Asphalt Pavement Distress Detection by Transfer Learning with Multi-head Attention Technique

Original Scientific Paper

Ahmed Bahaaulddin A. Alwahhab *
Middle Technical University, Technical College of Management, Information Technology management Department
Baghdad, Iraq
ahmedbahaaulddin@mtu.edu.iq

Vian Sabeeh
Middle Technical University, Technical College of Management, Information Technology management Department
Baghdad, Iraq
viantalal@mtu.edu.iq

Ali Abdulmunim Ibrahim Al-kharaz
Middle Technical University, Technical College of Management, Information Technology management Department
Baghdad, Iraq
ali.al-kharaz@mtu.edu.iq

*Corresponding author

Abstract *– Roads and highways represent a crucial lifeline between communities in all countries. They have to be healthy enough for safe and effective transportation. The traditional ways of inspecting roads by human inspectors consume time, and the inspection results may be subjective. For this reason, researchers are motivated to automate pavement distress detection to help the road monitoring and maintenance process. Additionally, many researchers have tried to present models to detect distress on road infrastructure. However, these models face accuracy challenges and overfitting because of the nature and complications of distress images. This paper proposes a model that combines pre-trained VGG16 with a multi-head attention layer. The proposed paradigm began with smoothing as a pre-processing step to eliminate the granular effect of the asphalt gravel and make asphalt damage more distinct. Then, data augmentation was conducted to improve model generalization by adding various distress scenes to the dataset in geometric, color, and intensity cases. This work also contributes to the broader body of research by collecting a local dataset that contains three types of asphalt distress (cracks, potholes, and ruts). The proposed model was tested using three benchmarked datasets in addition to the locally collected one, and it showed efficiency in detecting asphalt distress using offline and real-time images. The model achieved an accuracy 1.00 in the Pavmentscapes dataset, outperforming the UNET model, and a fully connected network was trialed with the same dataset. With the Deep Crack dataset, our model scored an accuracy of 1.00.*

In contrast, ResNet achieved an accuracy of 0.72 on the same dataset. The NHA12D dataset was also used to test the proposed model and achieved an accuracy of 1.00, but the VGG16 without an attention layer used on that dataset scored only 0.64. All previous obvious tests prove that the proposed VGG16 and multi-head attention paradigm outperform the earlier models. Additionally, the proposed model has undergone a real-time test on local roads. The future directions are to try to make the self-attention mechanism more explainable and implement an attention layer for multi-scales.

Keywords*:* *Asphalt distress detection, Computer vision, Transfer learning, Multi-head attention*

Received: July 25, 2024; Received in revised form: October 26, 2024; Accepted: October 27, 2024

1. INTRODUCTION

As the asphalt pavement is the major part of the transportation infrastructure that leads to possible transportation operations, it is substantial in the long-term investment maintenance to ensure safety and prolonged useful pavement life. However, the empirically controlled system for monitoring schedules can no longer meet the demands in many global areas, such as long waiting times, unstable inspections, and low adequate verification. The asphalt pavement, which has a poor ability to resist huge impacts from climate and traffic loadings, will exhibit various distresses, such as the effects of friction conditions, climatic conditions, and traffic loading classifications [1]. Therefore, many developed countries aim to maintain roads and highways as strategic target to boost their economies and reduce poverty [2]. For example, in 2018, the Philippines set aside 11 billion dollars to maintain their national roads and bridges. In 2019, the US spent about 29 billion dollars on infrastructure, with highway and roadway infrastructure accounting for nearly 50% of federal transportation spending [3]. In comparison, China spent about 702.6 billion Yuan in 2023 on high-speed roads, a 12% increase from the year before [4].

Many factors, such as road aging, traffic loads, construction materials, lack of maintenance on time, and weather conditions, significantly impact pavement damage. As a result, Pavement distress rates can instrumentally amount to the risk of losing significant worth of pavement around the world because of restated upkeep rather than rebuilding. This information helped to strengthen the concerning asset management and to promote the reorientation of the interests to fewer resources while maintaining infrastructure by adopting life cycle production. Detecting pavement distress in good time plays a crucial role in eliminating the degradation of pavement surfaces [2]. Conventional detection primarily depends on manual techniques and is plagued by solid subjectivity, expensive implementation costs, and is time-consuming and unsuitable for quick detection. Pavement distress detection has been undertaken utilizing different image processing technologies (DIP) for nearly two decades. Almost all the aforementioned methodologies lay down some limitations without providing ways to improve them. Existing methodologies lack the capability to accurately model the entire spectrum of pavement distresses for asphalt suitability in either environmental factors or management. The DIP methodologies encompass processes such as edge detection [5], threshold segmentation [3], and morphological processing [4]. Advancements in DIP for pothole detection have been significant, yet the system still faces challenges in achieving impeccable accuracy and reliability in automatic pothole detection. Simultaneously, machine learning (ML) techniques utilized training classifiers – Naive Bayesian Classifiers (NBC), Support Vector Machines (SVM), and Artificial Neural Networks (ANN) – to identify diverse forms of pavement distress, including damages and cracks. The process involves these specific classifiers to learn how to recognize particular pavement defects by focusing on their distinctive features within images. This necessitates prior knowledge coupled with engineering expertise. However, the complexity inherent in feature extraction may occasionally impact detection accuracy, which could demand customization, such as refining an algorithm for a particular detection scenario [6]. Recent years have highlighted the significant efficiency of deep learning models in solving various computer vision problems, including object detection, image classification, and segmentation [7]. Image segmentation and pothole/crack detection tasks often employ Convolutional Neural Networks (CNNs) due to their ability to extract crucial features from asphalt images, such as texture, edges, and corners [8]. U-NET is the most common type of CNN used for pothole segmentation. It depends on the encoding-decoding feature, which uses a bounding connection inside neural architecture to save most of the information after the down-sampling process. The U-NET mechanism aided in preserving the spatial information from images, yielding an accurate segmentation process. U-NET works more accurately in pothole segmentation and has begun to be used widely by researchers in this field [9]. After the success of CNN, a new approach was developed called transfer learning (TL), which uses one performed task to implement another job. TL is a set of models already trained with a vast dataset related to the problem under investigation. These models allow researchers to build accurate models with much less training time and they need to be fine-tuned for the dataset of the specific problem.

Rangoli et al. 2018 explored a transformers-based object recognition model for distress detection purposes called (YOLOv4) which stands for You Only Look Once. The model demonstrates a noticeable performance as it was pre-trained based on the asphalt distress detection objective. The model registered a precision of about 0.87 in identifying asphalt distress from images. The crucial issue in that work is the quality of images taken by echoes, which can be attracted by various conditions like camera settings, position, speed of the vehicle, and degree of sunlight. All the mentioned factors may cause a decrease in the accuracy of any transfer model [10].

In 2024, Vinodhini et al. utilized a pre-trained AlexNet model modified by adding a final layer to detect asphalt distress from images. Despite that, the model achieved an accuracy of 0.96, but this work does not mention other metrics like precision and recall. Furthermore, the paper did not discuss the conditions that might affect the model’s efficiency, such as lighting conditions and weather effects in real-time implementations [11].

Apeagyei et al. (2023) proposed a deep convolution network (DCNN) pre-trained using TL to detect eight pavement distress classes. Seven models were used for comparison. Low false negative values specified the best models. However, despite their low general accuracy, various models performed well in detecting specific distress types. Researchers have noticed that image quality impacts model performance regarding prediction speed and precision [12].

Recently, a more efficient novel multi-head attention mechanism was proposed, which enhances the network's learning capability to focus on different locations separated in the feature space in parallel. The attention mechanism can improve the performance of machine learning tasks such as NLP, speech recognition, object detection, recommendation systems, and time series data tasks, for example, in forecasting and diagnosis [13]

. For example, Xue et al. (2021) used the multi-head attention layer with a transfer model in face expression recognition. The model relies on a multi-head layer to drop attention maps during learning. This technique makes the model focus on various local points in the face while ignoring the weak points or features [14].

Zhao et al. (2023) designed a multi-head attention based on two-stream EfficientNet. The proposed model architecture recognizes human actions. Their model consists of two streams utilizing EfficientNet-B0 to extract spatial/temporal features from the video. Then, the model incorporates multi-head attention to extract crucial key points from extracted features [15].

Hong et al. (2021) model consists of convolution extraction blocks and attention modules to detect COVID-19. The first two convolutional blocks are made up of two depth-wise separable convolution layers and a maximum pooling layer. The multi-head attention mechanism (MHAM) is used to extract effective feature information from COVID-19 X-rays and CT images. This mechanism allows the model to focus on different parts of the input image simultaneously, enhancing the ability to capture relevant features across various scales. The multi-head worked by taking various filtered CNN features and putting these features into a multi-head layer to get attention arrays for various image parts[16]. Accordingly, using the multi-head mechanism proposed in the Transformers architecture, a multi-head attention method can get the different channels that can effectively extract different features from the input. In our case, the multi-head attention mechanism can learn different features from the hidden vector to get the different features of the input as well as get the different features across the multiple hidden vectors projected in parallel as vectors at the same input and then reduce to the final number of features. Therefore, applying the multi-head attention mechanism to the detection model can enhance the feature extraction from the input data.

According to previous works, the poor performance of several deep-learning transfer models can be largely attributed to image quality. However, this sparked an ingenuity that helped formulate a new and effective strategy to enhance the distress detection of asphalt surfaces. Image degeneration results from different factors, such as being captured as a low-resolution image due to, for example, using mobile phone cameras under variable lighting and weather conditions. Moreover, the coarse texture of road surfaces and irrelevant objects (e.g., pedestrians, vehicles, or trees) may adversely influence the detection rate [14].

We conclude from all the above that the distress in asphalt pavements may cause a reduction in the service life of the road. Humans inspect the road by tradition; however, an objective and unbiased evaluation using computer vision techniques is crucial to aid human inspectors in decision-making. The main problem addressed in this paper is caused by the local variability in the surface texture, and the distress contributes to the difficulty of automatic detection. The difficulties also result from (1) the fast weather changes resulting in changes in road surface coloration or asphalt texture, (2) the various distress, and (3) the artifacts of the road expansion joints. Consequently, the following are the primary contributions of the current study:

1. Develop an accurate asphalt distress detection model using TL by improving the VGG16 model with a multi-head attention layer that consists of four heads. The multi-head attention layer focuses on crucial features extracted from the VGG16 model that formulates the pattern of asphalt distress. The attention layer with a pre-trained VGG16 model has been tested for the first time in this type of application.
2. Collect a local dataset for asphalt-damaged images from Baghdad streets. This dataset comprised three classes (cracks, potholes, and ruts). This dataset is the first national dataset, and its distinction comes from the uniqueness of crack shapes and potholes caused by the abnormally high temperature in Iraq, which may reach over 50 degrees Celsius.
3. Propose a series of pre-processing and augmentation of the dataset that are used to evaluate the proposed paradigm. These augmentation operations enlarge the dataset to contain images of the various intensity conditions.
4. Evaluate the model on real-time stream images to detect asphalt distress.

The rest of this paper discusses the following subjects. First, the related works and image pre-processing techniques are discussed, including the steps (smoothing, edge detection, and dilation). After that, the proposed model is illustrated in detail with results and discussions. The paper ends with a conclusion.

