conference on Big Data (Big Data), 10-13 Dec. 2018; pp. 5220-5223.

[27] Arya, D.; Maeda, H.; Ghosh, S. K.; Toshniwal, D.; Mraz, A.; Kashiyama, T.; Sekimoto, Y. Transfer Learning-based Road Damage Detection for Multiple Countries. 2020; p arXiv: 2008.13101.

[28] Cha, Y.; Choi, W.; Suh, G.; Mahmoudkhani, S.; Büyüköztürk, O. Autonomous Structural Visual Inspection Using Region-Based Deep Learning for Detecting Multiple Damage Types. *Computer-Aided Civil and Infrastructure Engineering* 2018, *33*, 731-747, https://doi.org/10.1111/mice.12334

[29] Yang, X.; Li, H.; Yu, Y.; Luo, X.; Huang, T.; Yang, X. Automatic Pixel-Level Crack Detection and Measurement Using Fully Convolutional Network. *Computer-Aided Civil and Infrastructure Engineering* 2018, *33*, 1090-1109, https://doi.org/10.1111/mice.12412

[30] Liu, J.; Yang, X.; Lau, S.; Wang, X.; Luo, S.; Lee, V. C.; Ding, L. Automated pavement crack detection and segmentation based on two-step convolutional neural network. *Computer-Aided Civil and Infrastructure Engineering* 2020, *35*, 1291-1305, https://doi.org/10.1111/mice.12622

[31] Blum, A.; Mitchell, T. Combining labeled and unlabeled data with co-training. In Proceedings of Proceedings of the eleventh annual conference on Computational learning theory, Madison, Wisconsin, USA; pp. 92–100.

[32] Nigam, K.; McCallum, A. K.; Thrun, S.; Mitchell, T. Text Classification from Labeled and Unlabeled Documents using EM. *Machine Learning* 2000, *39*, 103-134, https://doi.org/10.1023/A:1007692713085

[33] Lee, D. Pseudo-Label: The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks. *ICML 2013 Workshop: Challenges in Representation Learning (WREPL)* 2013.

[34] Xie, Q.; Luong, M. T.; Hovy, E.; Le, Q. V. Self-Training With Noisy Student Improves ImageNet Classification. In Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 13-19 June 2020; pp. 10684-10695.

[35] Krizhevsky, A.; Sutskever, I.; Hinton, G. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of NIPS.

[36] Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper with Convolutions. 2014; https://doi.org/10.1109/cvpr.2015.7298594

[37] Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. 2014.

[38] He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition.