Construction and Building Materials 157 (2017) 322–330

Contents lists available at ScienceDirect

Construction and Building Materials

journal homepage: www.elsevier.com/locate/conbuildmat

Deep Convolutional Neural Networks with transfer learning for computer vision-based data-driven pavement distress detection

Kasthurirangan Gopalakrishnan [a,*], Siddhartha K. Khaitan [b], Alok Choudhary [a], Ankit Agrawal [a]

[a] Department of Electrical Engineering & Computer Science, Northwestern University, Evanston, IL 60201, USA
[b] Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011, USA

HIGHLIGHTS

- Pre-trained deep Convolutional Neural Networks (DCNN) were used for crack detection.
- Pavement images sampled from the FHWA/LTPP database were used as datasets.
- The truncated VGG-16 DCNN was used as a deep feature generator for road images.
- Various machine learning classifiers were trained using the semantic image vectors.
- A neural network classifier trained on deep transfer learning vectors gave the best results.

ARTICLE INFO

Article history:
Received 13 June 2017
Received in revised form 8 September 2017
Accepted 18 September 2017
Available online 23 September 2017

Keywords:
Pavement cracking
Digital Image
Deep learning
Transfer learning
Random Forest
Convolutional Neural Networks

ABSTRACT

Automated pavement distress detection and classification has remained one of the high-priority research areas for transportation agencies. In this paper, we employed a Deep Convolutional Neural Network (DCNN) trained on the 'big data' ImageNet database, which contains millions of images, and transfer that learning to automatically detect cracks in Hot-Mix Asphalt (HMA) and Portland Cement Concrete (PCC) surfaced pavement images that also include a variety of non-crack anomalies and defects. Apart from the common sources of false positives encountered in vision based automated pavement crack detection, a significantly higher order of complexity was introduced in this study by trying to train a classifier on combined HMA-surfaced and PCC-surfaced images that have different surface characteristics. A single-layer neural network classifier (with 'adam' optimizer) trained on ImageNet pre-trained VGG-16 DCNN features yielded the best performance.

1. Introduction

Transportation agencies routinely collect road condition data as part of their pavement management activities in an effort to maintain pavements in a cost-effective manner. Since pavements tend to deteriorate with time under the influence of repeated traffic loading and environmental variations, the use of accurate condition or health monitoring techniques becomes critical for timely detection of distress development in the pavements. The 2017 American Society of Civil Engineers ASCE) Infrastructure Report Card assigned a 'D' grade for United States road infrastructure, given that 1 out of every 5 miles of highway pavement is in poor condition [1]. Efficient condition monitoring strategies can aid engineers in developing appropriate scheduling of pavement maintenance and repair activities leading to significant reduction in pavement life-cycle maintenance costs.

Pavement distress evaluation has come a long way from manual visual surveys to acquiring pavement images using downward-looking high-speed digital camera attached to pavement data collection vehicles moving at highway speed [12,15]. Currently, many state highway agencies are moving towards pavement condition data collection using the so-called 3-D automated survey systems that can acquire high-resolution images containing more information elevation and intensity) than 2-D images and are capable of achieving 1-mm crack identification [30,34].

Once the high-resolution digital images of the pavement surfaces are obtained, they are processed through a compression subsystem to achieve size reduction without loss of quality before they are stored. The images are then processed using various algorithms to extract crack features (i.e., orientation, length, density, displacement, location), crack maps, and summary statistics, which

* Corresponding author.
E-mail addresses: rangan@northwestern.edu (K. Gopalakrishnan), skhaitan@iastate.edu (S.K. Khaitan), choudhar@eecs.northwestern.edu (A. Choudhary), ankitag@eecs.northwestern.edu (A. Agrawal).

https://doi.org/10.1016/j.conbuildmat.2017.09.110
0950-0618/

are then recorded in the surface distress database and can be linked to a Pavement Management System (PMS). As of 2012, more than 35 state highway agencies in the United States employed semi-automated and automated image-based methods for network-level pavement cracking data collection [32].

In the current state of practice, two software-based approaches to pavement crack detection workflow are common: manual typically developed in-house and used by public agencies at the project level) and semi-automatic typically developed and provided by private vendors at the network level). In the manual approach, the operator takes advantage of the tools built into the crack detection software to better digitize cracks, conduct measurements, and report results Lee and Kim 2005). In the semi-automatic approach, a set of crack detection algorithms are first applied by the software to automatically obtain the cracking information and the incorrect results are then adjusted manually through a series of human interventions [19,24]. Over the last three decades, a number of studies have focused on improving crack detection algorithms and developing automatic pavement crack detection methods. In fact, the whole field of computer vision based civil infrastructure defect detection is constantly evolving with steady advances being made in sensing technologies, hardware, and software. However, there are still limitations to the existing crack detection methods due to their inability to overcome inherent challenges associated with these pavement images, such as inhomogeneity of cracks, diversity of surface texture (eg., tining), background complexity, presence of non-crack features such as joints, etc.

We propose the use of a pre-trained Deep Convolutional Neural Network (DCNN) model with transfer learning for automated pavement crack detection. Deep Learning (DL) has recently been used for road crack detection but since it typically requires large amounts of labeled training data, here we propose the use of pre-trained DCNN with transfer learning for automated pavement distress detection, and demonstrate its advantages. The rest of the paper is divided as follows. A review of state-of-the-art vision-based automated pavement distress detection methods is discussed first under Related Works followed by a brief introduction to the concept of DCNN with transfer learning for cross domain image analysis. The Experimental Study is then presented which includes description of the datasets used, the proposed methodology, the experiments carried out and the results.

2. Related works

A number of studies reported in the literature have focused on the development of automated image-based pavement crack detection methods, which could broadly be classified into intensity-thresholding, edge detection, wavelet transforms, texture-analysis, and machine learning techniques [38].

The classical intensity-thresholding methods are based on fixing a single threshold for segmenting a grayscale pavement images and creating binary images that distinguish crack and non-crack pixels. More advanced intensity-thresholding methods have included the use of a dynamic optimization based method [31] for segmentation of low signal-to-noise ratio pavement images and embedding entropy [22] into the thresholding framework for pavement crack detection [38]. Recently, an automatic crack detection system was proposed by Zhang et al. [37] by employing a coarse-to-fine methodology that also included concepts like Region of Aggregation (ROA) and Region of Belief (ROB) for segmentation and localization of cracks. Chambon and Moliard [6] proposed a new approach to image-based crack detection, GaMM, based on a multi-scale extraction and Markovian segmentation, which reduced the percentage of false positives in comparison to morphological methods that combine thresholding and refinement by morphological analysis [6].

