Fusion Journal of Engineering and Sciences

Research Paper

Automating Pavement Health Assessment: Leveraging Machine Learning for Efficient Pavement Crack Detection and Classification

Touqeer Ali Rind[1]*, Shiraz Ahmed[1] and Muhammad Faisal Javed[1] and Muhammad Faarid Shah[1]

1 Department of Civil Engineering (DCvE), Ghulam Ishaq Khan (GIK) Institute of Engineering Sciences and Technology, Khyber Pakhtunkhwa; shiraz.ahmed@giki.edu.pk, arbabfaisal@giki.edu.pk, gcv2560@giki.edu.pk

* Correspondence: touqeerali@muetkhp.edu.pk

Abstract

Pavement degradation, including cracks, are a major road maintenance issue and a hazard to the safety of infrastructure. Conventional manual checking systems are time consuming and are subject to human error making automated cracking systems very essential. This research discusses four machine learning (ML) classifiers, including k-Nearest Neighbors (kNN), Support Vector Machine (SVM), Neural Network (NN) and Logistic Regression (LR), as efficient to automate the process of pavement crack detection based on image classification. The models were trained and tested on a set of 360 images (180 cracked and 180 uncracked pavement). The findings revealed that SVM, NN and LR had 100 percent accuracy in the classification of cracked and uncracked pavement whereas the kNN had 97.2 percent accuracy. Although the performance decreased slightly, kNN was found to be very reliable thus it can be effectively used to carry out the task of detecting cracks. Measures of evaluation, such as Area Under Curve (AUC), Classification Accuracy (CA), F1 Score, Precision, Recall, and Matthews Correlation Coefficient (MCC) showed that all models performed excellent. The results indicate the promise of ML in the domain of pavement condition assessment automation, which will result in the improved and less expensive infrastructure management. Future research must aim at extending the models to other more complicated situations, such as multi-class classification and real-time monitoring.

Keywords: Pavement Crack Detection, Machine Learning, k-Nearest Neighbors (kNN), Support Vector Machine (SVM), Neural Network (NN), Logistic Regression (LR).

1. Introduction

The development of cracks in pavements is a serious problem of infrastructural maintenance, and it has an influence on road safety, transportation rates, and overhead expenses [1], [4], [7]. Pavement cracks are not only aesthetically displeasing but also cause more damages to the road surface structure that results in structural failures when left uncontrollable [4], [7]. Early detection of cracks also prevents the occurrence of massive damages and enables the cracks to be repaired in good time, and hence prolongs the life of road infrastructure [1], [4].

Conventional processes of crack detection, which in most cases are manual based and visual process, are time consuming, labor intensive and can be prone to human error [3], [7]. Such techniques are also constrained in terms of scalability especially when dealing with huge networks of roadways [3], [4]. To overcome these issues, the automated pavement inspection field has received a considerable amount of interest, and the latest progress in image analysis and machine learning (ML) offers new opportunities of more efficient, precise, and scalable solutions [2], [7], [24]. Supervised learning model techniques of machine learning have presented significant potentials in ensuring the automation of crack detection by classifying the images [22], [24]. Using large volumes of image data, it is possible to train ML models to identify and categorize cracking and uncracking pavement, which provides an accurate and repeatable substitute to human inspection [12]–[15], [22]. These models are able to process massive datasets within a very short duration of time, and they learn based on labeled images to detect patterns and features that represent pavement deterioration [24], [25]. In addition, machine learning allows flexibility to keep enhancing models by adding new data and makes them adapt to different types and conditions of pavement [19], [24].

This paper discusses how four machine learning models, including k-Nearest Neighbors (kNN), Support Vector Machine (SVM), Neural Network (NN), and Logistic Regression (LR), can be used to handle the task of automated crack detection [22], [24]. The training and validation were done using a dataset of 360 pavement images, 180 of which contained pavement images of cracks and 180 of which contained pavement images without cracks. The performance measures applied to evaluate the models included Accuracy, Precision, Recall, F1 Score, Area Under Curve (AUC), and Matthews Correlation Coefficient (MCC), which gives a wide perspective of the classification abilities of the models [22], [24]. This study is important because the obtained results apply to the study of automated pavement maintenance since they can inform the community about the efficiency of various machine learning models in crack recognition [4], [7], [24]. In this paper, it is intended to prove that machine learning and specifically, deep learning, and advanced ML algorithms are effective in classifying the pavement, which will help to automate the infrastructure inspection and maintenance process [9], [15], [24], [25]. The study aims at identifying the best and most reliable techniques of real-time monitoring of pavement and detecting pavement damages by comparing the performances of different models that are important in ensuring that road infrastructure is sustainable [1], [4], [7].

