This article can be cited as N. T. Huong and N. T. Long, Deep Learning Methods for Classification of Road Defects, International Journal of Artificial Intelligence, vol. 19, no. 1, pp. 178-192, 2021.

Deep Learning Methods for Classification of Road Defects

Nguyen T. Huong[1,3] **and Nguyen T. Long**[2]

[1]Baikal School of BRICS
Irkutsk National Research Technical University, 664074, Irkutsk, Russia

[2]Laboratory of Artificial Intelligence and Machine Learning
Institute of Information Technology and Data Science
Irkutsk National Research Technical University, 664074, Irkutsk, Russia

[3]University of Information and Communication Technology
Thai Nguyen University, 70000, Thai Nguyen, Viet Nam
thuhuongyb@gmail.com, thelongit88@gmail.com

ABSTRACT

Many methods are proposed in the classification of pavement defects through the extraction of features from data and the use of machine learning algorithms to solve the problem. But there are still limitations such as training time, accuracy and the sensitivity of the system to environmental conditions. This research proposes the method of optimizing the automatic classification system of pavement defects based on Convolutional Neural Network, a deep learning method commonly used and highly effective in artificial intelligence and digital image processing. This system is guaranteed to operate stably in limited light conditions, shading and complex-shaped defects. Our experiment is performed with 3 data sets (INESC TEC - Portugal, Irkutsk - Russia, Thai Nguyen - Viet Nam). The data obtained from VGG-16 method is compared with data obtained from the Random Forest algorithm and Support Vector Machine method. The experimental data show that the proposed method allows the Random Forest algorithm to work faster, more stable and gives more accurate results. The result after classification is 97.07 %, 97.62 %, 98.50 %) respectively.

Keywords: computer vision, artificial intelligence, classification of road defects, features extraction, convolutional neural network, deep learning, ROC curves.

Mathematics Subject Classification: 68U10, 68T45.
Computing Classification System: 10010147.10010257, 10010147.10010178, 10010147.10010371.10010382.10010383, 10010147.10010178.10010224.10010245.10010247, 10010520.10010521.10010542.10010294.

1 Introduction

In recent years, deep learning models, especially Convolutional Neural Network (CNN), have been widely in machine learning applications (Dansh, Min, L. and et al., 2017) for classification problems (Yunchao, Wei, Min and et al., 2016). The main advantage of CNN: the process of features extraction and convolutional classes are trained in parallel together. Thus, the parameters of the training model will be adjusted linearly to the final classifier. This gives CNN the advantage of processing time and high sensitivity to data of special nature depending on light, noise as data on road defects. The rapid development of machine learning algorithms and methods has enabled artificial intelligence to be widely applied in all fields (Sinziana and Zack, 2012), (Wei, Ya and Tsai, 2012), (Lekha, Nair and Siddhanth, 2018), such as security, education, surveillance science, health, economics, etc. In recent years, a new development direction of machine learning is deep learning, which is being developed and applied by scientists around the world (Mohamed, Riyaz and Fariza, 2020), (Kim, Yang, Lessmann, T. and V., 2020). With the help of algorithms, the deep learning method will give us an optimal solution, saving both time and economy, yet still ensuring to improve work efficiency and the accuracy of the system. Deep learning is not an abstract term, it is an area of scientific research promised to bring scientific applications into practice to support and enhance intelligent processing systems.

Over the past few years, several studies have been carried out successfully in the field of automatic detection and classification of pavement defects based on machine learning methods. However, there are still many limitations. Assuming that only one type of defect has been detected, the effect of natural conditions on the quality of the acquired database, the resources of hardware and software for analyzing data, etc. as not been overcome yet. Therefore, in this article, it is proposed to perform automatic classification of objects based on deep learning methods to optimize the classification system.

The purpose of this study is to apply deep learning techniques to detect and classify road defects. This is important to overcome the issue of a continuously rising demand with limited road supply. The remainder of this paper is divided as follows: Section 2 discusses the studies that also apply AI for road defects detection and classification. Section 3 describes the methodology of Deep Learning technology to build a system to classify road defects. This section is subdivided into (3.1) Finding Region of Interest (ROI) and features extraction and (3.2) Apply Deep Learning for road defects classification. Section 4 shows and analysis the results of experiment. Section 5 provides a conclusion of this paper review.