Edge detection techniques proposed for pavement crack detection have involved the use of Canny filter, Sobel edge detector, morphological filters, etc. [2]. The use of a discrete wavelet transform DWT) and multi-scale resolution wavelet techniques for pavement crack detection has also attracted significant attention [28,33]. Ying and Salari [36] proposed a beamlet transform-based technique for pavement crack detection and classification, which is supposedly more robust in extracting linear features in the presence of noise.

Oliveira and Correia [21] proposed an integrated system, CrackIT, for automatic detection and characterization of cracks in flexible pavement surfaces using a combination of unsupervised learning (clustering) followed by supervised learning (classification), thus eliminating the need for manually labeling the samples. It was noted that although CrackIT was able to detect multiple cracks in the same image, it had difficulty in dealing with cracks less than 2 mm width. It was noted that the most existing commercial automatic crack detection systems aim to quantify cracks that are at least 3 mm in width [21]. Based on the fact that crack pixels in pavement images have distinct grayscale intensities compared to their surrounding non-crack pixels, Cheng et al. [7] proposed a pavement crack detection algorithm based on fuzzy logic. Neural Networks (NN) based classifiers for detection and classification of crack segments in flexible pavement surface video images have been shown to perform slightly better than traditional classifiers such as Bayes classifier and K-nearest neighbor (k-NN) decision rule [17]. A comprehensive review of the computer vision based defect detection and condition assessment of asphalt pavement was presented by Koch et al. [18] which identified Support Vector Machine (SVM) as the most popular machine learning technique for image based road crack detection.

The potential of Deep Learning DL) based approaches for automated pavement distress detection has been demonstrated through some studies in recent literature [27,35]. Some [27] employed DL on large street-view roadway image database to detect no-crack surfaces so that the street view image database can be reduced in size for further processing. A vehicle equipped with forward-looking 360-degree camera, Global Navigation Satellite System (GNSS), Inertial Measurement Unit (IMU), and laser scanners was used as a mobile mapping system to collect about 9712 images. All DL-related operations were performed with NVIDIA Deep Learning GPU Training System (DIGITS) version 2.0 software, an open-source web-based interactive DL GPU training system that can be used to train Deep Convolutional Neural Networks (DCNNs) with a few mouse clicks. DIGITS integrates the Caffe DL framework from the Berkeley Learning and Vision Center, and it supports GPU acceleration using cuDNN, a GPU-accelerated library of primitives for DCNNs, to massively reduce training time. Some [27] concluded that DL has significant potential in automated pavement crack detection and classification, and that future years will see the development of commercial DL-based road asset analysis systems.

Feng et al. [11] proposed a deep active learning strategy for civil infrastructure defect detection and classification where they used a deep residual network (ResNet) to train a small set of images with defect labels and use this low-accuracy defect detector to filter out many non-defect images. Only images that are difficult to classify would then be sent to human experts for ground truth labels. Thus, this approach is expected to significantly reduce time and resources involved in labeling all the samples before analysis [11]. Xie et al. [35] demonstrated the potential for DL-based pavement crack detection by applying ConvNets on a dataset of 500 images of size 3264×2448, collected using a low-cost smartphone.

3. Deep convolutional neural networks and transfer learning

Deep Neural Networks DNNs), more commonly referred to as Deep Learning DL), employ deep NN architectures to automatically learn hierarchy of features from raw input data without the need for feature engineering [16]. Loosely inspired by how the mammalian brain uses different areas of the cortex to abstract different levels of features when given an input percept, deep learning methods are characterized by deep architectures with several hidden layers that allow them to learn many levels of abstraction, as opposed to shallow architectures with 1 or 2 hidden layers [35]. A common example cited to explain the DL mechanism is how an object is perceived and learnt) by the primate visual system by processing the information through a sequence of transformations and representations: edge, primitive shape, and more complex features. Hinton et al. [16] proposed Deep Belief Networks (DBNs) with a new unsupervised training method called the layer-wise-greedy-training, which gave rise to the popularity of deep learning methods. Going beyond traditional machine learning and artificial intelligence approaches that depend on human-crafted features, DL technologies, with their ability for automatic high-level feature abstraction combined with exponential growth of data, have already produced breakthrough results in Natural Language Processing (NLP), speech recognition, computer vision, and image analysis. There are four main DL architectures, namely the Restricted Boltzman Machines (RBMs), DBNs, Autoencoder (AE), and Deep Convolutional Neural Networks (DCNNs or Deep ConvNets) [20].

DCCNs have shown to be highly effective in processing visual data, such as images and videos. DCNNs take raw input data at the lowest level and transforms them by processing them through a sequence of basic computational units to obtain representations that have intrinsic values for classification in the higher layers [3,14]. A DCNN typically consists of three layer types (Fig. 1): convolution layers, subsampling layers, and fully connected layers. A convolutional layer is parametrized by the number of channels, kernel size, stride factor, border mode, and the connection table. The convolution layer takes the input image and applies convolution filter on it to produce the output image or the filter response. Multiple convolutional layers are used to take into consideration the spatial dependencies among image pixels. The subsampling layer is used to make the neural network more invariant and robust. The max-pooling method is commonly used for the subsampling layer as it has shown to lead to faster convergence and better generalization. It is common to use multiple fully-connected layers after several rounds of convolution and the resulting structure of the last convolutional layer is flattened before connecting to the following fully-connected layer.

In addition to their superior classification and regression performance, the interpretability of DCNNs are becoming an appealing feature. Visualizing first-layer weights of a trained DCNN has been a widely adopted practice to understand what has been learned by the network. The cleanness of features learned by the first DCNN layer is an important indication of how well the network is trained. Another commonly practiced visualization technique is plotting and viewing activations of the network during the forward pass for some given images. For implementation of DCNNs that involve significant amount of parallelism in computations, a number of popular open-source DL frameworks exist such as Caffe (developed by the Berkeley Vision and Learning Center), The Microsoft Cognitive Toolkit, Google's TensorFlow, Theano Framework (with a Python interface), Torch Framework, dmlc MXnet, Chainer, Keras, etc.