2. Dataset

The dataset that has been deemed to be used in this study consists of a total number of 360 images, the images that have been chosen to be specific to one of the two types of pavement conditions, which is cracked and uncracked pavement. Out of the 360 images, 180 have cracks on the pavement, and the rest 180 have no cracks on the pavement. This equal distribution is a balanced dataset which makes both sets of data equally represented and hence the machine learning models may be trained and evaluated without any bias [22], [24]. The images were obtained to have different types of cracks that are typical of pavement surfaces such as longitudinal cracks, transverse cracks and block cracks so that there was diversity in the data set [7], [24]. The photos were also captured in different light conditions in order to recreate real world situations which are essential in determining the strength of the models once being applied in diverse settings [19], [24].

In order to optimize the training process and prove the performance of the model efficiently, the dataset was divided into two separate parts: the training (80 percent of the images or 288 images) and the validation (20 percent of the images or 72 images) ones. The machine learning models were developed and fine-tuned using the training set, which they were given an opportunity to learn the distinguishing characteristics between cracked and uncracked pavement [22], [24]. The validation set comprised of the images that were not shown in the training period and evaluated the capability of the models to generalize and correctly act with new unknown data [22], [24]. This partitioning plan, 80 percent training and 20 percent validation allows the models to not be overfitting the training data but be able to make accurate predictions on unseen data which is important in real world application in pavement crack detection [19], [24].

3. Materials and Methodology

The machine learning models used to automate the pavement cracks identification process in this paper are k-Nearest Neighbors (kNN), Support Vector Machine (SVM), Neural Network (NN), and Logistic Regression (LR). The selection of each model was made according to its capability to deal with classification problems and give reflections of the most effective methodologies in the crack detection of pavement [22], [24]. The models were trained with the 80% training and tested with the 20% validation and different performance metrics were used to analyze the performance effectiveness of the models [22], [24].

3.1 k-Nearest Neighbors (kNN)

The k-Nearest Neighbors (kNN) is a simple and quite a powerful classification algorithm. In the kNN, the k-nearest neighbors of an image in the feature space are compared with the majority image class and the image is classified by that image value [22]. The kNN algorithm predicts by finding the similarity between the image features and those of the training images and gives the prediction of the class using the closest similarity [22]. This non-parametric model has been known to be effective in small to medium sized datasets and is easily implemented [22].

3.2 Support Vector Machine (SVM)

The Support Vector Machine (SVM) is a supervised learning algorithm that is usually applied in binary classification tasks. SVM attempts to identify the linear or the best hyperplane that will divide the classes on the basis of the optimal margin [22]. SVM has been applied to detect cracks and it is effective in this case because it maps images of the input to a space of higher dimension feature space and in this way, it will be able to better distinguish between images of cracked and uncracked pavement [22], [24]. A fundamental component of this study is the SVM model which was optimized by choosing an appropriate kernel function (Radial Basis Function - RBF) which is most applicable in non-linear distributions of data [22].

3.3 Neural Network (NN)

A Neural Network (NN) model is a brain-like computational model that is a series of interconnected computational nodes/neurons, arranged in layers. A feedforward neural network was employed to detect cracks and this method is applied to images via input, hidden, and output layers [24]. Backpropagation was used to train the network with help of adjusting weights to reduce the prediction error [24]. Neural networks are very efficient in learning complex trends in data, and they are extensively applied in image classification tasks because they can learn non-linear associations [12]–[15], [20], [21], [24].

3.4 Logistic Regression (LR)

A statistical model employed in binary classification is the Logistic Regression which approximates the probability of a given data point fitting into a given class. The use of Logistic Regression in this research study was to fit the relationship between the input variables (image characteristics) and the likelihood of an image depicting broken pavement [22]. Although it is simple, the Logistic regression can be useful in the classification task when the decision boundary is in the form of approximation [22].

3.5 Training and validation Process

Each model was trained using the training dataset, which comprised of 288 images (80 percent of the total dataset), and each model was trained to classify the images to be either “cracked” or “uncracked” using the extracted features [22], [24]. The rest of the images (20% of the total dataset) that had been stored aside during the training process were then used to validate the models. This validation procedure enabled testing how each model was able to perform on unknown data and supports generalisation assessment in pavement crack detection studies [19], [22], [24].