2 Related work

Automating the processes to handle the transportation industry issues has attracted a lot of attention from researchers (Ali and Hamzeh, 2019), (Abduljabbar, Dia and Liyanage, 2019), (Dia, 2001). One of the important factors affecting the transportation economy is road surface defects. Indeed, supporting infrastructure for reliable operations and cost savings is an important task. In order to maintain and plan road repairs, road companies need accurate and timely information about road surface defects. There are millions of kilometers of roads in the world that need to be checked every year. In the past, periodic testing was done manually by engineers. This method took a lot of time.

Available datasets have been exploited for years to implement an approach based on machine learning, using computer vision-based classification features and classification algorithms such as SVM to detect cracks (Hoang and Nguyen, 2018) and potholes, Random Forest to classify road defects (Nguyen, Nguyen and Greglea, 2018), (Nguyen and Nguyen, 2019). Gradient-based methods usually trap in local optimum so that training process becomes a big challenge. Chaotic time series have complex behavior, and their analysis is difficult and these commonly used for modeling natural behaviors. Six machine learning algorithms are investigated and implemented (Mobyen, Staffan and et al., 2020), i.e. Artificial Neural Network, Support Vector Machine, Decision Tree, Random Forest, Extra Tree, and Gradient Boosted Trees to classify different Pedestrians events based on Inertial Measurement Unit (IMU) and Global Positioning System (GPS) signals. Three different experiments are conducted to evaluate the performances in terms of the accuracy of the MLs models in different circumstances. In (Danielle, Rebecca and et al., 2018) presented a comparative study of nine state-of-the-art machine learning techniques for the classification of diseases. These are ID3, deep learning, artificial neural networks, nave bayes, logistic regression, partial decision trees, k-nearest neighbor, classification via clustering and voting feature intervals. The authors assessed the machine learning techniques using the following performance metrics: accuracy, precision, and area under the ROC curve. Results show that they can be very beneficial over a wide range of diseases. The proposed method uses two different strategies to make decisions for patients with unknown features (Kiarash and Mehdi, 2018). The proposed method is evaluated by comparing it to some of the most common cost-sensitive learning methods and show superior results in many cases. The best performing MLs models determined by the average accuracy across all experiments are Extra Tree (ET). The proposed method in (Cai, Zhang and et al., 2007) used in different science, such as financial, economic, traffic, and weather. (Stefan, Radu-Emil and et al., 2007) presented aspects concerning two iterative methods in control design: The iterative feedback tuning and the iterative learning control approaches. The theoretical results are validated in the case of PI controllers for servo systems with second-order integral type controlled plants and real-time experiments are included. The combination between two iterative methods and fuzzy control is successful in applications leading to performance enhancement.

Many researchers have demonstrated the advantages of Deep Learning to build applications. An example of that includes transforming the traffic sensors on the road into a smart agent that detects accidents automatically and predicts future traffic conditions (Klugl, Bazzan and Ossowski, 2010). Also, there are many Deep Learning methods used in transport such as ANNs. ANNs can be used for road planning (Dogan and Akgngr, 2013), public transport (Budalakoti, Srivastava and Ote, 2009) Traffic Incident Detection (Wang, Fan and Work, 2016), and predicting traffic conditions (Lv, Duan, Kang and Li, 2014). It is classified into supervised and unsupervised learning methods. Supervised methods include support Vector Machine (SVM), Probabilistic Neural Network (PNN), Radial Basis Network (RBN), K-Nearest Neighbors and Decision Tree, etc. while unsupervised NNs include greedy layer-wise and cluster analysis. To detect pavement defects often based on methods, such as intensity thresholding, edge detection, graph theory, texture analysis, machine learning algorithms, and neural network-based methods. Thresholding algorithms are based on the assumption that cracks are represented by local intensity minima. Authors in (Gopalakrishnan, Khaitan, Choudhary and A., 2017) 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 essence of work in (Zhang, Li, Chen and Cao, 2017) is based on the gray-level difference between crack pixels and background pixels, which has a direct limitation for those cracks without obvious intensity changes with surrounding areas. Depth information as another innate character associating with cracks would be advantageous in the detection when the gray level characteristics are not sufficiently salient. Edge detection techniques include the usage of Canny filters, the Sobel edge detector, and other morphological filters (Wu, Sun, Zhou and Zhang, 2016), (Oliveira, Caeiro and Correia, 2016). With the development of artificial intelligence in recent years, some classification systems are built based on the machine learning algorithms that were introduced into the automatic pavement defects detection field such as support vector machine (Schlotjes, Burrow and Evdorides, 2015), Random Forest (Nguyen et al., 2018), (Nguyen and Nguyen, 2019). With the advent of deep learning technology, Convolutional Neural Networks have started to dominate the field of object detection and recognition.