DCNNs typically require large annotated image datasets to achieve high predictive accuracy. However, in many domains, acquisition of such data is difficult and labeling them is costly. In light of these challenges, the use of 'off-the-shelf' DCNN features of well-established DCNNs such as VGG-16, AlexNet, and GoogLeNet pre-trained on large-scale annotated natural image datasets (such as ImageNet) have been shown to be very useful for solving cross domain image classification problems through the concept of transfer learning and fine-tuning [25]. In DCNN, representations learned at different layers of the network correspond to different levels of abstractions present in the input images. The initial layers extract edges and color information while the latter layer filters encode shape and texture. The idea behind transfer learning is that it is cheaper and efficient to use deep learning models trained on "big data" image datasets (like ImageNet) and "transfer" their learning ability to new classification scenario rather than train a DCNN classifier from scratch [4]. With adequate fine-tuning, pre-trained DCNN has been shown to outperform even DCNN trained from scratch for some medical imaging applications [25,29].

4. Experimental study

The goal of this research study is to develop a simplified vision-based pavement crack detection system by using a pre-trained deep learning model to detect cracks from pavement images through transfer learning and domain adaptation approach. First, we discuss the pavement images dataset which come from an open-source, online pavement performance repository. The overall methodology is then described which involves the use of VGG-16 DCNN, pre-trained on massive ImageNet database that contains millions of natural images, to vectorize the labeled pavement images and then train a machine learning classifier to predict the labels. Various experiments are conducted to identify the best-performance crack detection models. The experiments and the results are discussed in detail in the end.

4.1. Dataset

We used a subset of the pavement distress images dataset from the Federal Highway Administration's FHWA's) Long-Term Pavement Performance LTPP) program which contains research quality pavement performance information for over 2500 test sections on in-service highway pavements located throughout the United

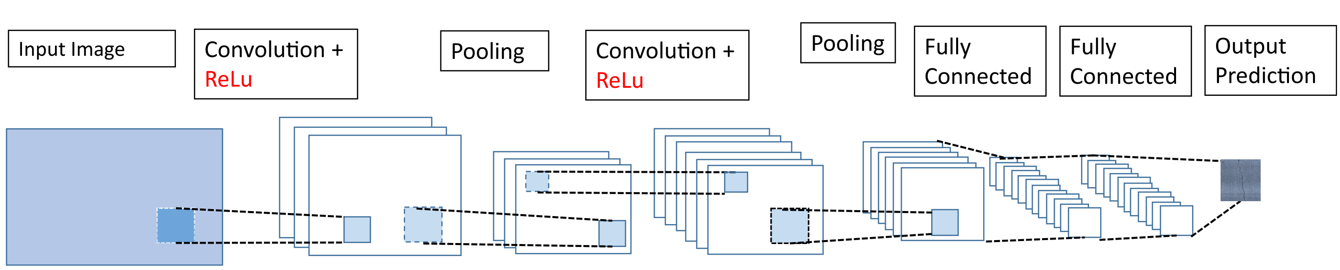

Fig. 1. Schematic of convolutional neural network architecture.

States and Canada [10]. According to LTPP InfoPave, these pavement images were recorded on 35-mm (1.4-in) black-and-white films (between 2002 and 2004) and were then digitized in the office, with each digital image resulting in 2048 × 3072 pixels in dimension [10]. Sample HMA-surfaced and PCC-surfaced pavement images from the FHWA/LTPP database are displayed in Fig. 2.

We used a total of 1056 pavement images (an unspecified combination of HMA-surfaced and PCC-surfaced pavement images) from the FHWA/LTPP database to evaluate the performance of pre-trained DNNs explored in this study. Since the initial and primary goal of this study was crack detection, the images were manually labelled to fall into two separate categories: Crack ('1') or No_Crack ('0'). Among the 1056 pavement images, 337 images had low, medium, or high-severity cracks ('1') and 719 images had no cracks ('0'). Both longitudinal and transverse cracks were considered although most of the crack images included mainly transverse cracks. The number of instances used in our experiments for this binary classification problem are summarized in Table 1. It can be observed that this is a challenging machine learning problem since there are relatively fewer instances to learn from for a real-world classification problem that is far from simple. It should also be noted that the pavement images used in our experiments contained a variety of artifacts, including but not limited to, lane markings, oil stains, non-uniform background caused by varying surface textures, dirt, shadow, foreign objects, etc. making this a very challenging image classification problem. The fact that we included both HMA-surfaced and PCC-surfaced pavement images in the same dataset makes the automated crack detection task even harder.

Table 1
Details of the pavement images dataset for crack detection.

	Crack	No_Crack	Total
Number of training samples	250	510	760
Number of validation samples	28	56	84
Number of testing samples	59	153	212
Total	337	719	1056

4.2. Methodology

We used the Keras deep learning framework [8] that includes pre-trained deep learning models made available alongside weights within Keras Applications. Specifically, we used the Keras implementation of the VGG-16 model (16-layer DCNN developed by the Visual Geometry Group (VGG) at the University of Oxford), with weights pre-trained on ImageNet database in this study [23,26]. The ImageNet database, built upon the hierarchical structure of WorldNet, contains more than 3.2 million cleanly annotated images distributed over 5247 categories [9]. Since the pre-trained VGG-16 model has learned to extract features from images that can distinguish one image class from another, they have shown to achieve excellent performance even when applied to image recognition and classification datasets in other domains [23,26].

The VGG-16 DCNN model or the VGGNet, involving 144 million parameters, contains 16 convolutional layers with very small receptive fields 3 × 3), five max-pooling layers of size 2 × 2 for carrying out spatial pooling, followed by three fully-connected layers, with the final layer as the soft-max layer [26]. Rectification nonlinearity (ReLu) activation is applied to all hidden layers. The model also uses dropout regularization in the fully-connected layers. A schematic of the VGG-16 architecture trained on the ImageNet database is shown in Fig. 3. When the fully-connected classifier (or the bottleneck layer) is removed from the pre-trained VGG-16 network, it can be used as a deep feature generator for producing semantic image vectors for our pavement images. These semantic image vectors can then be trained and tested using another classifier (like Neural Networks [NN], Support Vector Machine [SVM], Random Forest [RF], etc.) for predicting the labels.

4.3. Experiments, results, and discussion

In this study, we evaluate and analyze the performance of DCNN transfer learning approach from non-pavement to pavement image domain for crack detection. The truncated VGG-16 DCNN is used as deep feature generator for our pavement images. We train only the final classifier layer using ImageNet pre-trained VGG-16 DCNN features for the extracted semantic image vectors. The overall framework of the proposed methodology is presented in Fig. 4.