3.6 Evaluation Metrics (with references)

In order to test the effectiveness of the machine learning models, one used the following measures (commonly reported in pavement crack detection and classification literature) as given in table 1[19], [22], [24]:

Table 1: Description and Interpretation of Evaluation Metrics Used for Crack Detection Model Performance [19], [22], [24]

Metric	What it measures	Range / Best value	How to interpret (in crack detection)
Area Under Curve (AUC)	The model's ability to **separate** the two classes (cracked vs uncracked) across different thresholds.	**0 to 1** (best = **1.0**)	Higher AUC means the model is better at distinguishing cracked from uncracked images even when the decision threshold changes.
Classification Accuracy (CA)	The proportion of **all correctly classified** images out of total images.	**0 to 1** or **0% to 100%** (best = **100%**)	Higher accuracy means more correct predictions overall. Can be misleading if classes are imbalanced, so it should be reported with other metrics.
F1 Score	The **harmonic mean** of Precision and Recall (balances both).	**0 to 1** (best = **1.0**)	High F1 indicates the model is good at detecting cracks **without** producing too many false alarms, and it also catches most true cracks. Useful especially when data is imbalanced.
Precision	Of all images predicted as **cracked**, how many are **actually cracked**.	**0 to 1** (best = **1.0**)	High precision means **few false positives** (the model doesn't wrongly label uncracked pavement as cracked).
Recall	Of all **truly cracked** images, how many the model correctly predicts as cracked.	**0 to 1** (best = **1.0**)	High recall means **few false negatives** (the model misses fewer real cracks).
Matthews Correlation Coefficient (MCC)	A balanced measure using **TP, TN, FP, FN**; strong even with imbalanced data.	**-1 to +1** (best = **+1**)	**+1** = perfect prediction, **0** = random/guessing, **-1** = completely wrong. Higher MCC means overall classification quality is strong across both classes.

4. Results

The results of each machine learning model, which include k-Nearest Neighbors (kNN), Support Vector Machine (SVM), Neural Network (NN), and Logistic Regression (LR), were measured according to some major metrics, namely, Area Under Curve (AUC), Classification Accuracy (CA), F1 Score, Precision, Recall, and Matthews Correlation Coefficient (MCC). These measures were estimated with respect to each model and the validation set comprising of 72 images (20 percent of the total dataset). Figure-1 shows the workflow of machine learning model development.