2. RELATED WORKS

In the past few years, researchers have conducted numerous computer vision-based studies with the specific aim of automatically identifying asphalt distress. These investigations employ various approaches, including Gabor filters [17], binary patterns [18], tree structure algorithms, and shape-based methods [19], among others. Although generally valuable, these methods require assistance to extract distinguishing characteristics from images to discern between non-cracked and cracked pixels. Furthermore, these techniques must enhance their ability to detect asphalt distress in real-world scenarios accurately, varying pavement textures and lighting conditions. Deep learning (DL), however, has demonstrated significant potential to address comparable problems and deliver superior accuracy results, notably through the utilization of DCNN equipped with TL – an approach that Gopalakrishnan et al. employed within the context of computer vision-based pavement distress detection [20]. After initial training with the ImageNet database, the DCNN detects pavement image cracks on Hot-mix asphalt (HMA) and Portland cement concrete (PCC) surfaces. The research achieved a significant increase in complexity by training a classifier using combined images of pavements featuring diverse surface properties - HMA and PCC. Optimal results are achieved when utilizing a single-layer neural network classifier that is pretrained on ImageNet and trained with features from the DCNN. Employing pre-trained DCNN models for cross-domain image classification, a general approach, has proven to be efficient in computer vision-based automated pavement crack detection. However, certain drawbacks were also observed, like the inclusion of non-crack characteristics such as joints, the inhomogeneity of cracks, and diversity within surface texture, all compounded by background complexity.

In 2019, Liu and his colleagues proposed a Deep Crack CNN model. This innovative approach featured multilevel convolutional layers. Additionally, demonstrating their commitment to advancing research, they introduced an invaluable dataset termed 'Deep Crack.' The utilized model, a variation of the VGG architecture, employed its first 13 layers. Deep crack with augmented data emerged as the most exemplary tested model; it yielded unprecedented performance in experimental tests, with an F-score and precision both measuring at 0.96 and recall registering at 0.86. The sole constraint identified in this work was the need to supplement the dataset with additional non-crack images [21]. In their 2020 study [22], Fan et al. introduced a system of multiple DCNNs specifically designed for automated crack detection and measurement in pavements. These CNNs, working collectively, recognize patterns of small gaps within raw images. They combine these findings to produce not only an overfitting-reducing result but also a predictive probability map. The approach outperforms alternative methods, achieving superior precision, recall, and F1 scores in evaluations using two publicly available crack databases. The proposed algorithm also facilitates the length and width measurement for various types of cracks. However, the suggested model faced two limitations: firstly, the system failed to detect cracks from the video streaming as it necessitates a more extensive and diverse dataset on which to offer performance evaluation, and secondly, an improved functioning is required, i.e., there is a need to test the system using more data. The researchers assembled a dataset consisting of 21,000 images taken from three different nations containing four different crack types.

Mandal et al. (2020) used three pre-trained models to detect pavement distress. These models were Hourglass-104, CSPDarknet53, and EfficientNet. The CSPDarknet53 model received the highest F1 score (0.58). Hourglass-104 came in second with (0.48), and EfficientNet came in third with a score (0.43). When compared to such models, the YOLO-based CSPDarknet53 model performed quite well; however, it has some drawbacks in terms of shadow-related conditions. In addition, EfficientNet encountered difficulties when attempting to locate cracks in roads that were wet [3].

A TL method was presented by Li et al. [23] in their article from 2021. This approach was designed to solve the difficulty of varying model performance across various types of cameras as well as mounting positions in the context of pavement distress detection. The approach is comprised of two primary components: model transfer and data transfer. The use of a distress detection model in unfamiliar settings is made possible through components that significantly reduce the requirement for considerable training data by no less than 25%. Also, such an approach enhances model accuracy by an amazing 26.55% compared to traditional approaches. Yet, it is essential to keep in mind that the efficiency of the training model could be affected by differences brought about by the use of multiple cameras that capture a wide variety of data and settings. This might potentially limit the potential for the model to be generalized. It turns out that obtaining labeled data for new scenes is very necessary, but given the framework that we have suggested, this process could cost a significant amount of time and effort. The utilization of GANs in data synthesis and transfer could result in a potentially hazardous circumstance. Distress annotations that have been created could contain inaccuracies or errors, which is one of the consequences that might have adverse impacts on model performance. It is necessary to perform manual screening of synthesized images after the completion of GAN style transfer to mitigate that danger. Even though this can be time-consuming and may require the removal of some training data validities, this stage is crucial for achieving optimal results. Errors or inconsistencies in the model's initial labeling could negatively affect the quality of the synthesized images and, consequently, the model's performance.

Smadi and Gosh (2021) used DL techniques, including YOLOv3 and Faster R-CNN, to perform the automatic categorization and identification of pavement problems from high-resolution 3D surface images. This was a powerful strategy. In terms of demonstrating robust performance, such models performed quite well, with an average precision rate for distress detection and classification reaching as high as 89.2% through Faster R-CNN. YOLO achieved an even higher level of efficiency, reaching 90.2%. A feasible alternative to manual Quality Assurance and Quality Control (QA/QC) methods is presented by the developed methodology. It reflects the outputs of QA/QC in an efficient manner, which is a big step towards streamlining operational procedures. One of the research's limitations is that its testing and training datasets are smaller than the image datasets that are typically used [24].

Abbas et al. (2021) used advanced image processing techniques to automate the detection of pavement distress like cracks and potholes. The proposed model employs various image processing techniques like mathematical morphology to identify cracks. In addition, the model used segmentation methods to improve crack detection, using dynamic segmentation techniques that relied on six segmentation algorithms. Their model outperforms ML models because of the dynamic optimization approach designed to handle noise better than traditional methods, allowing for more precise identification of crack patterns even in less-than-ideal imaging conditions. The proposed model can also detect the degree of curvature and make the model distinguish between potholes and cracks accurately. This model's limitations are associated with variability in lightning conditions, as the model's performance can be reduced under various lighting conditions. Second, environmental factors like strain lane marking may affect the view of the cracks. Third, the model's effectiveness depends significantly on the image's quality. Lastly, the model may not detect all types of pavement distress. These limitations resulted in the need for further collaboration to build a more generalized model [25].

During their research conducted in the year 2022, Zhu and colleagues [5] suggested using an Unmanned Aerial Vehicle (UAV) equipped with a high-resolution camera to collect pavement damage data. To train a dataset that contained images of pavements showcasing six different kinds of damage, they used three object-detection algorithms: YOLOv3, Faster R-CNN, and YOLOv4. The YOLOv3 algorithm produced a good performance with a mean average precision (MAP) score of 56.6%. This result considerably improves the effectiveness of non-destructive automated pavement condition evaluations. Yet, in order to have a comprehensive understanding of this study, additional information regarding dataset size throughout the training of the model is required. It is expected that the research would provide significant insights into the adaptability of trained models to a wide variety of types of pavement conditions and surroundings.

Yihan et al. (2022) introduced a new Transformer-based approach called LeViT for automatically classifying asphalt pavement images. LeViT's architecture incorporates convolutional layers, transformer stages, and two classifier heads. This method has been found to achieve excellent performance in terms of accuracy, precision, recall, and F1 score when tested on Chinese and German asphalt pavement datasets, surpassing the capabilities of existing state-of-the-art models. LeViT exhibits faster inference speed than the original Vision Transformer and other CNN-based models. Additionally, the paper proposes a visualization technique that combines Grad-CAM and Attention Rollout to enhance the interpretability of the results, while it does not provide information regarding overfitting [26].

Zheng et al. (2022) have contributed to collecting a benchmarked Pavementscapes dataset. Pavementscapes comprised 4000 images with a resolution of 1024 x 1024 pixels for each image. Several pre-trained DCNN models were examined. The CNN models used were variations from VGG16 to detect cracks, potholes, and ruts. The best model was the segmentation transform. The main limitation noticed while conducting this work is the inefficiency of detecting small damage instances [27].

Huang et al. (2022) collected a dataset of cracks called NHA12D. Their work is a comparison study between a set of state-of-the-art crack detection algorithms. The NHA12D dataset comprised 80 pavement images divided into 40 asphalt and 40 concrete images. Three models were tested using the proposed dataset: first VGG16, Deep Crack and ResNet 3. The Deep Crack model shows performance with a 90.3 recall and precision of 0.35 for asphalt cracks. Meanwhile, for concrete cracks, the recall was 0.96, and the precision was 0.25. The Huang et al. work's limitation was the failure to classify concrete joints from cracks [28].

Eslami et al. (2023) [29] examined the performance of DCNN classifiers in the context of automated pavement assessment. Among all the factors tested, using multi-scale inputs had the most significant positive impact, resulting in an average performance improvement of 20% as measured by the F-score. Interestingly, when distinguishing between road distress and non-distress classes, the CNN classifiers performed better on area-based objects (patches) than linear objects (cracks). The M-VGG19 model achieved the highest F-score and demonstrated reduced variation in classification accuracy across different class types. Additionally, adding more layers to shallow networks with fewer than four convolution layers improved classification accuracy, particularly for smaller objects. However, there are some limitations to consider in this study. Firstly, the paper should provide a more detailed explanation of the rules governing the DCNN classifiers used in the research. Secondly, it is essential to note that the study focuses exclusively on pavement assessment and does not explore the broader applications of DL algorithms. Furthermore, the paper must compare non-deep learning-based methods for classifying road objects. Lastly, the study should address the computational requirements and training time associated with the DCNN classifiers used in the research.

The model known as Crack Forest was designed by Shi et al. for crack detection using a function for specifying features relying on detecting cracks based on intensity inhomogeneity. The method is based on a random structure forest, which is an improved ML method that combines algorithms for learning patterns to make decisions without being programmed explicitly. The proposed algorithm is superior to the previous models. Crack forest based on the SVM classifier registered a precision of about 90.28%, recall 0.86, and F1 89.39%. The only limitation of this work is that video streaming was not taken into consideration [30].

All the previous researchers either implemented existing transfer models or proposed their own models. They implemented the models directly on the images without pre-processing steps. In contrast, the current study presents an innovative improvement on the pre-trained CNN VGG16 model by adding a multi-head attention layer with a vast number of augmented images.

Cano-Ortiz et al. (2024) focused on comparing various YOLOv5 variants. These models have been evaluated according to their efficiency in detecting the targeted objects. Their main contribution is a novel filtering post-processing mechanism. This filtering mechanism is used to reduce false positive detection by 20.5%. The proposed post-processing mechanism relies on a rule-based approach that includes several rules to reject overlapped detection cases. The proposed model was evaluated on two datasets and achieved a precision of about 0.56 and 0.57 for the RDD2022 and CPRI datasets, respectively. This study has limitations because it evaluated the model on the existing dataset, which may not accommodate real-world conditions. Also, this study focused on one type of distress, cracks, which means that this model may not assess all types of distress comprehensively. Lastly, the results were validated on an open dataset that raises concerns about the generalization capabilities of the proposed architecture [31].

A. Nasertork et al. (2024) designed a model to improve the detection of pavement distress inception. This work utilized a proposed image processing feature extraction with AI techniques. The proposed model combined a set of texture features such as Gray Level Co-occurrence Matrix (GLCM), Local Binary Patterns (LBP), and Histogram of Oriented Gradients (HOG). All these features are used as discriminators to detect pavement images. Various ML algorithms trained by the extracted features, including XGBoost (XGB), Logistic Regression (LR), K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Random Forest (RF), Artificial Neural Networks (ANN), and Convolutional Neural Networks (CNN) were used. The best classifiers that achieved accuracy higher than 90% were SVM, XGB, and KNN. The most crucial limitation of this work was the variation of the feature selected effectiveness that relied on the dataset itself. The datasets were limited, leading to the specific model, because the model's training and evaluation depended on the quality and diversity of the image dataset [32].