Various experiments were conducted to assess the pavement crack detection performance of the pre-trained VGG-16 DCNN network with transfer learning. As shown in Fig. 4, the experiments described in this section followed a standard process:

a) Preprocess raw pavement digital images sampled from the FHWA/LTPP database by uniformly eliminating the outer edges of the images (in the vertical direction) that contain views of the shoulder and/or the centerline. This reduced the dimension of the images from 3072 × 2048 pixels to 3072 to 1425 pixels.
b) Run the labeled pavement images dataset (categorized into training, validation, and test set) through the truncated VGG-16 DCNN network to generate image vectors;
c) Train a new classifier (NN, SVM, RF, ERT, and LR) using the training set image vectors to predict the labels;
d) Predict test set labels using the trained classifier in step b and using the test set image vectors generated in step a;
e) Report crack detection performance metrics: training performance curves (where applicable), confusion matrix, accuracy, precision, recall, F-value, Cohen's Kappa Score, etc.

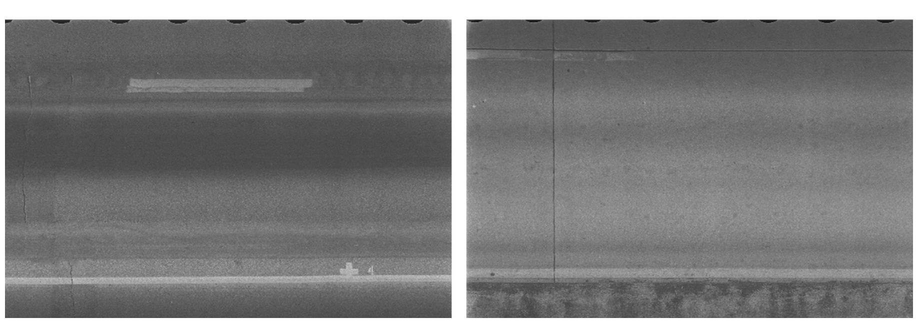

Fig. 2. Sample pavement images from the FHWA/LTPP database: HMA-surfaced pavement image (left); PCC-surfaced pavement image (right).

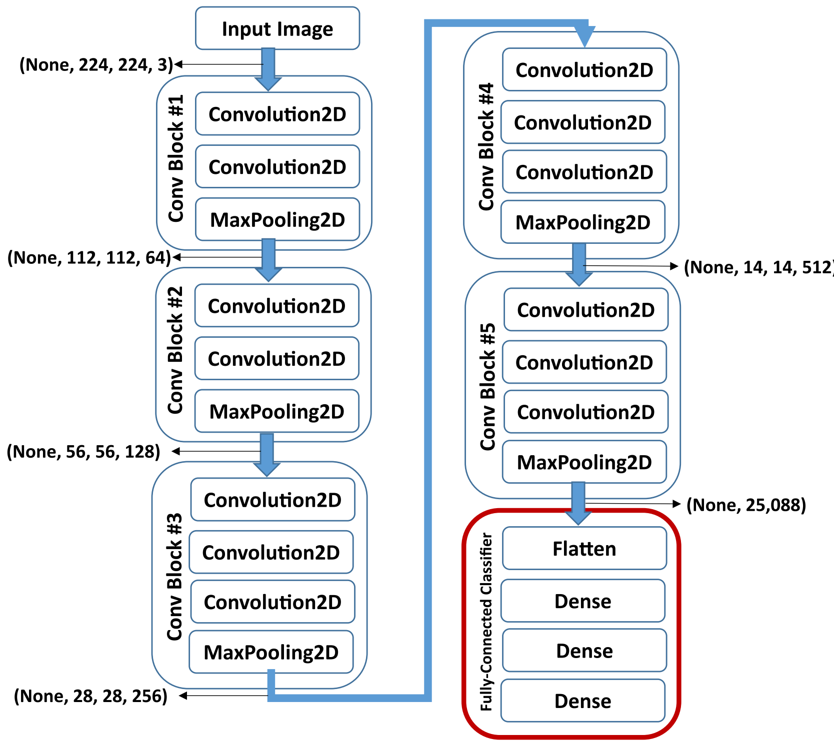

Fig. 3. A schematic of the VGG-16 Deep Convolutional Neural Network (DCNN) architecture trained on ImageNet database.

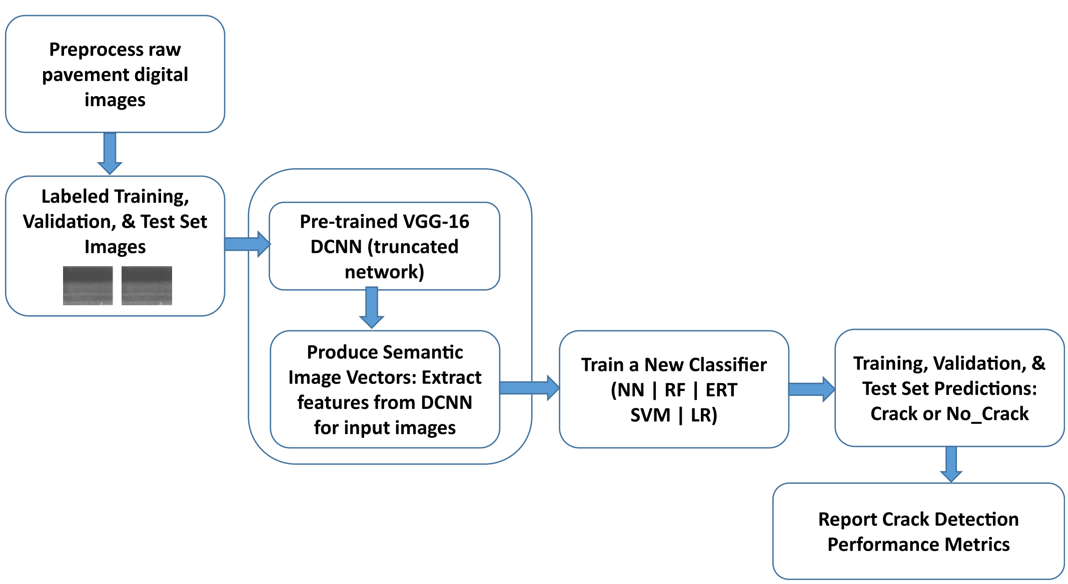

Fig. 4. Overall framework of the proposed methodology.