3 Methodology

3.1 Finding Region of Interest (ROI) and features extraction

Locating the Region of Interest (ROI) (fig.1) is an essential precursor for the feature extraction step. This step identifies the main or interesting portion of the image (or signal) from where the features traits are extracted. The use of computer vision algorithms to identify regions of interest based on image segmentation is proposed. Image quality is the leading factor affecting the classification system of pavement defects. There are two subjective factors (the resolution of the device, the distance of the lens to the road surface) and objectivity (light, noise ...) affecting the quality of the device. amount of image. Thus, a problem raised is to improve the image quality, but not lose information, keeping the local features (but leave local features intact). Therefore, it is necessary to develop an image preprocessing operation to delete environmental interference as much as possible. This is done by the Laplacian filter (Aubry, Hasinoff, Kautz and Durand, 2014). Use the Canny algorithm (Canny, 1986) to detect edges. Apply the morphological method to perfect edges. To make easier to work and to visualize the processed image is transformed into grayscale with size 512×512. All the steps of the preprocessing stage are explained below.

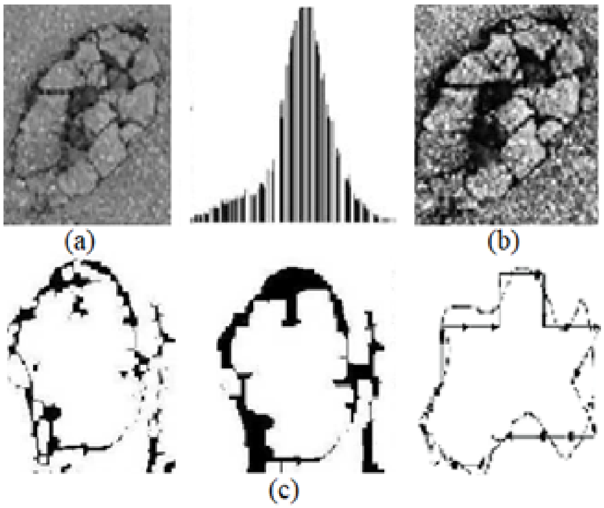

Figure 1: The result of image pre-processing, (a) original image, (b) image after filter step, (c) image of edge detection and morphology step.

Image filter step: The Laplacian of an image highlights regions of rapid intensity change and is an example of a second-order or a second derivative method of enhancement. It is particularly good at finding the fine details of an image. Any feature with a sharp discontinuity will be enhanced by a Laplacian operator. The Laplacian is a well-known linear differential operator approximating the second derivative given by the equation.

$$\Delta^2 f = \frac{\delta^2 f}{\delta x^2} + \frac{\delta^2 f}{\delta y^2}, \tag{3.1}$$

where f denotes the image.

Laplacian filtering is a computationally intensive algorithm. To speed up processing, to approximate the algorithm by the intensity range into a number of samples defined by the parameter. This parameter can be used to balance speed and quality. Random noise is one of the main problems which may affect the results. Due to road surface physical characteristics, all the pictures may be affected by random and uniform noise, which has to be removed. The Laplace filter is commonly used to smooth an image without losing edge precision.

Edge detection based on the Canny algorithm: In images often exist components such as smooth areas, angles, edges, and noise. Edges in an image have an important feature, usually in an object. Therefore, to detect edges in an image, the Canny algorithm is one of the most popular algorithms in image processing. The Canny algorithm consists of five main steps: *(1) Noise reduction* - previous image filter step; *(2) Gradient calculation* - The Gradient calculation step detects the edge intensity and direction by calculating the gradient of the image using edge detection operators. Edges correspond to a change of pixel's intensity. To detect it, the easiest way is to apply filters that highlight this intensity change in both directions: horizontal (x) and vertical (y). The result is almost the expected one, but some of the edges are thick and others are thin. Non-Max Suppression step will help us mitigate the thick ones; *(3) Non-maximum suppression* - Ideally, the final image should have thin edges. Thus, a performance of non-maximum suppression to thin out the edges is needed. The principle is simple: the algorithm goes through all the points on the gradient intensity matrix and finds the pixels with the maximum value in the edge directions; *(4) Double threshold* - The step aims at identifying 3 kinds of pixels: strong, weak, and non-relevant; *(5) Edge Tracking by Hysteresis* - Based on the threshold results, the hysteresis consists of transforming weak pixels into strong ones, if and only if at least one of the pixels around the one being processed is a strong one.