M.Guerrieri et al. (2024) employed a pre-trained DL model YOLOv3 detection algorithm, which is known because of its efficiency in real-time object detection. YOLOv3 utilized Darknet, which used 3x3 filters inspired by ResNet that efficiently detect small objects in real-time. The dataset includes a diverse range of pavement damage types. The variety in the dataset allows the model to learn and extract relevant features that distinguish between different types of distress, enhancing its classification capabilities. The limitation of YOLOv3 is its difficulty in managing scale variations, especially when detecting small or large objects. The model faced a challenge from relying on a public dataset for training and validation. While this dataset provided substantial data, it may not encompass [33].

K. Ijari et al. (2024) utilized the EfficientNetB3 architecture, one of the EfficientNet variations. This model is notable for its compound scaling method that optimally adjusts depth, width, and resolution. The EfficientNetB3 model achieved superior performance with fewer parameters than traditional models like ResNet. The model detects various types of distress, like cracks and potholes. The researchers utilized a Swin Transformer-based GAN to generate images of synthetic pavement cracks. This augmentation was crucial for improving the efficiency and accuracy of the pavement damage assessment process. The efficientNetB3 model, when combined with the SwinGAN data augmentation process, achieved impressive testing accuracy ranging from 76.7% to 78.2%. This paper addressed a set of limitations. First, data quality issues, such as poor data quality, can lead to inaccurate predictions and classifications. Second, the model's sensitivity to environmental factors: the model's performance can be adversely affected by environmental factors such as shadows, reflections, and road markings, which can introduce noise into the image data. Third, handling complex crack geometry because the study identifies the lack of existing models for complex crack topologies. Many CNN models face difficulties classifying cracks with irregular shapes, which can decrease their accuracy in real-world applications [34].

3. BACKGROUND

This research paper proposes a model for asphalt distress detection. The proposed model is based on a pre-trained model that relies on improving the VGG16 with a multi-head attention layer. Before that, a batch of processes was conducted to prepare the datasets prior to training the model. These processing procedures included a smoothing process and then data augmentation processes. The following sections illustrate the methods used in implementing the proposed system. These methods make the datasets more suitable for efficiently training the model to produce more accurate results.

3.1. SMOOTHING PROCESS

Smoothing, also known as averaging, is used to smooth any image by spatial filters to reduce sharp details in images. It is used to lessen the sharpness of irrelevant details in the image [35]. A bilateral, linear filter replaces each pixel's intensity with a nearby pixel's average weight. The bilateral smoothing can preserve edges at the same time. Each neighbor is weighted by spatial components, considering the distant pixels and the difference between pixels of various intensities. Their combination value ensures that only nearby similar pixels contribute to the final pixels of the same region. Bilateral works are based on Eq.1 for each pixel p and q, whose loop is nested within p. The equation relies on taking the central pixel p and its neighborhoods such that $|p\text{-}q| < 2\ \sigma s$, considering the contribution of pixels outside the range of σs is negligible because of the spatial kernel.

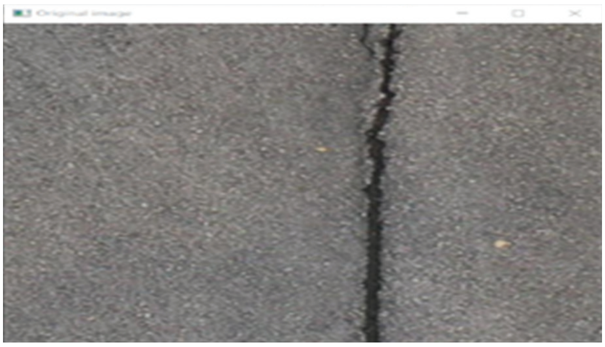
$$g\ \sigma s(||q-p||)g\sigma s(f(q)-f(p)).(f(q)) \qquad (1)$$

It is used for unwanted texture removal. In our approach, the granular texture in asphalt must be removed [36]. (Fig. 1) illustrates how the smoothing preprocess has been applied to the original asphalt image.

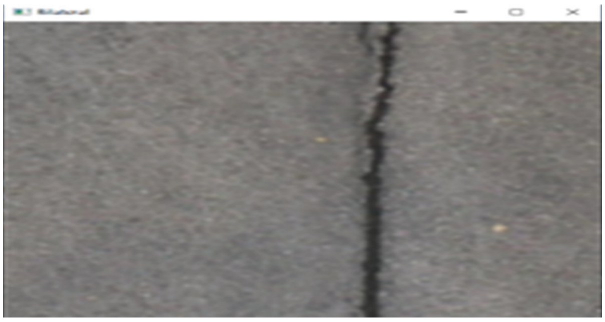
3.2. TRANSFER LEARNING

TL models are those pre-build models trained using a specific dataset to be used later for building new models for various purposes using a different dataset. The *TL* principle relies on generating a model for one purpose and utilizing it for other activities. In these models, knowledge is gained from previously trained models on past tasks. As a result, this paradigm is beneficial when the data is limited because the limited data makes the model difficult to generalize. In addition, *TL* makes the model faster to train than developing the training model from scratch. The idiom of transfer learning comes from transferring existing knowledge to learn a new model with or without a labeled dataset [37]. (Fig. 2) summarizes the principle of *TL*.

The source dataset domain D_s and training task Tt, and target domain Dt with training model for task Tt. The target of *TL* is to use the knowledge in Ds and Tt to learn the targeted prediction model f in Dt given that $Ds \neq Dt$ and $Ts \neq Tt$ [37].



(a)



(b)

Fig. 1. Crack image **a**) Before smoothing, **b**) after smoothing

3.3. ATTENTION LAYER

The paradigm of using the attention layer was inspired by how humans penetrate a specific region of a scene. The attention layer works as a spotlight within the architecture of neural networks, specifying essential features in an image.

Therefore, the neural can adaptively adjust the neural weight according to the image features to learn from special regions in an image [38]. Special attention is widely used to focus on specific regions in any image. Generally, the operations of the attention layer consist of the following steps. First, compute addressing scores between various regions of the input image, such as pixels in an image. Second, weights are determined based on scores to indicate the importance of areas. Third, weight is used to refine the output, focusing on the most relevant features [39].

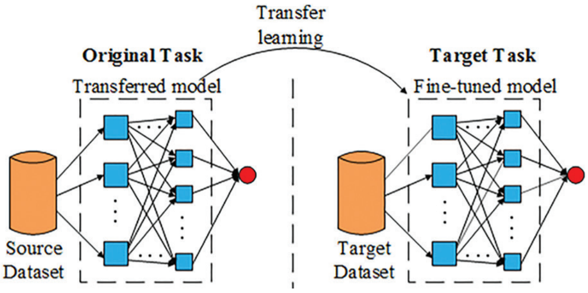


Fig. 2. TL for a new model [36]

Self-attention can be defined as an attention spatial filter applied to a single context or pixel instead of multiple contexts. So, queries, keys, and values are used to extract features from the spatial domain.

For example, in (Fig. 3) below, to extract the feature set of a pixel $xi, j \in R^{din}$.

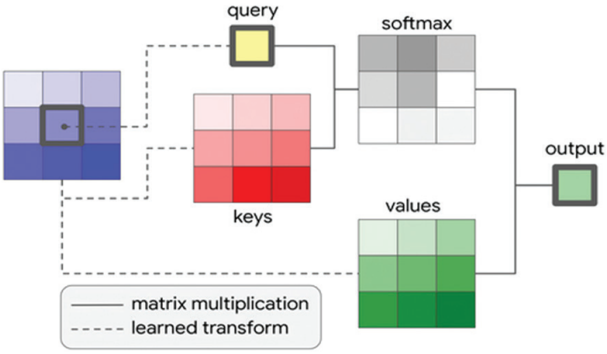


Fig. 3. The technique of attention layer [36]

As a spatial region, the attention layer has to extract the local regions of pixels in position around the specific pixel xi, j or $\in Nk\,(i, j)$. Then, the single-head attention process is computed by applying *softmax* on the query multiplied by the key to get the attention-focused features in the Eq. 2 below

$$yij = \sum_{ab\,\in Nk(i,j)} Softmax(qij^{T}kab)vab \qquad (2)$$

Where qij^T is the query, kab is the key, and vab are values $Wvxab$, which represent the transformation of the pixel in position ij and the surrounding pixels. The softmax operation is conducted on all learned transforms. Self-attention works in a way that is similar to a spatial convolution filter by collecting information over the pixel and its neighbors, and the aggregation is done by using a convex combination of value vectors by using the softmax function [40]. This operation is repeated for every pixel in an image.

The advanced type of attention mechanism is the multi-headed attention that is illustrated in (Fig. 4) The multiple attention heads paradigm is used to gain various representatives for the input. The multi-head attention begins with partitioning the pixel features into N parts $xi, j \in R^{din/N}$ by conducting a single-head attention operation on each group separately with various transformations $W^n\ Q$, W^nK, $W^nv \in R^{dout/N\ x\ din/N}$ for each head. Then, concatenating the output from each head into one final output $yij \in R^{dout/N}$. CNN begins by extracting the features from the image, and then the attention layer maps the essential features using the weighted average of the values. Lastly, the multi-headed attention blocks output are concatenated into one feature set [41].

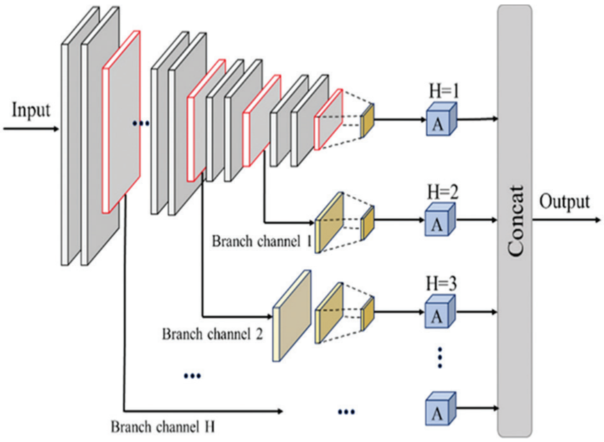


Fig. 4. The mechanism of multi-head attention layers [40]

3.4. DATA AUGMENTATION

Data augmentation is a technique used to generate new training samples from the existing seed of the dataset. This operation is like taking from the existing training samples and producing modified copies to train any classification model. There are various augmentation processes:

- Geometric transformations: These operations alter some geometric features in images like flipping images, cropping parts from an image, scaling an image (zooming in, zooming out), rotating to a specific degree, and shearing by distorting the image along an axis to rectify the perception angles.
- Color space augmentation: relying on modifying color within an image. The image is augmented by changing lights, saturation, and hue. Changing the colors within images can add realistic light variations and other color elements. This process makes the model less susceptible to overfitting.
- Noise injection is a technique in image augmentation that depends on adding a specific amount of noise to existing samples of images. This injects variations that simulate real effects or camera noise. For example, Gaussian noise, which is commonly used, is implemented by adding random values with normal distributive values to each image pixel. The intensity of noise is controlled by standard deviation [42].

4. THE PROPOSED MODEL

The proposed asphalt distress detection model has been inspired to detect three types of asphalt damage: cracks, potholes, and ruts. This system overcomes the issues noticed in previous works, as some systems have low accuracy or overfitting. The issues in accuracy came from datasets with a small number of images in each class or specific classes or poor-quality images. The proposed system consisted of stages. This work proves the system is free from overfitting because the model must be generalized to unseen data, not just memorize the training data. (Fig. 5) represents the proposed system stages.