For the final classifier layer, we employ and compare specific machine learning classifiers that have shown promising results in previous pavement crack detection studies, such as NN, SVM, and RF. Apart from these, we also included Logistic Regression LR) and Extremely Randomized Tree ERT) [13] in our investigations. To control the experimental test factorial within reasonable limits, parameter values of the individual classifiers were not extensively varied. Rather, specific parameters of the classifiers, thought to influence the prediction accuracy, were varied within reasonable ranges. For instance, experiments with the RF and ERT classifiers involved varying the number of trees from 50 to 4000. Similarly, SVM related experiments were carried out by comparing 'linear', 'rbf', and 'sigmoid' kernel types with C (penalty parameter of the error term) values between 1.0 and 5.0. With LR classifier, 'L1' and 'L2' norms used in the penalization were varied and compared. The following designations are used for the base model configurations evaluated in this study for automated pavement crack detection (parameters varied during the experiments are enclosed in parenthesis; other parameters were fixed at their default values):

- TL + NN: Single NN layer classifier trained on ImageNet pre-trained VGG-16 DCNN features for pavement images (optimizers: 'adam' (AM), 'adadelta' (AD), stochastic gradient descent or 'sgd' (SG), 'adamax' (AX) as implemented in Keras)

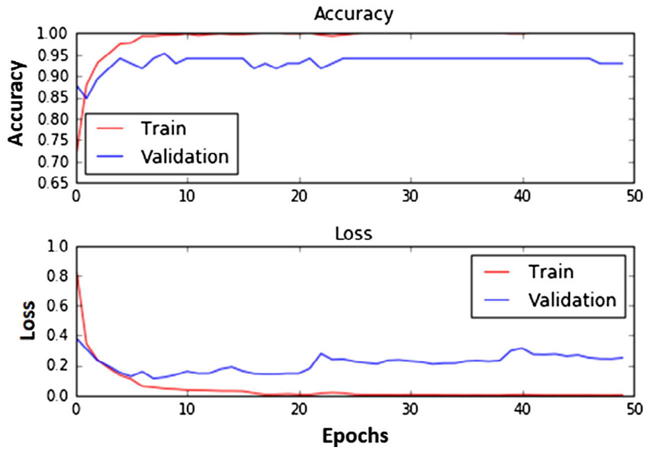

Fig. 5. Traces of training and validation accuracy (top) and loss (bottom) during training single-layer Neural Network (NN) with ImageNet pre-trained VGG-16 DCNN deep image extractors (TL + NN[optimizer: adam]) for pavement crack detection.

- TL + RF: Random Forest classifier trained on ImageNet pre-trained VGG-16 DCNN features for pavement images (number of trees: 50, 100, 200, 300, 400, 500, 1000, 5000)
- TL + ERT: Extremely Random Trees classifier trained on ImageNet pre-trained VGG-16 DCNN features for pavement images (number of trees: 50, 100, 200, 300, 400, 500, 1000, 5000)
- TL + SVM: Support Vector Machines classifier trained on ImageNet pre-trained VGG-16 DCNN features for pavement images (C = 1.0, C = 5.0; kernel types = 'linear', 'rbf', 'sigmoid')
- TL + LR: Logistic Regression classifier trained on ImageNet pre-trained VGG-16 DCNN features for pavement images (penalty = 'L1', 'L2')

Deep transfer learning was implemented with the Keras [8] deep learning framework, using a Intel® Core™ i7-5600U CPU on 64-bit Windows® 10 OS. The ImageNet pre-trained VGG-16 DCNN implemented within Keras Applications takes in a default image input size of 224×224. Other image dimensions not smaller than 48×48 are valid as well. As seen in Fig. 3, a 224×224 image input dimension results in 25,088 deep transfer learning features that serve as inputs to the final classifier. Our pre-processed images are of dimensions 3072×1425 pixels. However, this input size will result in a massively sparse feature matrix after running it through the ImageNet pre-trained VGG-16 DCNN, not amenable to CPU computations. Therefore, the input image size was set to 1000×500 which maintains the original aspect ratio of the pre-processed images. This resulted in 238,080 ImageNet pre-trained VGG-16 DCNN features as inputs to the final classifier.

For the single-layer NN classifier implemented in Keras, 256 neurons were used in the hidden layer and a dropout value of 0.5 was used. The 'relu' activation was used in the hidden layer and 'softmax' activation in the output layer. The image batch size was set to 32 and all models were trained for up to 50 epochs. The early stopping criteria was used with the final model being the one with low validation loss. Traces of training and validation accuracy and loss (Mean-Squared-Error or MSE) during the training of single-layer NN classifier (optimizer: 'adam') with ImageNet pre-trained VGG-16 DCNN deep image features are shown in Fig. 5.

For all other classifiers, standard implementations in 'scikit-learn' machine learning library in Python were used. A comparison of pavement crack detection results using various deep transfer learning models is presented in Table 2. Numbers in bold indicate the best performance model configuration for each classifier category. Classification performance metrics include the accuracy, precision, recall, F1-score and Cohen's Kappa score [5]. The five selected best-performance classifiers for further investigation

Table 2
Comparison of pavement crack detection results on test set images using various deep transfer learning models. Numbers in bold indicate the best performance model under each classifier category.

Final Classifier	Model	Accuracy	Precision	Recall	F1-score	Cohen's Kappa Score
Single-layer Neural Network (NN)	**TL + NN (AM)**	**0.90**	**0.90**	**0.90**	**0.90**	**0.742**
	TL + NN (AD)	0.83	0.84	0.83	0.84	0.591
	TL + NN (SG)	0.86	0.86	0.86	0.86	0.658
	TL + NN (RM)	0.84	0.84	0.84	0.84	0.601
	TL + NN (AX)	0.88	0.88	0.88	0.88	0.695
Random Forest (RF)	TL + RF (50)	0.82	0.82	0.83	0.82	0.529
	TL + RF (100)	0.83	0.82	0.83	0.82	0.533
	TL + RF (200)	0.82	0.82	0.83	0.81	0.506
	TL + RF (300)	**0.86**	**0.86**	**0.86**	**0.85**	**0.622**
	TL + RF (400)	0.84	0.84	0.84	0.83	0.570
	TL + RF (500)	0.84	0.84	0.84	0.83	0.549
	TL + RF (1000)	0.85	0.85	0.85	0.84	0.596
	TL + RF (4000)	0.85	0.85	0.85	0.84	0.580
Extremely Randomized Trees (ERT)	TL + ERT (50)	0.84	0.83	0.84	0.83	0.564
	TL + ERT (100)	0.86	0.87	0.86	0.85	0.618
	TL + ERT (200)	0.85	0.86	0.85	0.84	0.586
	TL + ERT (300)	**0.87**	**0.87**	**0.87**	**0.86**	**0.629**
	TL + ERT (400)	0.85	0.85	0.85	0.84	0.576
	TL + ERT (500)	0.86	0.86	0.86	0.85	0.602
	TL + ERT (1000)	0.85	0.86	0.85	0.84	0.581
	TL + ERT (4000)	0.85	0.86	0.85	0.84	0.586
Support Vector Machine (SVM)	TL + SVM (L,1)	0.87	0.87	0.87	0.86	0.649
	TL + SVM (L,5)	0.87	0.87	0.87	0.86	0.649
	TL + SVM (R,1)	0.86	0.86	0.86	0.85	0.622
	TL + SVM (R,5)	**0.87**	**0.87**	**0.87**	**0.87**	**0.656**
	TL + SVM (S,1)	0.81	0.82	0.81	0.78	0.429
	TL + SVM (S,5)	0.81	0.80	0.81	0.79	0.459
Logistic Regression (LR)	TL + LR (L1)	0.86	0.86	0.86	0.85	0.620
	TL + LR (L2)	**0.88**	**0.88**	**0.88**	**0.87**	**0.674**