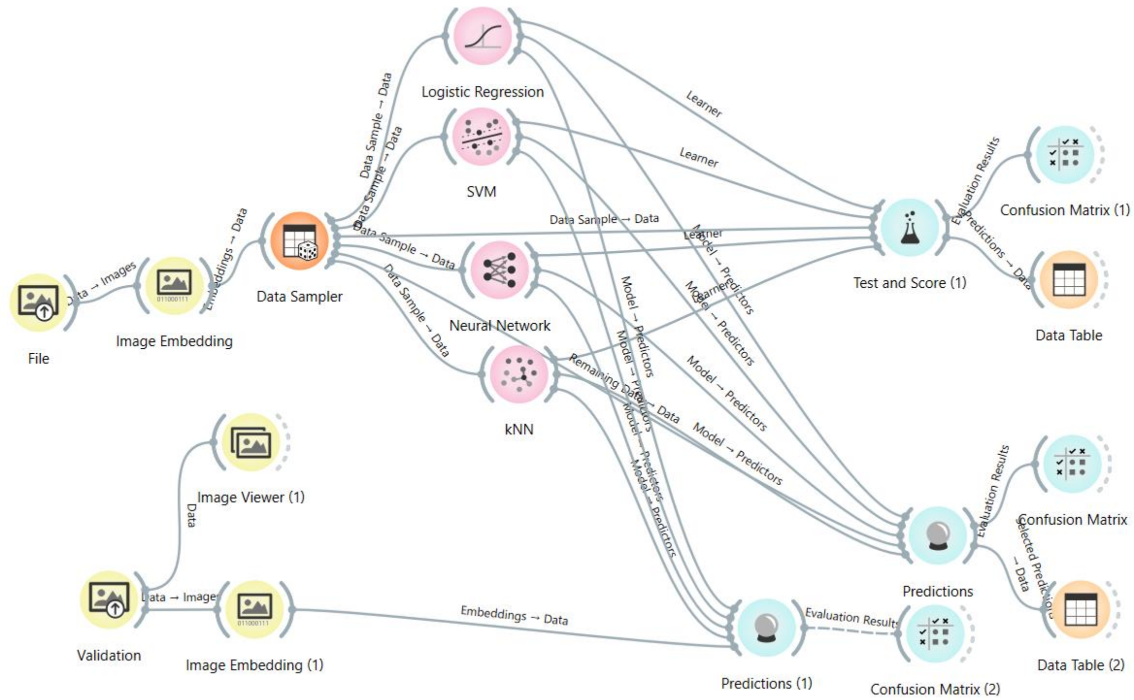
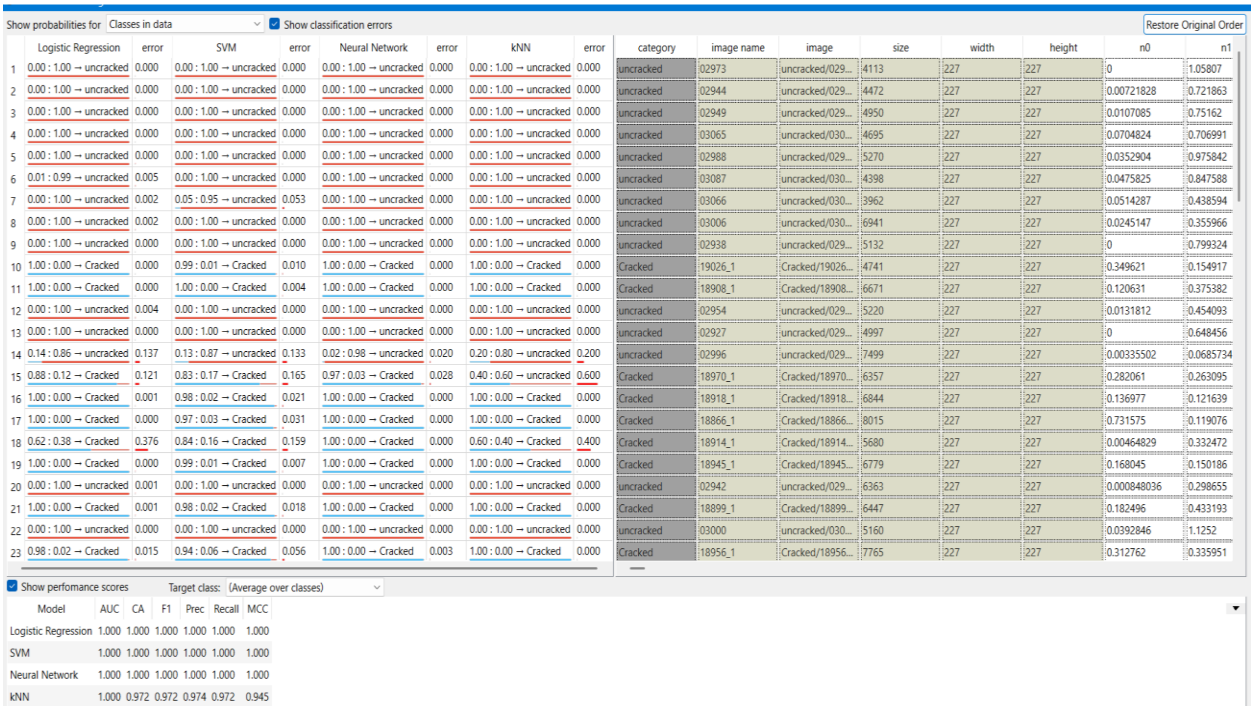


Figure 1: Workflow of the machine learning model development.

4.1 Machine Learning Model Performance Metrics.

Table 1 summarizes the performance of each model and gives the evaluation metrics of each model in terms of the validation set. These measures are AUC, CA, F1 Score, Precision, Recall, and MCC and figure gives the validation output of the machine learning model development.



Show probabilities for Classes in data | Show classification errors | Restore Original Order