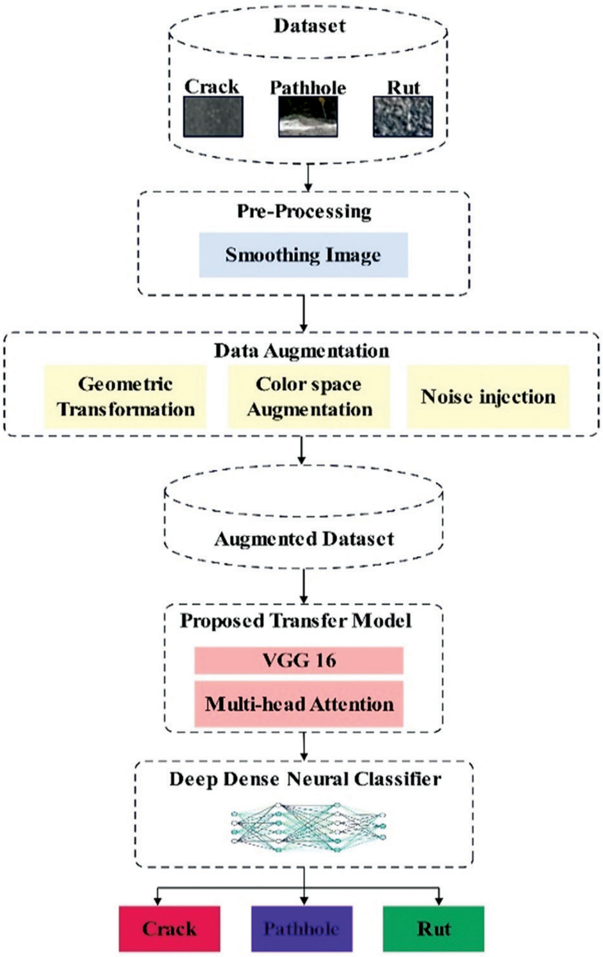


Fig. 5. General view of the proposed model

In Fig. 5, the model begins with the pre-processing stage to enhance the asphalt image quality. This enhancement is conducted by applying a smoothing operation to eliminate the granular shape of the gravel within the asphalt texture because pieces of gravel add noise to the image, especially when the system tries to detect cracks from non-cracked asphalt cases. The result of pre-processing is saved in a dataset repository.

The second stage is data augmentation to increase the number of training samples that make the system more generalizable. The augmentation process is useful in asphalt distress classification because it aids the classification model in being responsive to variations of the scenes in the real world. For instance, asphalt cracks may look different according to various lighting conditions. So, by implementing augmentation operations like cropping, flipping, blurring, and adding noise, we can make the dataset wider, containing various cases of images that will help the model learn as much distress in as many conditions as possible. (Fig. 6) Shows the steps for each of the augmentation operations

According to (Fig. 6), we begin the data augmentation with geometric transformation. The first process in geometric transformation is flipping; flipping the image from left to right horizontally helps the model to identify the distress pattern regardless of the image orientation. The second geometric transformation is cropping part of the image because the image does not always capture the entire area of interest. Consequently, cropping helps the classifier detect the distress area even when the damage does not occupy all the image scenes. The next operation is scaling the image by zooming in and zooming out. Zooming in can make the model focus on a magnified area. Zooming out lets the model learn the pattern from a more comprehensive view.

Consequently, these provide a broader context, allowing the model to learn to identify larger-scale distress features like potholes. The fourth process is the rotation of the image by a slight random angle. Rotation may simulate the variations in camera orientation. The last geometric transformation is shear, which tilts the image slightly in a specific direction, either horizontal or vertical. The next batch of operations is the color space augmentation, which begins with the flowing operations. First, Gaussian blur is an image augmentation technique widely used in computer vision tasks such as asphalt distress detection applications. Gaussian blur applies a Gaussian filter to the image, yielding a smooth image by blurring its details. The second is multiplying the image by a value greater than 1 to increase brightness.

In contrast, multiplying the image with a value less than 1 increases the darkness of the image. Third, contrast normalization is a data augmentation technique used in image processing to improve the overall contrast and visibility of features within an image. It is particularly helpful for DL tasks where models rely on extracting meaningful features from image data. The last augmentation process noise injection is implemented by adding Gaussian noise. It involves adding random noise following a Gaussian distribution to the image. This injects a controlled level of "artificial noise" that mimics real-world variations or sensor imperfections.

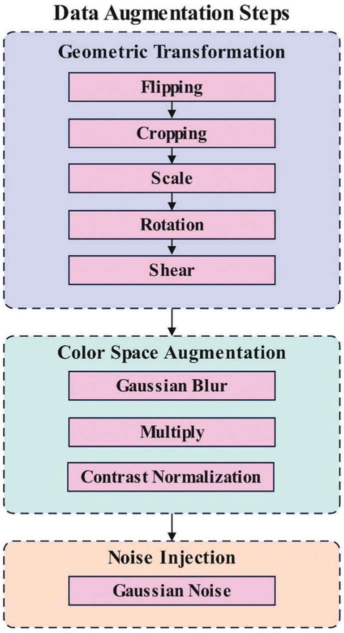


Fig. 6. Image augmentation stage process

The third stage is the proposed VGG16 model with a multi-head attention layer that has to be trained using the prepared dataset. (Fig. 7) illustrates the details of the improvement.

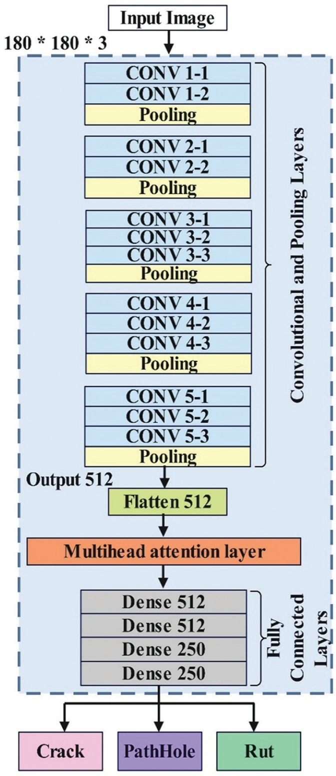


Fig. 7. VGG16 with multi-head attention layer structure

The image is size 180x180 and has three channels entering the model. The first part of the model is the VGG16, which consists of five layers. Each layer has CNN filters and one pooling layer. The output feature set from the VGG16 layer consists of a two-dimensional 512 array. Then, this feature set is flattened to be a one-dimensional array. The flattened feature set is the input to the multi-head attention layer that focuses on different parts in the feature set to extract the best representative for each image. The final output from the attention layer enters the final component of the deep dense neural network DDNN. This DDNN is used to classify the input image into one of three classes after learning from the features of each image during the training epochs.

(Fig. 8) illustrates the details of the mechanism used to extracting the features from the dataset's images. Initially the image was divided into four parts. Each of the parts entered the VGG16 transfer model. Extracting features from the image parts gives the attention process a richer understanding of the data. This can lead to more accurate and informative attention weights. In addition to that, dividing the image into parts allows the system to catch different aspects of the input image. By concatenating them, we allow the attention mechanism to consider these various aspects simultaneously, potentially leading to a more comprehensive understanding of the data. All the feature sets extracted by VGG16 are flattened into a 512 array. These feature sets enter the process of a multi-head attention layer that consists of four heads, one head for each part of the divided image. Accordingly, the number of iterations within the multi-head attention layer would be 16. Each element corresponds to a different region or aspect on the input image. Multiple heads allow the capture of various patterns. The multi-head attention layer begins the iteration 16 times by choosing arbitrary values of K and Q. Q: Represents the information you want to attend to, K Represents the information you want to attend with, and V: Represents the information you want to retrieve if there's a match between query and key. (Fig. 9) represents the details of each attention head. Fig. 9 shows the architecture of the single-head attention layer. The similarity between the query and key is scaled between +1 and -1, calculated by finding the dot product of two vectors. Multiplying the key (K) and query (Q) yields an attention filter.

$$A = QK^T \tag{3}$$

Then, scale the attention scores in the attention filter. The attention filter scores enter the softmax process to get more detailed crucial features, as in Eq. 4.

$$A_{softmax} = softmax(A) \tag{4}$$

After that, the attention filter is multiplied by the original image to remove unnecessary details. The features set is concatenated with the original image to obtain a more focused and detailed final image. The concatenated values are projected back to the original dimensionality using a projection matrix W_O:

$$Output = (X \parallel V(A\_softmax))\, W_Q \tag{5}$$

The multi-head attention allows one to focus on various parts of the image. So, each attention head outputs an attention filter that may focus on different details inside the image.

The proposed VGG16 with a multi-head attention layer has been developed using a mathematical model, and the details of the mathematical model are in the following steps:

4.1. CONVOLUTION (FEATURE EXTRACTION IN VGG16)

The VGG16 operations are repeated four times to get the feature set map. Suppose I represent the input image of a 3-dimensional tensor (height, width, channels). The filter W is the learnable weight of 3 dimensions. The convolution operation of VGG16 to extract the feature set is as follows:

$$O_{ij} = \Sigma\big(W_{khw} * I_{(i+k)(j+h)(c)}\big) + b \tag{6}$$

Where O_{ij} is the value of the output of position (*i, j*) in the feature map. W_{khw} is a single value from the filter W. $I_{(i+k)(j+h)(c)}$ represents a specific pixel value in the input image at a shifted position (*k, h*) within the kernel and channel (*c*). b is the bias term for that particular feature map.

4.2. POOLING

The pooling function is used to select more significant features by applying either average pooling or max pooling. Pooling also reduces the dimensionality of the feature set. The max pooling function is applied after conducting each CNN layer as in the equation.

$$O_{ij} = max(F_{(i+k)(j+h)}) \tag{7}$$

For all k, h within the window, where $F_{(i+k)(j+h)}$ is a single element in the features set.

4.3. MULTI-HEAD ATTENTION:

This process is used to extract the distress region pattern within the asphalt images and produce a feature set that is focused on the distressed parts in the asphalt. The attention process is done by conducting self-attention with many parts within the image to extract the more crucial part in deciding the image class. So, suppose X is the flattened feature vector extracted by the VGG16 pretrained model. Now, define three weight matrices, W_Q, W_K, and W_V, for projecting the input vector into a query (X_Q), key (X_K), and value (V) vectors, respectively.

The attention process is implemented as in the equation:

$$S = X_Q * X_K^{\,T} \tag{8}$$

Where S is the attention score.