Bold indicate the best performance model configuration and results for each classifier category.

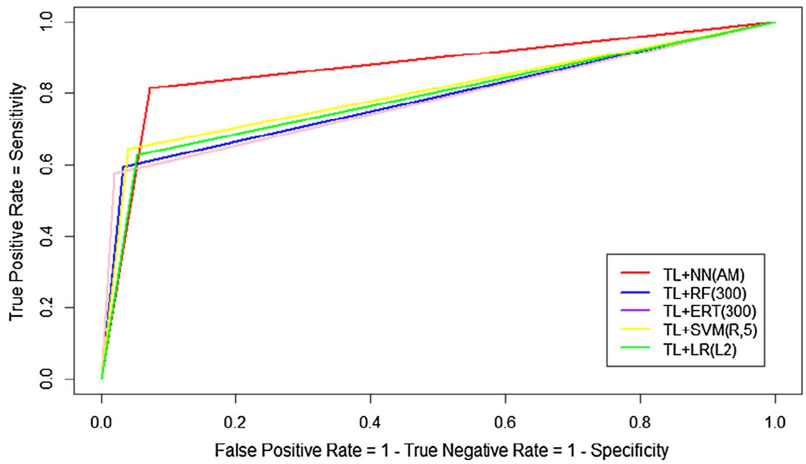

Fig. 6. ROC curves using select classifiers trained on ImageNet pre-trained VGG-16 DCNN features for pavement images.

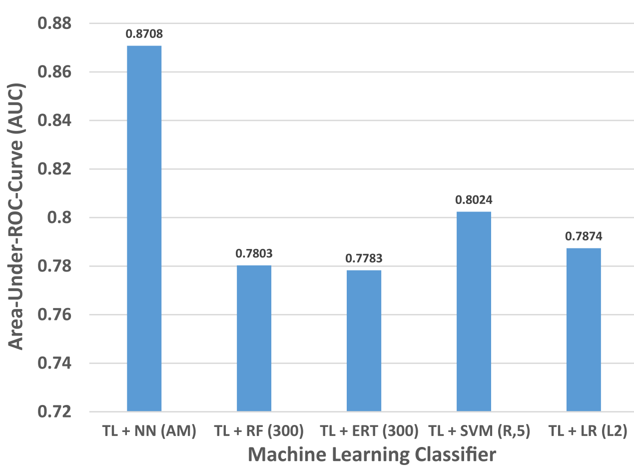

Fig. 7. Comparison of AUC values for select classifiers trained on ImageNet pre-trained VGG-16 DCNN features for pavement images.

include: TL + NN (AM), TL + RF (300), TL + ERT (300), TL + SVM (R,5), and TL + LR (L2). Among these 5 classifiers, the single-layer NN classifier trained on ImageNet pre-trained VGG-16 DCNN features seem to yield the best performance, followed by SVM and LR classifiers. This is further confirmed by the ROC curves and the area-under-the-ROC-curve (AUC) values for the select classifiers in Fig. 6 and Fig. 7, respectively, as well as a comparison of other evaluation metrics from the confusion matrices (Table 3).

We are of the opinion that the pavement images datasets used in this study are significantly more complex than those used in most previous studies. Apart from the complexities introduced by lane markings (thick white lines along the horizontal edges and some times in the middle of the image running vertically), embossed numbers on the pavement surface, oil spills, and the presence of other distresses (such as raveling in HMA-surfaced pavements) which are all common sources of false positives in this domain, a significantly higher order of complexity is also introduced by trying to train a classifier on combined HMA-surfaced and PCC-surfaced images that have different surface characteristics. Thus, an AUC (see Fig. 7) of 0.87 for TL + NN (AM) seems satisfactory for this study, although it could be further improved. Examples of misclassified crack images (see Fig. 8) of both false positives and false negatives tend to indicate that the best-

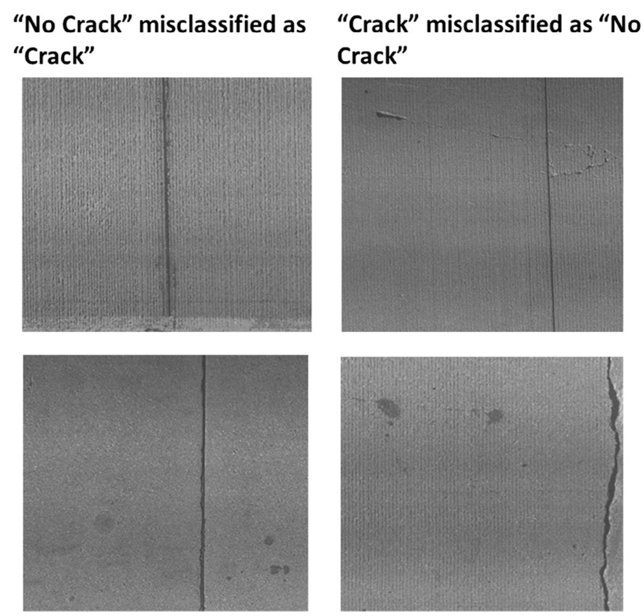

Fig. 8. Examples of misclassified crack images (only patches of the images of are shown) of both false positives (left) and false negatives (right).

Table 3
Summary of classification performance metrics for select classifiers trained on ImageNet pre-trained VGG-16 DCNN features for pavement images.