A mathematical morphology method to reduce the possible noise influence: The contour of the object becomes more smooth and continuous after application of the opening operator of the mathematical morphology method. Morphological filtering (MF) is a widely used technique useful in image analysis. Previous porosity and cell detection studies have relied heavily on basic MF operators such as erosion, dilation, opening, and closing to manipulate the geometrical properties of the image. Mathematical morphology can separate clumped objects, estimate the background, fill imperfectly stained regions (in the case of cells) and/or rectify intensity gradients. For this research, the higher magnification images were dilated with a disk structuring element. In addition, any objects containing zero intensity pixels surrounded by one intensity pixels (holes) were filled and any objects touching the image border were removed using 5-pixel connectivity to prevent including partial pores in the characterization statistics. So, the contour of the object is clearer and fuller when applied to the open operator of the morphology method.

Types of road defects.

Block crack (fig.2): Interconnected portion of the pavement area but sometimes will occur only in non-traffic areas. Cracks that divide the pavement up into rectangular pieces. Blocks range in size from approximately 0.1 m^2 to 9 m^2. Larger blocks are generally classified as longitudinal and transverse cracking.

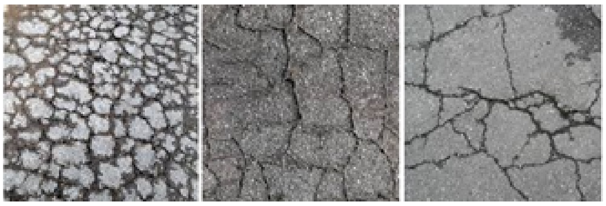

Figure 2: Examples of block crack defect.

Longitudinal Cracks (fig.3): Cracks parallel to the pavement's center line or lay down direction. Usually a type of fatigue cracking. Allows moisture infiltration, roughness, indicates a possible onset of fatigue cracking and structural failure.

Figure 3: Examples of longitudinal defect.

Potholes(fig.4): Generally, potholes are the end result of fatigue cracking. They generally have sharp edges and vertical sides near the top of the hole. As fatigue cracking becomes severe, the interconnected cracks create small chunks of pavement, which can be dislodged as vehicles drive over them. The remaining hole after the pavement chunk is dislodged is called a pothole

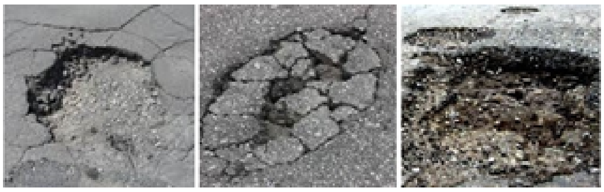

Figure 4: Example of potholes defect.

Features extraction. The images collected in this paper mainly include longitudinal cracks, block cracks, and potholes. To improve the identification accuracy of these defects, Hu-moments, the feature details regarding the connected domain, contours feature (chain code histogram) are added according to the extracted distress region and shape feature

- *Hu-moments.* The most notable are Hu-Moments which can be used to describe, characterize, and quantify the shape of an object in an image. Hu-Moments are normally extracted from the shape of an object in an image. By describing the shape of an object, there is able to extract a shape feature vector (i.e. a list of numbers) to represent the shape of the object.
- *Connected domain feature.* The number of connected domains of potholes and longitudinal cracks is 1. The number of connected domains of block cracks is more than 1. Therefore, the number of connected domains can be used to distinguish the types of pavement defects.
- *Chain code histogram.* The chain code histogram (CCH) is meant to group together objects that look similar to a human observer. It is not meant for exact detection and classification tasks. The CCH is calculated from the chain code presentation of a contour.

3.2 Application Deep Learning for road defects classification

In this section, several CNN classes and fully connected classes to build the training model (fig.5) were used. A model with two CNN layers followed by pooling and dropout then finally a dense layer (fully connected) was created. The first layer has 64 filters, the next layer has 128 filters and uses kernel size = 3 for all of them. The max-pooling application after each CNN class is used to avoid over-fitting. The purpose of pooling is simple, it reduces the number of hyper-parameters that need to calculate, thereby reducing calculation time, avoiding over-fitting. The most common type of pooling is max pooling, taking the largest value in a pooling window. Pooling works almost the same as convolution, it also has a sliding window called a pooling window, this window slides through each value of the input data matrix (usually the feature map in the convolutional layer), picking a price. The values from the values in the sliding window (with max-pooling will get the maximum value).