After that, the softmax is applied to extract important features from the attention layer as in the equation:

$$A = softmax(S) \tag{9}$$

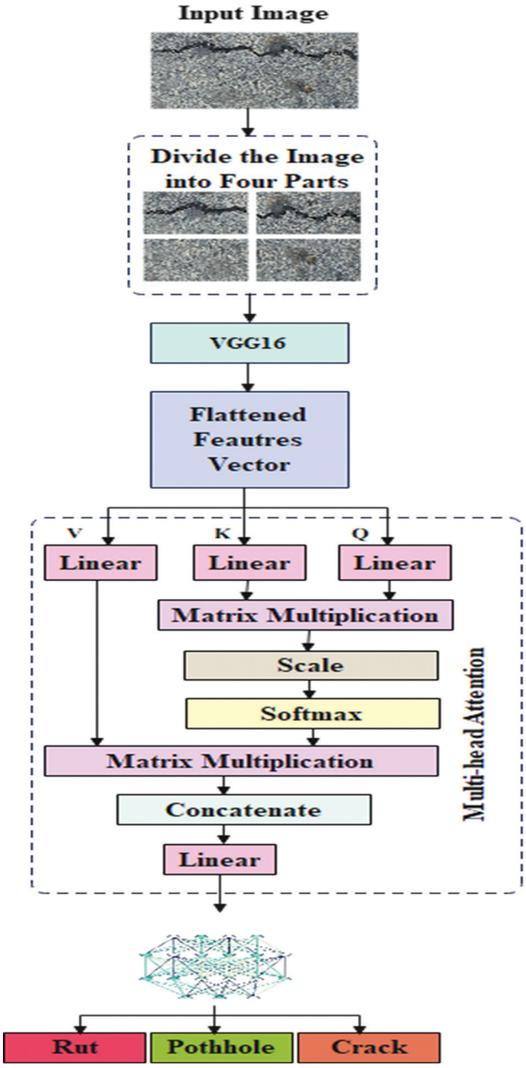


Fig. 8. Detailed architecture of the extraction features process in the proposed model

Now highlight the significant feature by multiplying the value of the image by the max-pooled feature set in the equation below:

$$\text{Context vector: } C = A * X_V \qquad (10)$$

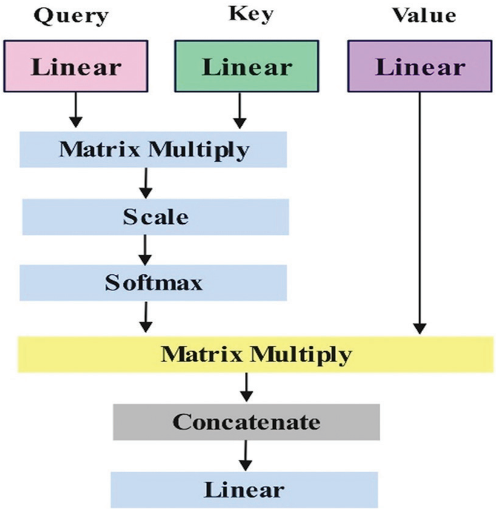


Fig. 9. Attention layer architecture

4.4. DENSE LAYERS

A deep dense classifier consists of four layers, trained over the features extracted from the multi-head attention layer to predict the input images later as cracks, potholes, or ruts. So, C represents the features vector, W is a set of weights, and B is the bias vector. The decision of an image is calculated by implementing the equation:

$$Y = ReLU(W * C + b) \qquad (11)$$

So, the image is predicted by multiplying the W after training to the feature of an image after adding the bias vector.

4.5. SOFTMAX (OUTPUT LAYER)

SoftMax is implanted on Y to reach the final decision about the image class, which is either a crack, pothole, or rut. So, suppose Y is the output vector from the final dense layer. Then implement the softmax function for each class probability (i):

$$P(i) = exp(Z_i) / \Sigma(exp(Z_j)) \qquad (12)$$

For all classes j, the last stage in the proposed model is the deep dense neural that accepts the feature set from the proposed VGG16 to decide the input image to which class it belongs.

5. EXPERIMENTAL RESULTS

This section presents the tests conducted on four datasets for various transfer models for computer vision problems. Initially, the experimental environment, datasets, and evaluation metrics must be explained.

5.1. EXPERIMENTAL ENVIRONMENT AND DATASET

The experiments were conducted using Python version 3.1.10 on Windows 10, CPU core I 7, and GPU Gforce 940 MX to accelerate the data training time while the transfer models are executed. Libraries like TensorFlow and Keras were used to build the proposed models. Three well-known benchmark datasets were utilized and divided into training and testing sets with a proportion of 0.8 training and 0.2 testing sets to evaluate the performance of the suggested models more precisely. The details of each dataset are as follows:

- The Pavementscapes dataset conducted by Zhang et al. contained 4000 images, each with a size of 1024*1024 pixels. The dataset was labeled with six classes related to asphalt detection. In this work, we use only the data related to cracks, potholes and ruts, which are 2300 in total. (Fig. 10) illustrates the three types of asphalt distress.

- The Deep Crack dataset contains 537 RGB color images, each of which is a fixed size of 544*304 pixels. The dataset was annotated manually and labeled into two classes: cracks and non-cracks.

- NHA12D dataset consists of 80 pavement images divided into 22 cracks and 58 normal images. Each image has a size of 1920*1080 pixels.
- Iraq asphalt dataset: This dataset was collected from Iraqi streets because there is a need for a national dataset because of the shape of the cracks and distress in this country. The asphalt distress in Iraq is produced by high temperatures over 50 degrees Celsius and unauthorized digging. The dataset images were collected using a mobile camera with 48 megapixels. Each image consisted of 3000 x 4000 pixels and was saved as a JPG file. The photos were labelled by the research group of this paper and consisted of 250 for each class (cracks, potholes and ruts). Anyone who wants the data should contact the authors.

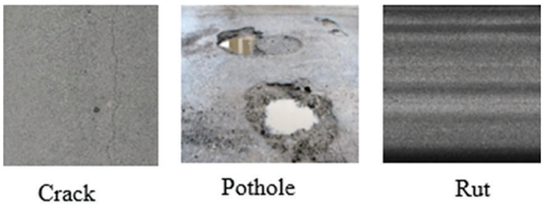


Fig. 10. Three types of asphalt distress under the study

5.2. EVALUATION METRICS

Four evaluation metrics were used to check the proposed model's efficiency. First, accuracy in Eq. (12) measures the model and predicts the outcomes correctly.

$$Acc = \frac{TP+TN}{all\ predictions} \tag{12}$$

Second, precision in Eq. (13) represents how often the model correctly predicts the positive class. Precision will be better when it is closer to 1.

$$precsion = \frac{TP}{TP+FP} \tag{13}$$

Third, recall in Eq. (14) measures how often classification learning correctly identifies positive instances of the positive class. Recall will be better when it reaches 1.

$$Recall = \frac{Tp}{TP+FN} \tag{14}$$

Lastly, the harmonic mean of precision and recall.

$$F1 = 2 \times \frac{precsion \times recall}{precsion+recall} \tag{15}$$

We have to introduce the following idioms to understand the metrics used to evaluate the performance of the models. (Fig. 11) explains the components of the confusion matrix. True positive Tp represents when the classifier correctly predicts an instance related to a positive class. For example, when the model predicts an image holding crack damage to the crack class. True negative TN represents when the model correctly predicts an instance related to a negative class. For example, when a classifier predicts an input image without a crack as a normal image without damage. False positive FP means the classifier model incorrectly predicts a case as positive when it belongs to the negative class. For example, if a normal asphalt image is classified as a damaged case. False negative FN represents an error case. This happens when the predictor model mispredicts an instance as negative. For example, an image of damaged asphalt can be classified as normal.

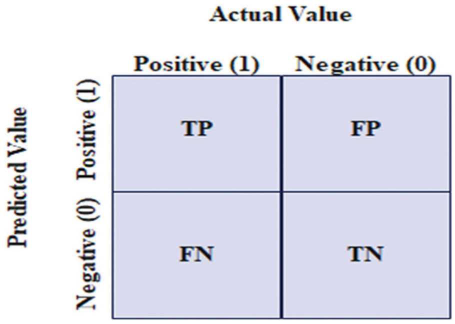


Fig. 11. Confusion Matrix Shape

6. RESULTS AND DISCUSSIONS

The performance of this proposed asphalt distress detection model was evaluated relying on matrices of accuracy, precision, recall, and F1 score. We depend on Macor's average accuracy with related matrices because the datasets are imbalanced. This research tested the possibility of overfitting by plotting the difference between training and validation accuracies and loss during the training epochs. Table 1 illustrates the performance of the proposed model.

The proposed pavement distress detection model was conducted on four benchmarked datasets, including the IRAQ asphalt dataset. The Pavementscapes dataset, consisting of three distress types (cracks, potholes, and ruts), has been used to evaluate the proposed model. The precision and recall for cracks both reached 1.0. At the same time, the general macro average precision and recall were 0.99. Pothole precision is 1.00, while the ruts precision reached 0.99. Ruts predicting achieved a true positive of about 0.99, as in the confusion matrix in (Fig. 12). The imbalanced data caused these differences between the precision values of classes.

Table 1. Performance Of Various Transfer Models

Dataset	Precision	Recall	F1	Accuracy
Pavementscapes	0.99	0.99	0.99	1.00
NHA12D	0.99	0.99	0.99	0.99
Deep Crack	1.00	1.00	1.00	1.00
Crack Forest	1.00	1.00	1.00	1.00
IRAQ dataset	0.96	0.96	0.96	0.96

The behavior of the proposed system towards the possibilities of overfitting was acceptable during the training process. (Fig. 13) registers the system's accuracy during training epochs, which refers to high training accuracy compared to the validation accuracy in the same training cycles. At the same time (Fig. 14) shows the difference between the validation loss and training loss. Validation loss was lower than the training loss in all the training epochs. An early stop mechanism was used to terminate the training process in five epochs to ensure achieved weights for the model.

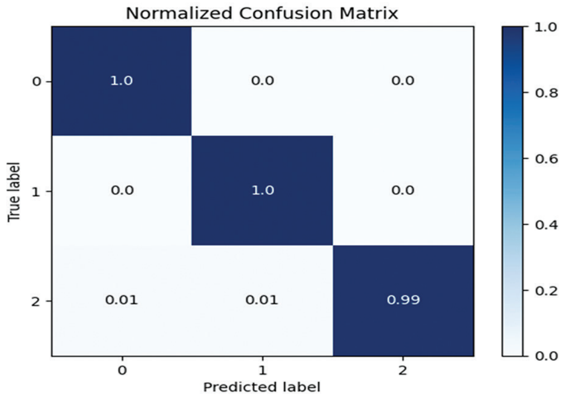


Fig. 12. Pavement scape dataset confusion matrix

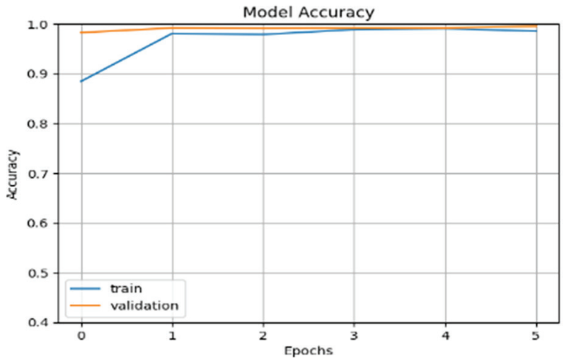


Fig. 13. Model training accuracy to validation accuracy for Pavement scape dataset

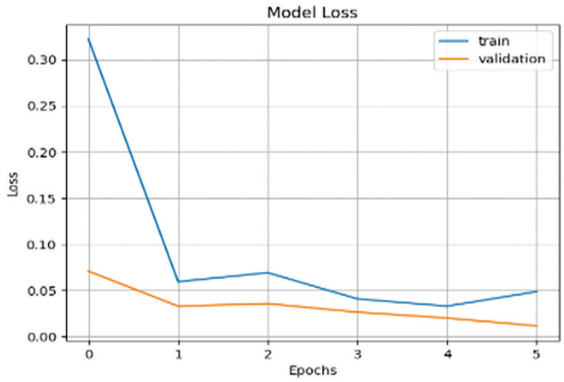


Fig. 14. Model training Loss to validation loss Pavement scape dataset

NHA12D was also used to check the validity of our paradigm, although this dataset consisted of two classes (crack, non-cracked) of asphalt images. The precision and recall were 0.99 despite the difference in precision between cracks and non-cracks classes. The model predicts crack classes with a precision reaching 0.98, while the precision for the non-cracked class was 1.00. This small difference is because of the imbalanced dataset. The confusion matrix is clear in (Fig. 15). The 0 label refers to the crack class, and the 0 class refers to the non-crack class.