	Logistic Regression	error	SVM	error	Neural Network	error	kNN	error	category	image name	image	size	width	height	n0	n1
1	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	uncracked	02973	uncracked/029...	4113	227	227	0	1.05807
2	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	uncracked	02944	uncracked/029...	4472	227	227	0.00721828	0.721863
3	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	uncracked	02949	uncracked/029...	4950	227	227	0.0107085	0.75162
4	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	uncracked	03065	uncracked/030...	4695	227	227	0.0704824	0.706991
5	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	uncracked	02988	uncracked/029...	5270	227	227	0.0352904	0.975842
6	0.01 : 0.99 → uncracked	0.005	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	uncracked	03087	uncracked/030...	4398	227	227	0.0475825	0.847588
7	0.00 : 1.00 → uncracked	0.002	0.05 : 0.95 → uncracked	0.053	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	uncracked	03066	uncracked/030...	3962	227	227	0.0514287	0.438594
8	0.00 : 1.00 → uncracked	0.002	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	uncracked	03006	uncracked/030...	6941	227	227	0.0245147	0.355966
9	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	uncracked	02938	uncracked/029...	5132	227	227	0	0.799324
10	1.00 : 0.00 → Cracked	0.000	0.99 : 0.01 → Cracked	0.010	1.00 : 0.00 → Cracked	0.000	1.00 : 0.00 → Cracked	0.000	Cracked	19026_1	Cracked/19026...	4741	227	227	0.349621	0.154917
11	1.00 : 0.00 → Cracked	0.000	1.00 : 0.00 → Cracked	0.004	1.00 : 0.00 → Cracked	0.000	1.00 : 0.00 → Cracked	0.000	Cracked	18908_1	Cracked/18908...	6671	227	227	0.120631	0.375382
12	0.00 : 1.00 → uncracked	0.004	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	uncracked	02954	uncracked/029...	5220	227	227	0.0131812	0.454093
13	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	uncracked	02927	uncracked/029...	4997	227	227	0	0.648456
14	0.14 : 0.86 → uncracked	0.137	0.13 : 0.87 → uncracked	0.133	0.02 : 0.98 → uncracked	0.020	0.20 : 0.80 → uncracked	0.200	uncracked	02996	uncracked/029...	7499	227	227	0.00335502	0.0685734
15	0.88 : 0.12 → Cracked	0.121	0.83 : 0.17 → Cracked	0.165	0.97 : 0.03 → Cracked	0.028	0.40 : 0.60 → uncracked	0.600	Cracked	18970_1	Cracked/18970...	6357	227	227	0.282061	0.263095
16	1.00 : 0.00 → Cracked	0.001	0.98 : 0.02 → Cracked	0.021	1.00 : 0.00 → Cracked	0.000	1.00 : 0.00 → Cracked	0.000	Cracked	18918_1	Cracked/18918...	6844	227	227	0.136977	0.121639
17	1.00 : 0.00 → Cracked	0.000	0.97 : 0.03 → Cracked	0.031	1.00 : 0.00 → Cracked	0.000	1.00 : 0.00 → Cracked	0.000	Cracked	18866_1	Cracked/18866...	8015	227	227	0.731575	0.119076
18	0.62 : 0.38 → Cracked	0.376	0.84 : 0.16 → Cracked	0.159	1.00 : 0.00 → Cracked	0.000	0.60 : 0.40 → Cracked	0.400	Cracked	18914_1	Cracked/18914...	5680	227	227	0.00464829	0.332472
19	1.00 : 0.00 → Cracked	0.000	0.99 : 0.01 → Cracked	0.007	1.00 : 0.00 → Cracked	0.000	1.00 : 0.00 → Cracked	0.000	Cracked	18945_1	Cracked/18945...	6779	227	227	0.168045	0.150186
20	0.00 : 1.00 → uncracked	0.001	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	uncracked	02942	uncracked/029...	6363	227	227	0.000848036	0.298655
21	1.00 : 0.00 → Cracked	0.001	0.98 : 0.02 → Cracked	0.018	1.00 : 0.00 → Cracked	0.000	1.00 : 0.00 → Cracked	0.000	Cracked	18899_1	Cracked/18899...	6447	227	227	0.182496	0.433193
22	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	0.00 : 1.00 → uncracked	0.000	uncracked	03000	uncracked/030...	5160	227	227	0.0392846	1.1252
23	0.98 : 0.02 → Cracked	0.015	0.94 : 0.06 → Cracked	0.056	1.00 : 0.00 → Cracked	0.003	1.00 : 0.00 → Cracked	0.000	Cracked	18956_1	Cracked/18956...	7765	227	227	0.312762	0.335951

Show perfomance scores | Target class: (Average over classes)

Model	AUC	CA	F1	Prec	Recall	MCC
Logistic Regression	1.000	1.000	1.000	1.000	1.000	1.000
SVM	1.000	1.000	1.000	1.000	1.000	1.000
Neural Network	1.000	1.000	1.000	1.000	1.000	1.000
kNN	1.000	0.972	0.972	0.974	0.972	0.945

Figure 2: Machine learning model prediction results.