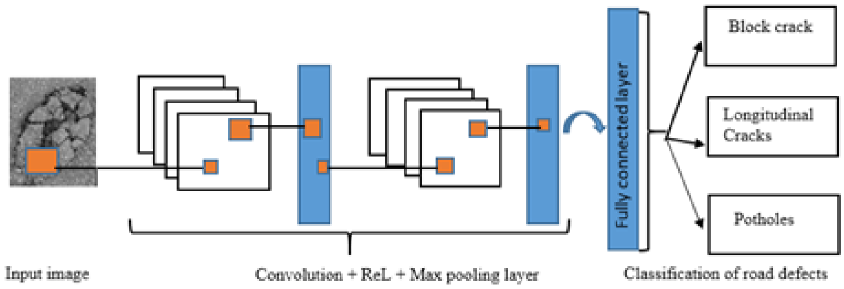

Figure 5: Schematic of convolutional neural network architecture.

CNN and Keras (*Keras backends,* 2008) are combined to apply to the classification problem of pavement defects. CNN consists of two components:

Hidden layer or feature extraction: in this section, the network will perform a series of convolution and pooling to detect features. In CNN, convolution is performed on the input value of the data and kernel/filter to create a feature map. The performance of convolution by sliding the kernel/filter according to the input data is proposed. At each position needs to multiply the matrix and sum the values to include it in the feature map.

Convolution is performed on 3-dimensional space. Because each image is represented in 3 dimensions: width, height, and depth. The convolution performed on input many different times. Use a different kernel/filter each time. As a result, different feature maps will be created. Finally, combine the entire feature map into the final result of the convolution. Normally, after each convolution layer, the result is going through a pooling layer. The purpose of this floor is to quickly reduce the number of dimensions. This helps reduce learning time and limits overfitting. A commonly used simple merge is max pooling, which takes the maximum value of a region to represent that region. The size of the area will be predefined to reduce the size of the feature map quickly but still retain the necessary information

Classification: in this section, a class with full links will work as a classifier of features extracted earlier. This layer gives the probability of an object in the figure. Several layers were used with sufficient connectivity to handle the result of the convolution. Because the input of the full link network is 1 dimension, the input before layering must be flattened. The last layer in the CNN network is a fully linked layer, this part works similar to the regular neural network. The final result will also be a vector with probability values for predictions like the regular neural network.

Algorithm 1 CNN-VGG16 with Keras for classification of road defects.

1. Load the image and pre-processing (convert to gray-scale, filter image, morphology).
2. Performs a final preprocessing step by converting the data to a 'float32' datatype NumPy array.
3. Initialize the training data augmentation object.
4. Load the VGG16 network using pre-trained ImageNet weights

```
baseModel = VGG16(weights='imagenet',include-top=False,
input-tensor=Input(shape=(224, 224, 3)))
headModel = baseModel.output
headModel = Flatten(name="flatten")(headModel)
headModel = Dense(512, activation="relu")(headModel)
headModel = Dropout(0.5)(headModel)
headModel = Dense(len(config.CLASSES), activation="softmax")(headModel)
```

5. Initialize *LearningRateFinde*. Start training and learning rate and exponentially increase it.
5.1: Review the generated plot.
5.2: Update *config.py* with *MIN_LR* and *MAX_LR*, respectively.
5.3: Train the network on our full dataset.
6. Train the network

```
H = model.fit_generator(aug.flow(trainX, trainY, batch_size=config.BATCH_SIZE),
validation_data=(valX, valY),
steps_per_epoch=trainX.shape[0] ,
epochs=config.NUM_EPOCHS,
callbacks=[clr],
verbose=1
```

7. Evaluate the network and show a classification report

```
predictions = model.predict(testX, batch_size=config.BATCH_SIZE)
predictions.argmax(axis=1), target_names=config.CLASSES))
```

4 Analysis results of experimental

4.1 Dataset

In this work, three different sets of data collected from the Center for Telecommunications and Multimedia, INESC TEC, Portugal (1200 images) (*Center for Telecommunications and Multimedia, INESC TEC*, 2015), Irkutsk city - Russian Federation (800 images) and Thai Nguyen city - Vietnam (550 images). They are collected by a camera (Canon D100 16 megapixel). Images are captured in conventional daylight condition, distance from the camera to the surface of the road is 1m-1.2m.) has been used for the experiments. They contain 512×512 images, classified in three classes: block cracks (BC), longitudinal cracks (LC), potholes (P). Imbalanced data follow the idea of cost-sensitive make more suitable for learning. Class weights are an essential tuning parameter to achieve the desired performance. The concept is introduced (table 1): True Positive - TP is classified correctly as positive, True Negative - TN is classified correctly as negative, False Positive - FP is classified wrongly as positive, False Negative - FN is classified wrongly as negative. For a Deep Learning algorithm, there is always a tradeoff between true positive rate and true negative rate and the same applies for recall and precision. *True negative rate* $= \frac{TN}{TN+FP}$. *True positive rate* $= \frac{TP}{TP+FN}$. *Precision* $= \frac{TP}{TP+FP}$