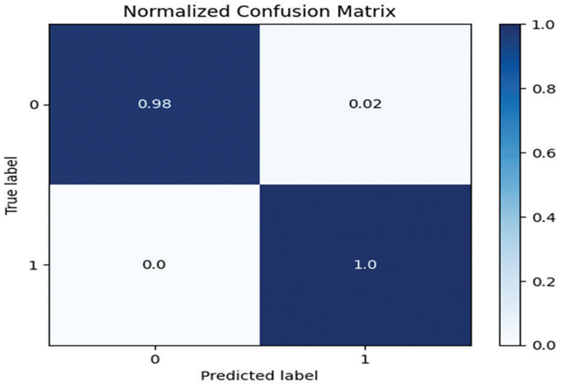


Fig. 15. Model confusion matrix on the NHA12D dataset

In this experiment, we also used an early stop mechanism; the model needed seven epochs for training. (Fig. 16) shows that the validation accuracy is close to the training accuracy during the training process. In contrast, the validation loss was lower than the training loss except for the last epoch before conducting the early stop, as shown in (Fig. 17). Model training accuracy. Early stop is used to prevent any possibility of overfitting or overtraining in the process.

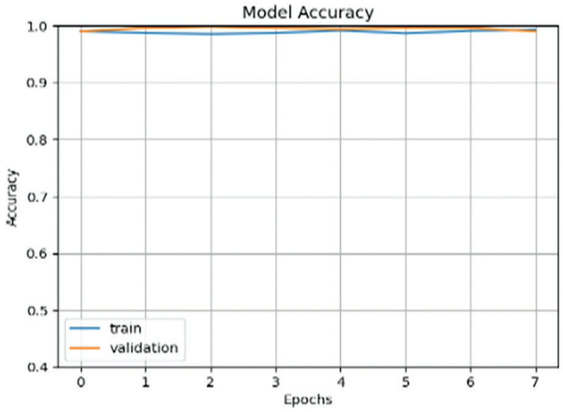


Fig. 16. Model training accuracy relate to validation for NHA12 Ddataset

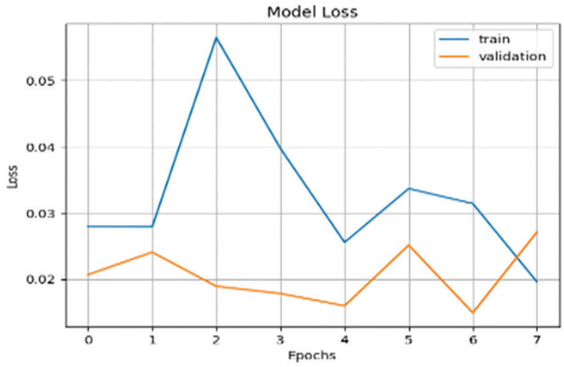


Fig. 17. Model training loss relate to validation loss for NHA12 Ddataset

The Deep Crack dataset also contains two classes (cracks and non-cracks). The cracks of multiple scales and scenes make this dataset a crucial benchmarked dataset to evaluate crack detection models. The dataset is relatively balanced, resulting in an efficient model for detecting cracks and normal asphalt without damage. The model achieved high precision and recall in this dataset, reaching 1.00 in the prediction of both classes. The confusion matrix is clear in (Fig. 18). Label (zero) represents the crack class, while Label (one) represents the non-cracks class in the confusion matrix.

The model's behaviour during the training was also investigated by plotting accuracy and loss. (Fig. 19) shows the accuracy during eight epochs of the training process. The X-axis in the figure represents the number of epochs, while the y-axis represents the accuracy. In all the training epochs, the validation accuracy was close to the training accuracy in a consistent trend between the two groups.

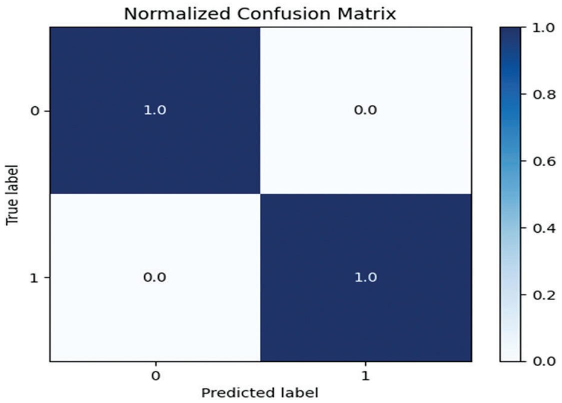


Fig. 18. Model confusion matrix for the deep crack dataset

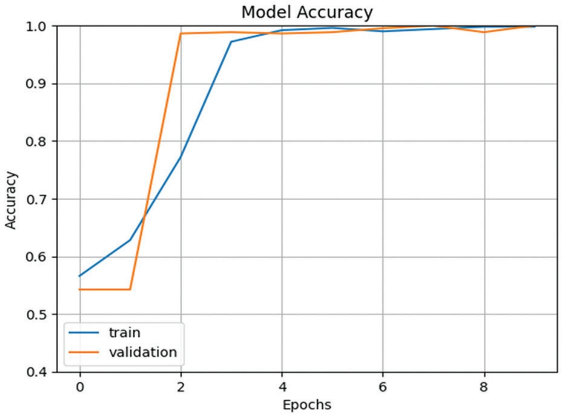


Fig. 19. Model training accuracy towards validation in deep crack dataset

In (Fig. 20), the validation loss is less than the training loss during the training process. Both accuracy and loss plots refer to the efficient behavior of the model in detecting new instances of cracks and non-cracks during the training and the trend of the model to stay away from overfitting in the working process.

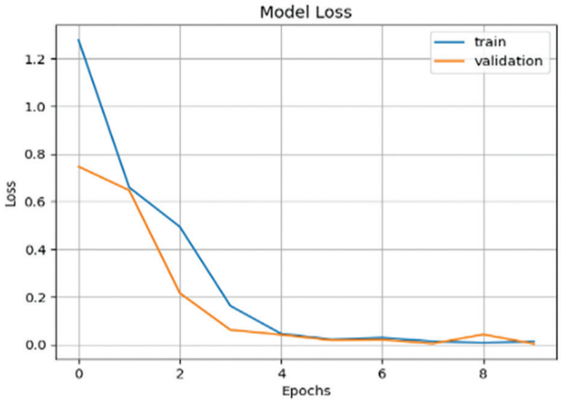


Fig. 20. Model training loss towards validation loss in deep crack dataset

The fourth test dataset was a Crack Forest that consisted of two classes (crack class and non_cracked class). The precision and recall for the cracked asphalt class were consistent and registered 1.0. The non-cracks class precision was 1.0; in contrast, the recall was 0.96. The low recall of the non-cracks class is due to the small number of images in the test set. The model's accuracy was 1.00, according to the confusion matrix in (Fig. 21).

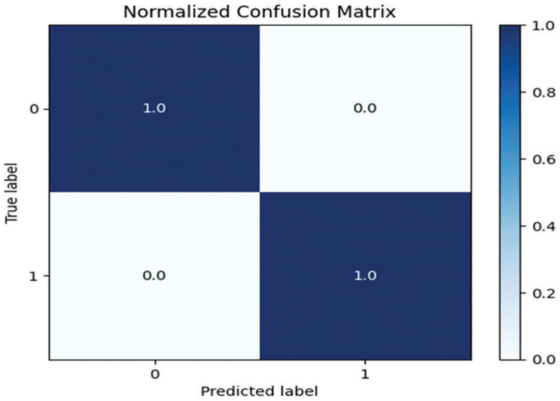


Fig. 21. Model confusion matrix on Crack Forest dataset

The model's behavior towards overfitting was apparent in the trend of increasing accuracy within training epochs, as in (Fig. 22).

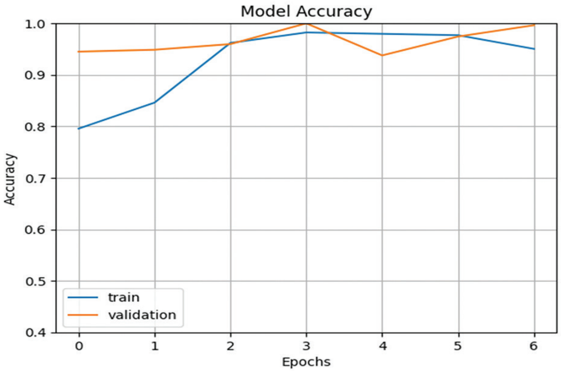


Fig. 22. Accuracy of the training model for crack forest dataset

Validation accuracy increases gradually during the model training. The loss of the validation also went lower than the loss of the training set except for the fourth epoch, which scored lower in the last two epochs, as in (Fig. 23).

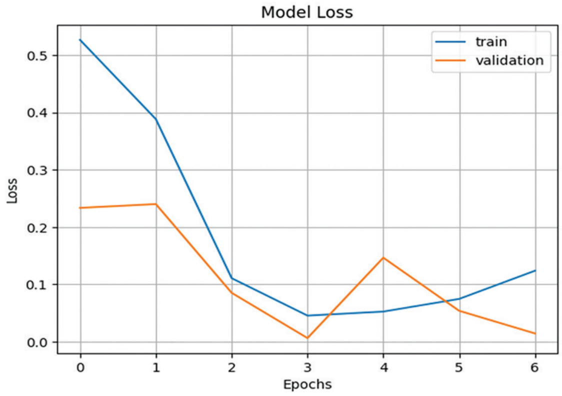


Fig. 23. loss of training model for crack forest dataset

The system has been tested on the Iraqi distress dataset. This dataset contains images taken under bright sunlight. Additionally, this dataset is distinguished by unique crack shapes due to the high temperature and drilling works, as mentioned. The system achieved an accuracy of 0.96. According to the confusion matrix in (Fig. 24), cracks were detected with an accuracy of 0.97. In contrast, potholes were detected with an accuracy of 0.91, and ruts were detected accurately in 1.0. The reason behind the low accuracy of pothole detection is that some overlap with cracks, as some images contain cracks and potholes at the same time. (Fig. 25). tracks the difference in training versus validation accuracy during the training process. In all 18 training epochs, the validation accuracy was close to the training accuracy.

In (Fig. 26), by comparing the loss validation to the loss of training in the 18 epochs, we can notice that the validation loss was generally less than the training loss, especially from epoch ten forward. Fig. 25 and Fig. 26. Refer to the fact that the system was far from entering the overfitting case within the training epochs.