Table 2: Summary of each models performance using evaluation metrics

Model	AUC	CA	F1	Precision	Recall	MCC
Logistic Regression	1.000	1.000	1.000	1.000	1.000	1.000
SVM	1.000	1.000	1.000	1.000	1.000	1.000
Neural Network	1.000	1.000	1.000	1.000	1.000	1.000
k-Nearest Neighbors (kNN)	1.000	0.990	0.990	0.990	0.990	0.979

4.2. Confusion Matrices

Confusion matrices are provided in Table 3 for each of the four models. The confusion matrix is a vital tool for evaluating the performance of classification models, as it shows the true positives (correctly predicted cracked pavement), false positives (incorrectly predicted cracked pavement), true negatives (correctly predicted uncracked pavement), and false negatives (incorrectly predicted uncracked pavement). Each model was evaluated using these metrics to calculate the errors and overall accuracy, as shown in the confusion matrices.

Table 3: Confusion matrices for each of the four models

Model	Actual/Predicted	Cracked	Uncracked	Total	Errors	Accuracy
Logistic Regression	**Cracked**	148	0	148	0	100%
	Uncracked	0	140	140		

Model	Actual/Predicted	Cracked	Uncracked	Total	Errors	Accuracy
SVM	**Cracked**	148	0	148	0	100%
	Uncracked	0	140	140		
Neural Network	**Cracked**	148	0	148	0	100%
	Uncracked	0	140	140		
k-Nearest Neighbors (kNN)	**Cracked**	146	2	148	3	97.2%
	Uncracked	1	139	140		

4.3 Analysis of Results

Logistic Regression, SVM, and Neural Network: The accuracy of all three of these models was 100 percent on the validation dataset. All the images of cracked and uncracked pavement were properly categorized and there were no false positives or false negatives. Such findings suggest that these models can be used effectively to detect cracked pavement provided that they get adequate training data and are optimally packed.

k-Nearest Neighbors (kNN): kNN also achieved very high results, but not as good as the previous one (97.2%). The model wrongly categorized very few images (3 errors in cracked images and 1 error in uncracked images). Nevertheless, even despite these small flaws, the kNN model remained very reliable and it can be regarded as a possible choice when it comes to crack detection in particular and when the values of simplicity in computation are of primary value.

All in all, the findings indicate that machine learning models are effective in crack detection in pavement images automatic detection. SVM, Neural Network and Logistic Regression performed perfectly with kNN having a slight reduction in accuracy.

Overall, it can be said that the models tested in this research, SVM, Neural Network, and Logistic Regression, performed at 100 percent at the validation dataset with the accuracy of 100 percent in classifying between cracked and uncracked pavement. These models are most useful in this activity, and SVM and Neural Networks are most useful in reflecting complex, non linear relationships, whereas Logistic Regression worked well given the evident linear separation in the data. Conversely, kNN is a little less accurate with an accuracy of 97.2 but did not fail to perform optimally and showcased the power of less complicated models in image classification exercises. The slight decrease in accuracy using kNN is explained by the following factors; k selection, sensitivity to data distribution and the use of local decision making by the algorithm. Although these are minor problems, kNN is a powerful and viable tool to use in detection of cracks, especially in the case where simplicity and efficiency are some of the major factors to be considered.

5. Conclusions and Recommendations

This paper has shown that machine learning (ML) models can be successfully used to automatically determine pavement conditions, namely, detecting cracks. They used four models k-Nearest Neighbors (kNN), Support Vector Machine (SVM), Neural Network(NN), and Logistic Regression (LR) on 360 images (180 images of cracked pavement and 180 images of uncracked pavement). The findings demonstrated that SVM, Neural Networks and Logistic Regression obtained perfect classification accuracy (100) with the validation set whereas kNN obtained low classification accuracy (97.2) with some misclassified images.

The major results of the research are as follows:

High Accuracy: With all models, except kNN, 100% classification accuracy with the models, it can be concluded that the ML algorithms are capable of distinguishing between a cracked and uncracked pavement.

Applicability of Models: SVM, Neural Networks and Logistic Regression have shown tremendous performance in terms of their versatility in handling image classification problems.

Strength of kNN: kNN was not harmed by this slight decrease in accuracy, but its performance was quite good that underscores its usefulness in simpler applications where computational efficiency is relevant.

It is important to note that the evaluation metrics, such as Area Under Curve (AUC), Classification Accuracy (CA), F1 Score, Precision, Recall and Matthews Correlation Coefficient (MCC) are all high, which supports the possibility of using ML in the pavement crack detection method.