Table 1: A tradeoff between true positive rate and true negative rate precision of database

Type of defects	True positive rate	True negative rate	Precision
Block cracks	0.960	0.556	0.941
Longitudinal cracks	0.88	0.698	0.991
Potholes	0.974	0.233	0.956

4.2 Experimental design

All experiments were carried out using a Win10 PC with Intel Core i3 CPU @ 3.00 GHz and 2.00 GB RAM. The CNN model is trained with a learning rate of 0.01 using a softmax activation function in the pooling class. The Relu trigger function is used all the hidden layers of the convolutional layer, the most learning times of the network is 100 times. The basic network architecture consists of: Input layer - Layers [Convolution layer - Max-pooling class - Activation class] - Output layer. When training the network, parameter Batch size = 32 was chosen to select the sample size to be learned and adjust the parameters in the network layers to have the best accuracy. The maximum number of network training sessions is 100 times. The model with the best classification results on the training set will be selected and tested on test sets for evaluation. The VGG16 is tested for model training. The result of classification is 97.07 %, 97.62 %, 98.50 %). The confusion matrix is as following (Table 2, 3, 4).

Table 2: The results classification road defect of Thai Nguyen Viet Nam dataset with VGG16

True class		Assigned class		Accuracy
97.07 %	BC	LC	P	(%)
BC	173	7	0	96.11
LC	3	193	4	96.50
P	1	2	167	98.24

Table 3: The results classification road defect of Irkutsk - Russia dataset with VGG16

True class		Assigned class		Accuracy
97.62 %	BC	LC	P	(%)
BC	195	2	3	97.50
LC	3	262	5	97.03
P	5	1	325	98.18

The CNN model is installed in Python language using the Keras library. Random forest algorithms (Breiman, 2001) (The number of the tree to train model = 2000, The function to measure

Table 4: The results classification road defect of Portugal dataset with VGG16

True class		Assigned class		Accuracy
98.50 %	BC	LC	P	(%)
BC	445	2	3	98.89
LC	2	317	1	99.09
P	7	3	420	97.67

the quality of a split - Gini Impurity, Bootstrap samples = True.), Support Vector Machine (Cortes and Vapnik, 1995) (Penalty parameter to measure error term = 1.0, Kernel: basis functions, Shrinking heuristic = True) are installed to compare experimental results on all three data sets. Table 5 shows the results of classification base on 3 methods.

Table 5: The results of classification based on VGG-16, Random Forest, SVM for 3 data sets

Criteria	Data set	Random Forest	SVM	VGG-16
Accuracy (%)	DB1	91.77	88.52	98.50
	DB2	91.05	86.76	97.62
	DB3	90.89	86.35	97.07
Time of classification	DB1	1775	1202	1800
	DB2	1287	815	708
	DB3	1153	870	513
MSE	DB1	0.305	0.670	0.200
	DB2	0.310	0.566	0.311
	DB3	0.230	0.412	0.296

ROC curves also give us the ability to assess the performance of the classifier over its entire operating range. The most widely-used measure is the area under the curve (AUC). The AUC (fig.6) for a classifier of road defects based on VGG-16, Random Forest algorithm, and SVM. The result showed that the result of classification is good because curves have status is nearer 1 value than 0.5 values.

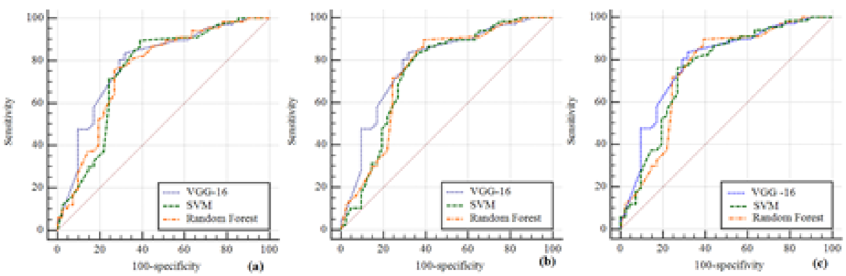

Figure 6: Results of analysis ROC for classification of road defect (a) INESC TEC, Portugal, (b) Irkutsk Russia, (c) Thai Nguyen Viet Nam.