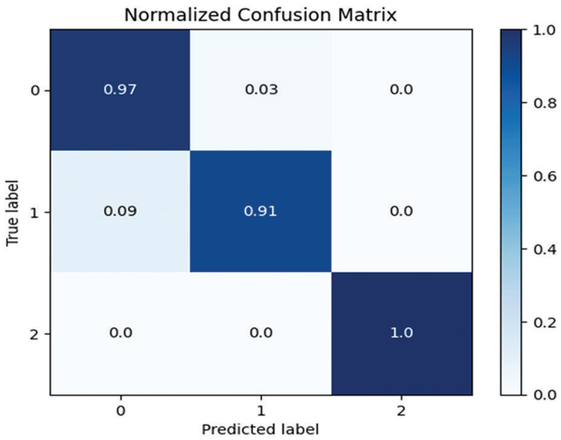


Fig. 24. Iraqi dataset confusion matrix

The system shows efficient behavior for both asphalt concrete distresses. The dataset Deep Crack contains the cracked concrete images used to train the proposed model in this research. (Fig. 27) represents a crack in concrete.

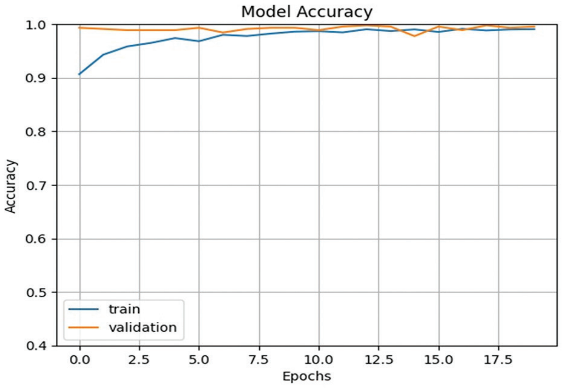


Fig. 25. Training accuracy of Iraqi dataset

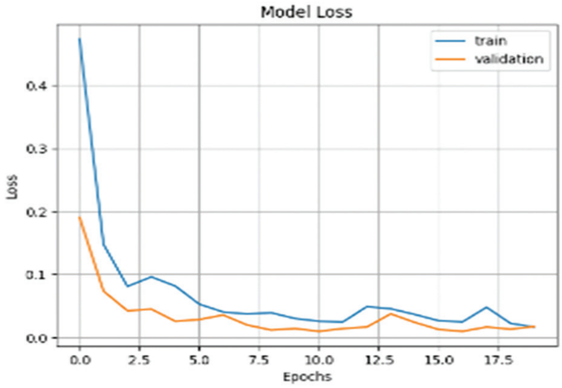


Fig. 26. Training loos of Iraqi dataset

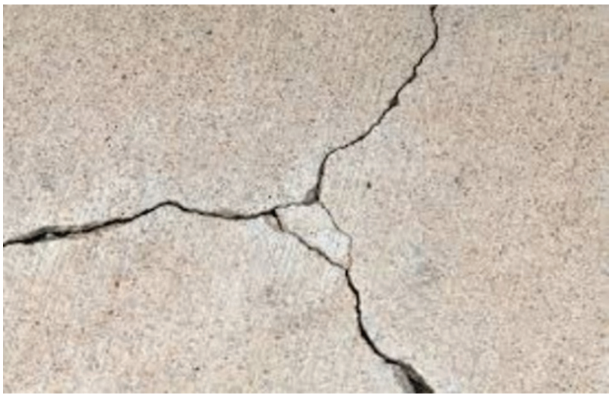


Fig. 27. Crack in concrete surface

The proposed model efficiently detects asphalt distress under various circumstances like high lighting intensity, shadows, and the existence of traffic signs. This efficiency of the proposed model is due to training relying on augmented datasets and the successes in covering most scene cases like rotating image, flipping and shear, providing a variety of scenes and angles of the same image. Color augmentation provides the system with high or low-intensity images by multiplying processes and contrast normalization. Additionally, Gaussian blur generates images with noise. (Fig. 28) shows samples of the augmented images for both concrete and asphalt cracked areas. The augmentation process resulted in building a generalized system responsive to a wide range of crack, pothole and rut scenes. In the real world, the system was tested on Iraqi streets.

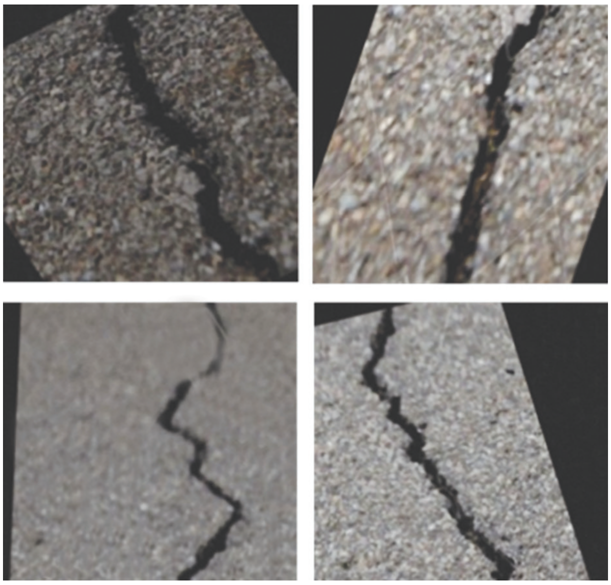


Fig. 28. samples of augmented images

The photos were captured using a web camera and sent directly to the system operated on a laptop. The system worked efficiently in the high-intensity light at noon and low-intensity light intensity near sunset. The time consumed was considerable, and the system took about 5 to 6 seconds to detect distress in the asphalt. The model is connected to a web cam of 1080p with 4k.

The execution time to detect the cracks increases gradually as the number of frames increases. (Fig. 29) represents the time of execution while the number of frames increased.

Additionally, the model was trained to detect special cases in Iraqi streets caused by insurgent drilling processes from people to establish water pipes through the streets, representing a crucial public problem in this country. (Fig. 30) presents one of the illegal drilling operations on Asphalt Street. We also compared the results of the proposed model with previous works that used the same benchmarked dataset adopted to evaluate this work. Table 2 compares those results with our proposed model's results.

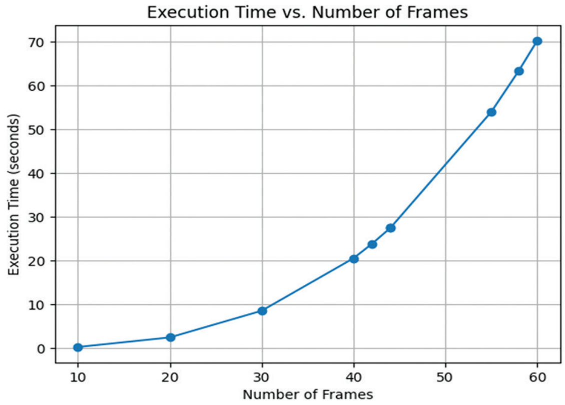


Fig. 29. proposed system execution time relate to the number of frames

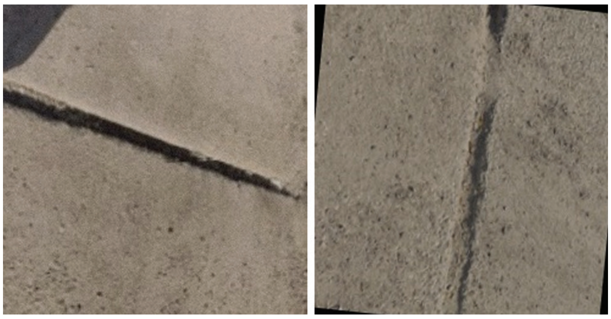


Fig. 30. cracks from illegal drilling from Iraqi streets

Zhang et al. registered 0.6 accuracy using the pavmentscapes dataset in their work. This low accuracy is caused by a wide variety of distress taken in their study model in addition to the noisy images that contain trees or other obstacles, such as environmental reasons like rain, snow, and sunlight, which may affect the quality of the image.

In contrast, our proposed model shows a performance accuracy of 1.00. The precision and recall in our model were higher than Zhang's work because we took three damage types to be predicted: cracks, potholes, and ruts. The second model used for comparison is Liu's Deep Crack model. Our model outperforms the Deep Crack model regarding precision, recall, and F-score metrics. Liu's model has about 0.87 precision, 0.85 recall, and 85.7 as the F-score, while our model got 1.00 for both precision and recall.

Table 2. compare the proposed model results with previous works

Paper	Dataset	Model	Accuracy	Precision	Recall	F-score
Zhang Tong	Pavementscapes	Segmentation transformer	0.60	0.4	0.73	8.42
Yahui Liu	Deep crack	Proposed model	1.00	0.99	0.99	0.99
Zhening Huang	NHA12D	Deep crack	-----	0.86	0.84	85.7
		Proposed model	1.00	1.00	1.00	1.00
		VGG16	----	0.35	0.90	0.5
		Proposed model	0.99	0.99	0.99	0.99
Shi et al.	Crack forest	Crack forest model	-----	0.82	0.89	95.68
		Proposed model	1.00	1.00	1.00	1.00

This difference in accuracy between the proposed model and the Deep Crack model is caused by a small number of non-cracked images in addition to the complexity of the surface that is coming from different asphalt textures and colors. The third work for comparison with the proposed system is for Huang et al. and the NHA12D dataset. Our model registered an average accuracy of 0.99 for the same dataset.In contrast, Huang used the straight VGG16 and achieved a precision of 0.35 and a recall of 0.90. The first reason behind the recall being higher than the precision in Huang's research is that the model is more biased toward the crack class than the non-crack class.

The second reason is that detecting concrete joints as cracks increases the false positive number, which leads to decreased precision. The Shi et al. model, called Crack Forest, achieved a precision of about 82.28 and a recall of 89.44, referring to the high number of false negative cases or high accuracy in detecting non-cracks in their model rather than cracks. Our model was implemented with the same dataset and achieved consistent precision and recall of 1.00, referring to balanced and efficient work predicting cracks and cracks cases. Furthermore, leveraging ImageNet pre-trained weights with the VGG16 model reduced the training time. In addition to this, VGG16 has efficient low-level feature crafting, helping the attention layer focus on high-level pattern understanding.

Our results are compared with a bench of baseline models such as (VGG16, ResNet, Unet, FCN, self-attention network, YOLOv8, YOLOv7, and RCNN). Table 3 presents a comparison of the main models used with the dataset used in this paper. These models were tested with the same datasets to test the proposed model. In [21], Liu et al. (2019) tried two models, VGG16 and ResNet. VGG16 achieved an accuracy of 0.30, while ResNet achieved an accuracy of 0.72 for the same dataset because ResNet has a residual connection that can collect deeper features without the gradient vanishing. Tong et al. [27] experimented with three models in their paper. The first model was UNET, which scored an accuracy of about 69.56.

Table 3. Compression table between a set of essential models

Model	Dataset	Accuracy	IOU
VGG16 [20]	Deep crack NHA12D	0.30 0.64	0.54
ResNet [20]	Deep crack	0.72	0.77
UNET [26]	Pavementscapes	69.56	54
FCN[27]	Pavementscapes	67	52
Self-attention network [27]	Pavementscapes	73.07	58.71
Yolov8 [43] Yolo v7 RCNN	RDD2022	78.4 57.8 49.4	
Texture feature extraction + Machine learning [28]	RDD2022	90.00	

The second model, the fully connected network FCN, was better, with 67% accuracy. The third model was the best, with the same dataset of Pavementscapes and an accuracy of 0.73. UNET shows low-performance returns due to noisy images in the Pavementscapes dataset containing shadows, traffic marks, etc. Therefore, FCN might work better than UNET in such cases. We noticed that a self-attention network was more effective because it can capture long-term dependencies. YOLOv8, YOLOv7, and RCNN were tested by Dong et al.(2024) [43] on the dataset, RDD2022. YOLOv8 outperforms YOLOv7 and RCNN with an accuracy of 78.4. YOLOv8 sometimes integrates label smoothing to regularize training and prevent overfitting.