These findings have implication to the infrastructure maintenance field. Automatic classification of pavement conditions can significantly save time and resources in order to conduct the manual inspection. It is possible to use machine learning models to detect the cracks in the earliest possible time to perform the maintenance in time and avoid the further worsening of the pavement, which could result in safer and more cost-efficient infrastructure maintenance.

Communication Future Improvements and Applications:

Although the models used in this study were effective, the following areas can be improved and further used in the future:

Enhancement of Model Robustness: SVM, NN and LR models performed perfectly; however, in the real world, data is usually noisy and pavement can highly change in terms of lighting, weather, and surface types. In subsequent studies, it might be important to enlarge the sample that consists of varied images, various types of pavement, weather, and light conditions. This may increase the strength and the external validity of the models.

Fusion with Real-Time Systems: To make this strategy more relevant to the real-world pavement maintenance, these models may be incorporated into the automated inspection systems, including drones or mobile vehicles which have cameras on board. This would enable the real-time recognition and constant surveillance of cracks on the pavements without involving human intervention.

To Multi-Class Classification: In this paper, the analysis was limited to cracked and uncracked pavements. This classification could be extended in the future to determine the severity of cracks (e.g. small, medium, large) or describe the various types of pavement damage (e.g. potholes, fissures, surface wear) to give a more granular view of pavement health.

Enhancing kNN Performance: Although the kNN was very effective; it was slightly less accurate than other models. Future research might include more fine-tuning of the kNN algorithm possibly by testing out alternative distance measures or integrating feature selection methods to enhance its classification capabilities.

Applications in Maintenance Programs: This machine learning model may be deployed on maintenance programs in pavement after the optimization process. Automated systems could be used by the cities and municipalities to scan the roads regularly and identify cracks and other damages at an early stage. This would help to use proactive maintenance strategies to enhance the safety of the roads and minimise the long run repair costs.

To sum up, machine learning can be a solution to the automation of the process of classifying pavement conditions, and the models analyzed in this paper could become a good basis in further development of automated pavement infrastructure analysis. As these models are continued to be improved and incorporated into actual systems, they can end up transforming the way pavement maintenance is carried out and eventually help in the development of safer and sustainable road networks.

6. Patents

Author Contributions: The conceptualization, development of the methodology, data collection, formal analysis, and the initial drafting were done by Touqeer Ali Rind. Shiraz Ahmed helped in model testing, and data validation as well as support on the manuscript. Muhammad Faisal Javed offered the supervision, conceptual and methodological advice, validation of the results, and critical review of the manuscript.

Funding: This research received no external funding.

Acknowledgments: The authors gratefully acknowledge Ghulam Ishaq Khan Institute of Engineering Science and Technology (GIKI EST), for providing support and facilities in conducting this research.

Conflicts of Interest: The authors declare that there is no conflict of interest regarding this study and affirm that the work is original, without any form of plagiarism. All sources of information have been properly cited and acknowledged.

References

[1] K. Kobayashi and K. Kaito, “Big data-based deterioration prediction models and infrastructure management: towards assetmetrics,” *Structure and Infrastructure Engineering*, vol. 13, no. 1, pp. 84–93, 2017.

[2] R. P. Loce, R. Bala, M. Trivedi, and J. Wiley, *Computer vision and imaging in intelligent transportation systems*. Wiley Online Library, 2017.

[3] T. Strain, R. Eddie Wilson, and R. Littleworth, “Computer vision for rapid updating of the highway asset inventory,” *Transp Res Rec*, vol. 2674, no. 9, pp. 245–255, 2020.

[4] R. Haas and W. R. Hudson, *Pavement asset management*. John Wiley & Sons, 2015.

[5] V. Balali, A. Ashouri Rad, and M. Golparvar-Fard, “Detection, classification, and mapping of US traffic signs using google street view images for roadway inventory management,” *Visualisation in Engineering*, vol. 3, pp. 1–18, 2015.

[6] H. Preston, K. C. Atkins, M. Lebens, and M. Jensen, “Traffic sign life expectancy,” 2014.

[7] S. C. Radopoulou and I. Brilakis, “Automated detection of multiple pavement defects,” *Journal of Computing in Civil Engineering*, vol. 31, no. 2, p. 04016057, 2017.

[8] N. Sairam, S. Nagarajan, and S. Ornitz, “Development of mobile mapping system for 3D road asset inventory,” *Sensors*, vol. 16, no. 3, p. 367, 2016.

[9] X. Zheng, S. Qian, S. Wei, S. Zhou, and Y. Hou, “The Combination of Transformer and You Only Look Once for Automatic Concrete Pavement Crack Detection,” *Applied Sciences (Switzerland)*, vol. 13, no. 16, Aug. 2023, doi: 10.3390/app13169211.