Model	Positive Predicted Value (PPV)	Negative Predicted Value (NPV)	Prevalence	Detection Rate	Detection Prevalence	Balanced Accuracy
TL + NN (AM)	0.93	0.81	0.72	0.67	0.72	0.87
TL + RF (300)	0.86	0.88	0.72	0.70	0.81	0.78
TL + ERT (300)	0.86	0.92	0.72	0.71	0.83	0.78
TL + SVM (R,5)	0.88	0.86	0.72	0.69	0.79	0.80
TL + LR (L2)	0.87	0.82	0.72	0.68	0.79	0.79

performance classifier (TL + NN [AM]) has failed to learn to distinguish cracks from joints in PCC-surfaced pavements.

5. Conclusions and future directions

In recent years, Deep Learning, a generalized form of DNN algorithms that can learn very complex mappings between inputs and outputs directly from the data, has achieved huge success in diverse fields such as automatic speech recognition, image recognition, Natural Language Processing (NLP), drug and materials discovery, etc. However, the large number of hidden neurons and layers used in DNNs result in computationally-intensive matrix and vector computations involving millions of parameters, requiring the use of high-performance computing systems. Also, it is practically impossible to get labeled "big data" samples in many domains to be able to train an entire DNN from scratch. In such situations, the use of a pre-trained deep learning model and fine-tuning it to the novel task at hand with smaller datasets, has shown to be successful across domains.

In this paper, we explored and evaluated the deep transfer learning approach, *viz.*, use deep learning models trained on "big data" image datasets (ImageNet) and "transfer" their learning ability to automatic pavement crack detection from digitized pavement surface images obtained from the FHWA/LTPP database. The following are some significant findings:

- The overall approach of using pre-trained deep convolutional neural network model for cross-domain image classification, already having demonstrated its feasibility in many medical imaging studies, was shown to be successful for vision-based automated pavement crack detection.
- The Keras deep learning framework provides a nearly ready-to-use platform for easy implementation of deep transfer learning approach for cross-domain image classification.
- Among the different machine learning classifiers tested, a single-layer neural network classifier (with 'adam' optimizer) trained on ImageNet pre-trained VGG-16 DCNN features yielded the best performance.
- The pavement images datasets used in this study are significantly more complex than those used in most previous studies. Apart from the common sources of false positives in this domain, a significantly higher order of complexity was also introduced by trying to train a classifier on combined HMA-surfaced and PCC-surfaced images that have different surface characteristics. This was done so to test the robustness of the proposed approach. Surprisingly enough, the deep transfer learning approach employed in this study was insensitive to surface color and texture variations and thus produced higher than expected true positives and true negatives for both pavement types.
- Based on the misclassified crack images (see Fig. 8) of both false positives and false negatives, it appears that even the best-performance classifier (TL + NN [AM]) has failed to learn to distinguish cracks from joints in PCC-surfaced pavements. It is believed that this problem could be alleviated by increasing the number of training samples (containing joints and cracks) in the transfer learning process and by training a separate classifier for crack detection in PCC-surfaced pavements.

Apart from training separate deep transfer learning classifiers for HMA-surfaced and PCC-surfaced images, training of classifiers to predict pavement crack severities (low, medium, and high) should also be pursued as part of future research efforts. Fine-tuning of the last one or two convolutional blocks of the DCNN for the classification task at hand in combination with deep transfer learning is another established method to pursue to increase the accuracy of pavement crack detection. Finally, data augmentation, a common technique for generating additional data without introducing extra labeling costs, was not used in this study. The common data augmentation methods include mirroring, scaling, and rotation. Aggressive data augmentation methods will be pursued as part of future research efforts to improve the accuracy of the computer-vision based automatic pavement crack detection classifier.

Acknowledgments

This work is supported in part by the following grants: AFOSR award FA9550-12-1-0458; NIST award 70NANB14H012; NSF award CCF-1409601; DOE awards DE-SC0007456, DE-SC0014330. The authors would like to gratefully acknowledge Dr. Ayush Singhal, R&D branch of Contata Solutions, LLC, Minneapolis, Minnesota, for his sincere contributions and many fruitful discussions in the course of preparing this article.

References

[1] ASCE, American Society of Civil Engineers (ASCE) 2017 Infrastructure Report Card, Roads, 2017.
[2] A. Ayenu-Prah, N. Attoh-Okine, Evaluating pavement cracks with bidimensional empirical mode decomposition, EURASIP J. Adv. Signal Process. 2008 (2008) 861701, https://doi.org/10.1155/2008/861701.
[3] S. Bai, Growing random forest on deep convolutional neural networks for scene categorization, Expert Syst. Appl. 71 (2017) 279–287, https://doi.org/10.1016/j.eswa.2016.10.038.
[4] Y. Bar, I. Diamant, L. Wolf, S. Lieberman, E. Konen, H. Greenspan, Chest pathology detection using deep learning with non-medical training, in: 2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI). Presented at the 2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI), 2015, pp. 294–297. 10.1109/ISBI.2015.7163871
[5] N.J.-M. Blackman, J.J. Koval, Interval estimation for Cohen's kappa as a measure of agreement, Statist. Med. 19 (2000) 723–741, https://doi.org/10.1002/(SICI)1097-0258(20000315)19:5<723::AID-SIM379>3.0.CO;2-A.
[6] S. Chambon, J.-M. Moliard, Automatic road pavement assessment with image processing: review and comparison, Int. J. Geophys. 2011 (2011) 20, https://doi.org/10.1155/2011/989354.
[7] H.D. Cheng, J.-R. Chen, C. Glazier, Y.G. Hu, novel approach to pavement cracking detection based on fuzzy set theory, J Comput Civil Eng 13 (1999).
[8] F. Chollet, keras. GitHub, 2015.
[9] J. Deng, W. Dong, R. Socher, L. Li, K. Li, L. Fei-fei, Imagenet: A large-scale hierarchical image database, in: In CVPR, 2009.
[10] G.E. Elkins, T. Thompson, B. Ostrom, A. Simpson, B. Visintine, Long-Term Pavement Performance Information Management System: Pavement Performance Database User Reference Guide (Final Report No. FHWA –RD-03-088 (revision)). Federal Highway Administration (FHWA), Washington, D. C., 2016.
[11] C. Feng, M.-Y. Liu, T.-Y. Lee, Deep Active Learning for Civil Infrastructure Defect Detection and Classification, in: 2017 International Workshop on Computing in Civil Engineering. Presented at the 2017 International Workshop on Computing in Civil Engineering (IWCCE 2017), American Society of Civil Engineers (ASCE), Seattle, Washington, 2017.
[12] G. Flintsch, K. McGhee, Quality Management of Pavement Condition Data Collection (No. NCHRP Synthesis 401). Transportation Research Board, Washington, D.C., 2009.
[13] P. Geurts, D. Ernst, L. Wehenkel, Extremely randomized trees, Mach Learn 63 (2006) 3–42, https://doi.org/10.1007/s10994-006-6226-1.
[14] H. Goh, N. Thome, M. Cord, J.H. Lim, Learning Deep Hierarchical Visual Feature Coding, IEEE Trans. Neural Networks Learn. Syst. 25 (2014) 2212–2225, https://doi.org/10.1109/TNNLS.2014.2307532.
[15] K. Gopalakrishnan, Advanced Pavement Health Monitoring and Management [WWW Document], IGI Global Videos, URL, 2016. http://www.igi-global.com/video.aspx?ref=advanced-pavement-health-monitoring-management&titleid=137625 (accessed 5.9.17).
[16] G.E. Hinton, S. Osindero, Y.-W. Teh, A fast learning algorithm for deep belief nets, Neural Comput. 18 (2006) 1527–1554, https://doi.org/10.1162/neco.2006.18.7.1527.
[17] M. Kaseko, Z.-P. Lo, S. Ritchie, Comparison of traditional and neural classifiers for pavement-crack detection, J. Transp. Eng. 120 (1994) 552–569, https://doi.org/10.1061/(ASCE)0733-947X(1994) 120:4(552).
[18] C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, P. Fieguth, A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure, Adv. Eng. Inf. Infrastruct. Comput. Vision 29 (2015) 196–210, https://doi.org/10.1016/j.aei.2015.01.008.
[19] J. Laurent, D. Lefebvre, E. Samson, Development of a new 3D transverse laser profiling system for the automatic measurement of road cracks, in: 6th