The test accuracy is greater than training accuracy (fig.7). This means that the model of VGG-16 has generalized very well. This comes from the fact that the model has been trained on much data, so it is finding the normal test data easier to classify.

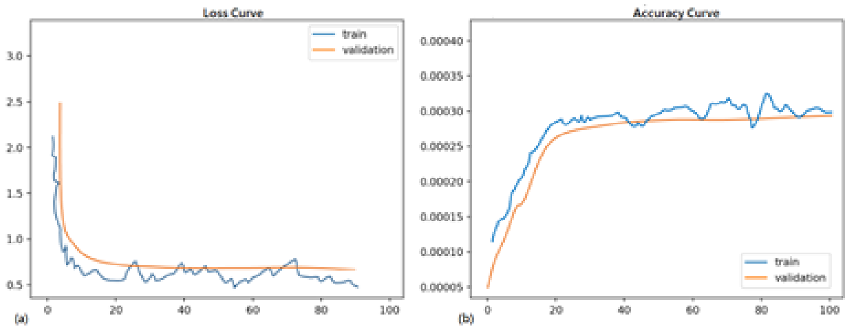

Figure 7: Traces of training and validation loss (a) and accuracy (b).

5 Conclusion

In this study, a method of classifying pavement defects based on CNN architecture - VGG16 is proposed using a combination of Convolutional layers in combination with ReLu and Max-pooling layers. Use Max-pooling before ReLu to minimize the level of computation instead of doing the opposite. The convolution max-pooling - ReL was used and spend 1-time calculating max-pooling, 1-time calculating ReLu. The reason for this was that significantly better results were obtained with pre-trained networks compared to the results of both accuracy and time of processing. Besides, the image preprocessing step that included finding a region of interest (ROI) and features extraction based on algorithms such as image filter, edge detection based on Canny algorithm, a mathematical morphology method to reduce the possible noise influence, which was proven to be critical for the success of the implementation of the defect road detection process. Test results on 3 different data sets with a large number of objects (2550 images) bring efficiency with high accuracy. This confirms that CNN not only works effectively for natural language processing but also can be applied effectively in areas such as computer vision, machine learning (a problem of object recognition in images, what high noise and light sensitivity).

6 Acknowledgements

The authors gratefully thank the referee for careful reading of the paper and valuable suggestions and comments. This research was performed at the Baikal School of BRICS and the Laboratory of Artificial Intelligence and Machine Learning, Irkutsk National Research Technical University (Irkutsk, Russia). Authors also are thankful to the Center for Telecommunications and Multimedia, INESC TEC, Portugal for providing the dataset.

References

Abduljabbar, R., Dia, H. and Liyanage, S. 2019. Applications of artificial intelligence in transport: An overview, *Sustainability* **11**: 189 – 213.

Ali, T. and Hamzeh, K.and Hassan, Y. 2019. Machine learning in transportation, *Journal of advanced transportation* **19**: 1–13.

Aubry, S. P. M., Hasinoff, S., Kautz, J. and Durand, F. 2014. Fast local laplacian filters: Theory and applications, *ACM Transactions on Graphics* **33**: 167 – 182.

Breiman, L. 2001. Random forests, *Machine learning* **1**: 5 – 32.

Budalakoti, S., Srivastava, A. and Ote, M. 2009. Anomaly detection and diagnosis algorithms for discrete symbol sequences with applications to airline safety, *IEEE Trans. Syst. Man Cybern. Part C Appl.* **5**: 101 – 113.

Cai, X., Zhang, N. and et al. 2007. Time series prediction with recurrent neural networks trained by a hybrid psoea algorithm, *Neurocomputing* **70**: 2342–2353.

Canny, J. 1986. A computational approach to edge detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **8**: 679 – 698.

Center for Telecommunications and Multimedia, INESC TEC 2015. [Online]. Available: https://www.inesctec.pt/en/centres/ctm?p=9605(current October 2015).

Cortes, C. and Vapnik, V. 1995. Support-vector networks, *Machine learning* **20**: 273 – 297.

Danielle, A., Rebecca, M. and et al. 2018. A comparative study of nine machine learning techniques used for the prediction of diseases, *International Journal of Artificial Intelligence* **16**: 25–40.

Dansh, W., Min, Z., L., Z. and et al. 2017. Modulation format recognition and osnr estimation using cnn-based deep learning, *IEEE Photonics Technology Letters* **28**: 1667 – 1670.