[10] H. I. Ashqar, M. Elhenawy, M. Masoud, A. Rakotonirainy, and H. A. Rakha, “Vulnerable road user detection using smartphone sensors and recurrence quantification analysis,” in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, IEEE, 2019, pp. 1054–1059.

[11] M. Elhenawy, H. I. Ashqar, M. Masoud, M. H. Almannaa, A. Rakotonirainy, and H. A. Rakha, “Deep transfer learning for vulnerable road users detection using smartphone sensors data,” *Remote Sens (Basel)*, vol. 12, no. 21, p. 3508, 2020.

[12] N. T. H. Nguyen, T. H. Le, S. Perry, and T. T. Nguyen, “Pavement crack detection using convolutional neural network,” in *Proceedings of the 9th International Symposium on Information and Communication Technology*, 2018, pp. 251–256.

[13] L. Ali, N. K. Valappil, D. N. A. Kareem, M. J. John, and H. Al Jassmi, “Pavement crack detection and localisation using convolutional neural networks (CNNs),” in *2019 International conference on digitisation (ICD)*, IEEE, 2019, pp. 217–221.

[14] C. Zhang, E. Nateghinia, L. F. Miranda-Moreno, and L. Sun, “Pavement distress detection using convolutional neural network (CNN): A case study in Montreal, Canada,” *International Journal of Transportation Science and Technology*, vol. 11, no. 2, pp. 298–309, 2022.

[15] J. Liu et al., “Automated pavement crack detection and segmentation based on two-step convolutional neural network,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 35, no. 11, pp. 1291–1305, 2020.

[16] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” *Adv Neural Inf Process Syst*, vol. 28, 2015.

[17] J. Greenhalgh and M. Mirmehdi, “Real-time detection and recognition of road traffic signs,” *IEEE transactions on intelligent transportation systems*, vol. 13, no. 4, pp. 1498–1506, 2012.

[18] N. Kryvinska, A. Poniszewska-Maranda, and M. Gregus, “An approach towards service system building for road traffic signs detection and recognition,” *Procedia Comput Sci*, vol. 141, pp. 64–71, 2018.

[19] Y. Du, N. Pan, Z. Xu, F. Deng, Y. Shen, and H. Kang, “Pavement distress detection and classification based on YOLO network,” *International Journal of Pavement Engineering*, vol. 22, no. 13, pp. 1659–1672, 2021, doi: 10.1080/10298436.2020.1714047.

[20] L. Pauly, H. Peel, S. Luo, D. C. Hogg, and R. Fuentes, “Deeper Networks for Pavement Crack Detection,” 2017, doi: 10.22260/isarc2017/0066.

[21] W. Wang and C. Su, "Convolutional Neural Network-Based Pavement Crack Segmentation Using Pyramid Attention Network," *Ieee Access*, 2020, doi: 10.1109/access.2020.3037667.

[22] S. Mokhtari, L. Wu, and H. Yun, "Comparison of Supervised Classification Techniques for Vision-Based Pavement Crack Detection," *Transportation Research Record Journal of the Transportation Research Board*, 2016, doi: 10.3141/2595-13.

[23] Z. Qingzhe and Z. Qin, "Application of Machine Vision Technology in Road Detection," *Stavební Obzor - Civil Engineering Journal*, 2018, doi: 10.14311/cej.2018.04.0041.

[24] K. Gopalakrishnan, "Deep Learning in Data-Driven Pavement Image Analysis and Automated Distress Detection: A Review," *Data (Basel)*, 2018, doi: 10.3390/data3030028.

[25] F. Yang, L. Zhang, S. Yu, D. Prokhorov, X. Mei, and H. Ling, "Feature Pyramid and Hierarchical Boosting Network for Pavement Crack Detection," *Ieee Transactions on Intelligent Transportation Systems*, 2020, doi: 10.1109/tits.2019.2910595.

[26] Q. Dong, S. Wang, X. Chen, R. Li, and X. Gu, "Pavement Crack Detection Based on Point Cloud Data and Data Fusion," *Philosophical Transactions of the Royal Society a Mathematical Physical and Engineering Sciences*, 2023, doi: 10.1098/rsta.2022.0165.

[27] L. Wu, D. Zhugeng, and C. Liang, "Research on Asphalt Pavement Disease Detection Based on Improved YOLOv5s," *J Sens*, 2023, doi: 10.1155/2023/2069044