Symposium on Pavement Surface Characteristics. ARRB Group Limited, Portoroz, Slovenia, 2008.
[20] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, F.E. Alsaadi, A survey of deep neural network architectures and their applications, Neurocomputing 234 (2017) 11–26, https://doi.org/10.1016/j.neucom.2016.12.038.
[21] H. Oliveira, P.L. Correia, Automatic Road Crack Detection and Characterization, IEEE Trans. Intell. Transp. Syst. 14 (2013) 155–168, https://doi.org/10.1109/TITS.2012.2208630.
[22] H. Oliveira, P.L. Correia, Automatic road crack segmentation using entropy and image dynamic thresholding, in: 2009 17th European Signal Processing Conference. Presented at the 2009 17th European Signal Processing Conference, IEEE, Glasgow, UK, 2009, pp. 622–626.
[23] S. Pal, Transfer Learning and Fine Tuning for Cross Domain Image Classification with Keras. GitHub: Transfer Learning and Fine Tuning for Cross Domain Image Classification with Keras, 2016.
[24] Pathway Services Inc., Automated Crack Detection and Analysis [WWW Document], Automated Road and Pavement Condition Surveys, URL, 2014. http://www.pathwayservices.com/crack_detection.shtml (accessed 6.13.17).
[25] H.-C. Shin, H.R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, R.M. Summers, Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning, IEEE Trans. Med. Imaging 35 (2016) 1285–1298.
[26] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, in: ICLR 2015. Presented at the 3rd International Conference on Learning Representations, San Diego, CA, 2014.
[27] L. Some, Automatic image-based road crack detection methods, KTH Royal Institute of Technology, Stockholm, Sweden, 2016.
[28] P. Subirats, J. Dumoulin, V. Legeay, D. Barba, Automation of Pavement Surface Crack Detection using the Continuous Wavelet Transform, in: 2006 International Conference on Image Processing. Presented at the 2006 International Conference on Image Processing, 2006, pp. 3037–3040. doi:10.1109/ICIP.2006.313007.
[29] N. Tajbakhsh, J.Y. Shin, S.R. Gurudu, R.T. Hurst, C.B. Kendall, M.B. Gotway, J. Liang, Convolutional neural networks for medical image analysis: full training or fine tuning?, IEEE Trans Med. Imaging 35 (2016) 1299–1312, https://doi.org/10.1109/TMI.2016.2535302.
[30] Y. Tsai, Z. Wang, Development of an Asphalt Pavement Raveling Detection Algorithm Using Emerging 3D Laser Technology and Macrotexture Analysis (Final Report No. NCHRP IDEA Project 163), NCHRP IDEA. Transportation Research Board, Washington, D.C., 2015.
[31] Y.-C. Tsai, V. Kaul, R.M. Mersereau, Critical assessment of pavement distress segmentation methods, J. Transp. Eng. 136 (2009) 11–19.
[32] W.R. Vavrik, L.D. Evans, J.A. Stefanski, S. Sargand, PCR Evaluation – Considering Transition from Manual to Semi-Automated Pavement Distress Collection and Analysis, Ohio Department of Transportation, Columbus, OH, 2013.
[33] K.C.P. Wang, Q. Li, W. Gong, Wavelet-Based Pavement Distress Image Edge Detection with À Trous Algorithm, Transp. Res. Rec.: J. Transp. Res. Board (2007).
[34] K.C.P. Wang, Q.J. Li, G. Yang, Y. Zhan, Y. Qiu, Network level pavement evaluation with 1 mm 3D survey system, J. Traffic Transp. Eng. (English Edition) 2 (2015) 391–398, https://doi.org/10.1016/j.jtte.2015.10.005.
[35] D. Xie, L. Zhang, L. Bai, Deep learning in visual computing and signal processing, Appl. Comput. Intell. Soft Comput. 2017 (2017) e1320780, https://doi.org/10.1155/2017/1320780.
[36] L. Ying, E. Salari, Beamlet transform-based technique for pavement crack detection and classification, Comput.-Aided Civil Infrastruct. Eng. 25 (2010) 572–580, https://doi.org/10.1111/j.1467-8667.2010.00674.x.
[37] D. Zhang, Q. Li, Y. Chen, M. Cao, L. He, B. Zhang, An efficient and reliable coarse-to-fine approach for asphalt pavement crack detection, Image Vis. Comput. 57 (2017) 130–146, https://doi.org/10.1016/j.imavis.2016.11.018.
[38] Q. Zou, Y. Cao, Q. Li, Q. Mao, S. Wang, CrackTree: Automatic crack detection from pavement images, Pattern Recogn. Lett. 33 (2012) 227–238, https://doi.org/10.1016/j.patrec.2011.11.004.