Dia, H. 2001. An object-oriented neural network approach to short-term trafc forecasting, *Eur. J. Oper. Res.* **131**: 253 – 261.

Dogan, E. and Akgngr, A. 2013. Forecasting highway casualties under the effect of railway development policy in turkey using articial neural networks, *Neural Comput. Appl.* **22**: 869 – 877.

Gopalakrishnan, K., Khaitan, S., Choudhary, A. and A., A. 2017. Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection, *Constr. Build. Mater* **157**: 322 – 330.

Hoang, N. and Nguyen, Q. 2018. A novel method for asphalt pavement crack classification based on image processing and machine learning, *Engineering with Computers* **35**: 487 – 498.

Keras backends 2008. [Online]. Available: http://keras.io (current October 2018).

Kiarash, Z. and Mehdi, T. 2018. Cost-sensitive neural network for optimizing medical expenditures of populations, *International Journal of Artificial Intelligence* **16**: 78–105.

Kim, A., Yang, Y., Lessmann, S., T., M. and V., J. J. E. 2020. Can deep learning predict risky retail investors. a case study in financial risk behavior forecasting, *European Journal of Operational Research* **283**: 217 – 234.

Klugl, F., Bazzan, A. and Ossowski, S. 2010. Agents in trafc and transportation, *Transp. Res. Part C Emerg. Technol.* **18**: 5970.

Lekha, R., Nair, S. D. and Siddhanth, D. S. 2018. Applying spark based machine learning model on streaming big data for health status prediction, *Computers and Electrical Engineering* **65**: 393 – 399.

Lv, Y., Duan, Y., Kang, W. and Li, Z. 2014. Trafc flow prediction with big data: A deep learning approach, *IEEE Trans. Intell. Transp. Syst.* **16**: 865 – 873.

Mobyen, U. A., Staffan, B. and et al. 2020. A machine learning approach to classify pedestrians event based on imu and gps, *International Journal of Artificial Intelligence* **17**: 154–167. **URL:** *http://www.es.mdh.se/publications/5255-*

Mohamed, A. A., Riyaz, A. A. H. and Fariza, H. N. 2020. Deep learning and big data technologies for iot security, *Computer Communications* **1511**: 495 – 517.

Nguyen, T. and Nguyen, T. L. 2019. Roc curve analysis for classification of road defects, *BRAIN Broad Research in Artificial Intelligence and Neuroscience* **10**: 65 – 73.

Nguyen, T., Nguyen, T. L. and Greglea, A. I. 2018. Machine learning algorithms application to road defects classification, *Intelligent Decision Technologies* **12**: 59 – 66.

Oliveira, H., Caeiro, J. and Correia, P. 2016. Accelerated unsupervised filtering for the smoothing of road pavement surface imagery, *I4 Proceedings of the 2014 22nd European Signal Processing Conference (EUSIPCO)*, Vol. 21, Lisbon, Portugal, pp. 2465 – 2469.

Schlotjes, M., Burrow, M. and Evdorides, H. 2015. Using support vector machines to predict the probability of pavement failure, *Proc. Inst. Civ. Eng. Transp.* **168**: 212 – 222.

Sinziana, M.and Michael, H. and Zack, Z. 2012. Online detection of freezing of gait with smartphones and machine learning techniques, *6th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth) and Workshops* **7**: 123–130.

Stefan, P., Radu-Emil, P. and et al. 2007. Iterative feedback and learning control. servo systems applications, *IFAC Proceedings* **40**: 16–27.

Wang, R., Fan, S. and Work, D. 2016. Efcient multiple model particle ltering for joint trafc state estimation and incident detection, *Transp. Res. Part C Emerg. Technol.* **71**: 521 – 537.

Wei, Y. L., Ya, H. H. and Tsai, F. 2012. Machine learning in financial crisis prediction: A survey, *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* **5**: 421 – 436.

Wu, G., Sun, X., Zhou, L. and Zhang, H. 2016. Research on morphological wavelet operator for crack detection of asphalt pavement, *In Proceedings of the 2016 IEEE International Conference on Information and Automation*, Vol. 25, Ningbo, China, pp. 1573 – 1577.

Yunchao, W., Wei, X., Min, L. and et al. 2016. A flexible cnn framework for multi-label image classification, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **11**: 1901 – 1907.

Zhang, D., Li, Q., Chen, Y. and Cao, M. 2017. An efficient and reliable coarse-to-fine approach for asphalt pavement crack detection, *Image Vis. Comput.* **57**: 130 – 146.