In [29], their model combined a set of texture features such as GLCM, LBP, and HOG. Just the SVM, XGBOOST, and KNN classifiers gain an accuracy of over 0.90. The proposed model outperforms all mentioned models because the model incorporates VGG16 feature extracting with a multi-head attention layer that can understand long-range dependencies as one integrated feature set.

7. CONCLUSION

The infrastructure of roads and highways plays a vital role in the economy by connecting producers to markets and enabling more accessible transportation across regions and countries. Because of that, many countries are trying to enhance their transportation networks.

Detecting cracks and other damaged areas is essential because asphalt distress creates unseen surfaces that may increase the risk of accidents for drivers. Consequently, catching small cracks early is much cheaper than waiting for the damage to turn into major repairs.

As a result, the early detection of these issues allows repairs to be undertaken before damage worsens. Previous researchers have worked on designing CNNs to detect distress and damaged parts, while others have experimented with pre-trained models. However, their efforts have faced issues with accuracy because of an imbalanced dataset or the nature of the images.

The proposed system leverages the strength of VGG16 and the multi-head attention approach to focus on asphalt distress parts. VGG 16, a pre-trained CNN model on a massive dataset, extracts general features from images. Then, adding a multi-head attention layer makes the system focus on specific relationships between different parts of images. The proposed paradigm is beneficial for asphalt distress detection where the spatial context or how the cracks or potholes are internally connected is essential for accurate classification. For future work, a large, diverse dataset encompassing various climates and pavement types is needed to enhance model generalization. The models also require a lightweight architecture that enables real-time deployment on mobile devices for on-the-go road inspections. Last, the researchers must work on an explainable model that produces reasoning behind the classification or decision.

8. REFERENCES

[1] F. R. Bruinsma, S. A. Rienstra, P. Rietveld, "Economic Impacts of the Construction of a Transport Corridor: A Multi-level and Multiapproach Case Study for the Construction of the A1 Highway in the Netherlands", Regional Studies, Vol. 31, No. 4, 1997, pp. 391-402.

[2] J.-A. R. Sarmiento, "Pavement Distress Detection and Segmentation using YOLOv4 and DeepLabv3 on Pavements in the Philippines", arXiv:2103.06467, 2021.

[3] V. Mandal, A. R. Mussah, Y. Adu-Gyamfi, "Deep Learning Frameworks for Pavement Distress Classification: A Comparative Analysis", Proceedings of the IEEE International Conference on Big Data, Atlanta, GA, USA, 10-13 December 2020, pp. 5577-5583.

[4] L. Jia, S. Yang, W. Wang, X. Zhang, "Impact analysis of highways in China under future extreme precipitation", Natural Hazards, Vol. 110, No. 2, 2022, pp. 1097-1113.

[5] J. Zhu, J. Zhong, T. Ma, X. Huang, W. Zhang, Y. Zhou, "Pavement distress detection using convolutional neural networks with images captured via UAV", Automation in Construction, Vol. 133, 2022, p. 103991.

[6] N. G. Sorum, T. Guite, N. Martina, "Pavement Distress: A Case Study", International Journal of Innovative Research in Science, Engineering and Technology, Vol. 3, 2014, pp. 274-284.

[7] X. Chen, S. Yongchareon, M. Knoche, "A review on computer vision and machine learning techniques for automated road surface defect and distress detection", Journal of Smart Cities and Society, Vol. 1, No. 4, 2022, pp. 259-275.

[8] W. Tang, S. Huang, X. Zhang, L. Huangfu, "PicT: A Slim Weakly Supervised Vision Transformer for Pavement Distress Classification", Proceedings of the 30th ACM International Conference on Multimedia, Lisboa, Portugal, 10-14 October 2022, pp. 3076-3084.

[9] V. Pereira, S. Tamura, S. Hayamizu, H. Fukai, "Semantic Segmentation of Paved Road and Pothole Image Using U-Net Architecture", Proceedings of the International Conference of Advanced Informatics: Concepts, Theory and Applications, Yogyakarta, Indonesia, 20-21 September 2019, pp. 1-4.

[10] A. Ragnoli, M. R. De Blasiis, A. Di Benedetto, "Pavement Distress Detection Methods: A Review", Infrastructures, Vol. 3, No. 4, 2018, p. 58.

[11] K. A. Vinodhini, K. R. A. Sidhaarth, "Pothole detection in bituminous road using CNN with transfer learning", Measurement: Sensors, Vol. 31, 2024, p. 100940.

[12] A. Apeagyei, T. E. Ademolake, M. Adom-Asamoah, "Evaluation of deep learning models for classification of asphalt pavement distresses", International Journal of Pavement Engineering, Vol. 24, No. 1, 2023, p. 2180641.

[13] H. S. Sharaf Al-deen, Z. Zeng, R. Al-sabri, A. Hekmat, "An Improved Model for Analyzing Textual Sentiment Based on a Deep Neural Network Using Multi-Head Attention Mechanism", Applied System Innovation, Vol. 4, No. 4, 2021, p. 85.

[14] F. Xue, Q. Wang, G. Guo, "TransFER: Learning Relation-aware Facial Expression Representations with Transformers", Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada,11-17 October 2021, pp. 3581-3590.

[15] A. Zhou et al. "Multi-head attention-based two-stream EfficientNet for action recognition", Multimedia Systems, Vol. 29, No. 2, 2023, pp. 487-498.

[16] G. Hong, X. Chen, J. Chen, M. Zhang, Y. Ren, X. Zhang, "A multi-scale gated multi-head attention depthwise separable CNN model for recognizing COVID-19", Scientific Reports, Vol. 11, No. 1, 2021, p. 18048.

[17] R. Shahabian, A. M. Moghaddam, S. A. Sahaf, H. reza Pourreza, "Second-Order Statistical Texture Representation of Asphalt Pavement Distress Images Based on Local Binary Pattern in Spatial and Wavelet Domain", Rehabilitation in Civil, Vol. 7, No. 3, 2019, pp. 48-67.

[18] M. Salman, S. Mathavan, K. Kamal, M. Rahman, "Pavement crack detection using the Gabor filter", Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems, Hague, Netherlands, 6-9 October 2013, pp. 2039-2044.

[19] A. Cubero-Fernandez, F. J. Rodriguez-Lozano, R. Villatoro, J. Olivares, J. M. Palomares, "Efficient pavement crack detection and classification", EURASIP Journal on Image and Video Processing, No. 2017, 2017, pp. 1-11.

[20] K. Gopalakrishnan, S. K. Khaitan, A. Choudhary, A. Agrawal, "Deep Convolutional Neural Networks with transfer learning for computer vision-based data-

driven pavement distress detection", Construction and Building Materials, Vol. 157, 2017, pp. 322-330.

[21] Y. Liu, J. Yao, X. Lu, R. Xie, L. Li, "DeepCrack: A deep hierarchical feature learning architecture for crack segmentation", Neurocomputing, Vol. 338, 2019, pp. 139-153.

[22] Z. Fan et al. "Ensemble of Deep Convolutional Neural Networks for Automatic Pavement Crack Detection and Measurement", Coatings, Vol. 10, No. 2, 2020, p. 152.

[23] Y. Li, P. Che, C. Liu, D. Wu, Y. Du, "Cross-scene pavement distress detection by a novel transfer learning framework", Computer-Aided Civil and Infrastructure Engineering, Vol. 36, No. 11, 2021, pp. 1398-1415.

[24] R. Ghosh, O. Smadi, "Automated Detection and Classification of Pavement Distresses using 3D Pavement Surface Images and Deep Learning", Transportation Research Record, Vol. 2675, No. 9, 2021, p. 1359.

[25] H. Abbas, M. Q. Ismael, "Automated Pavement Distress Detection Using Image Processing Techniques", Engineering, Technology & Applied Science Research, Vol. 11, No. 5, 2021, pp. 7702-7708.

[26] Y. Chen, X. Gu, Z. Liu, J. Liang, "A Fast Inference Vision Transformer for Automatic Pavement Image Classification and Its Visual Interpretation Method", Remote Sensing, Vol. 14, No. 8, 2022, p. 1877.

[27] Z. Tong, T. Ma, J. Huyan, W. Zhang, "Pavementscapes: a large-scale hierarchical image dataset for asphalt pavement damage segmentation", arXiv:2208.00775, 2022.

[28] Z. Huang, W. Chen, A. Al-Tabbaa, I. Brilakis, "NHA12D: A New Pavement Crack Dataset and a Comparison Study Of Crack Detection Algorithms", arXiv:2205.01198, 2022.

[29] E. Eslami, H.-B. Yun, "Comparison of deep convolutional neural network classifiers and the effect of scale encoding for automated pavement assessment", Journal of Traffic and Transportation Engineering, Vol. 10, No. 2, 2023, pp. 258-275.

[30] Y Y. Shi, L. Cui, Z. Qi, F. Meng, Z. Chen, "Automatic Road Crack Detection Using Random Structured Forests", IEEE Transactions on Intelligent Transportation Systems, Vol. 17, No. 12, 2016, pp. 3434-3445.

[31] S. Cano-Ortiz, L. L. Iglesias, P. M. R. del Árbol, P. Lastra-González, D. Castro-Fresno, "An end-to-end computer vision system based on deep learning for pavement distress detection and quantification", Construction and Building Materials, Vol. 416, 2024, p. 135036.

[32] A. Nasertork, S. Ranjbar, M. Rahai, F. M. Nejad, "Pavement raveling inspection using a new image texture-based feature set and artificial intelligence", Advanced Engineering Informatics, Vol. 62, 2024, p. 102665.

[33] M. Guerrieri, G. Parla, M. Khanmohamadi, L. Neduzha, "Asphalt Pavement Damage Detection through Deep Learning Technique and Cost-Effective Equipment: A Case Study in Urban Roads Crossed by Tramway Lines", Infrastructures, Vol. 9, No. 2, 2024.

[34] K. Ijari, C. D. Paternina-Arboleda, "Sustainable Pavement Management: Harnessing Advanced Machine Learning for Enhanced Road Maintenance", Applied Sciences, Vol. 14, No. 15, 2024, p. 6640.

[35] R. C. Gonzalez, R. E. Woods, "Digital Image Processing", 3th Edition, Prentice-Hall, 2006.

[36] R. G. Gavaskar, K. N. Chaudhury, "Fast Adaptive Bilateral Filtering", IEEE Transactions on Image Processing, Vol. 28, No. 2, 2019, pp. 779-790.

[37] Q. He, Z. Xiang, P. Ren, "A CLSTM and transfer learning based CFDAMA strategy in satellite communication networks", Plos One, Vol. 16, No. 3, 2021, p. e0248271.

[38] M.-H. Guo et al. "Attention Mechanisms in Computer Vision: A Survey", Computational visual media, Vol. 8, No. 3, 2022, pp. 331-368.

[39] J. Gupta, S. Pathak, G. Kumar, "Deep Learning (CNN) and Transfer Learning: A Review", Proceedings of the International Conference on Applications of Intelligent Computing in Engineering and Science, Raipur, India, 12-13 Febreuary 2022, p. 012029.

[40] P. Ramachandran, N. Parmar, A. Vaswani, I. Bello, A. Levskaya, J. Shlens, "Stand-Alone Self-Attention in Vision Models", Advances in Neural Information Processing Systems, Vol. 32, 2019.

[41] G. Hong, X. Chen, J. Chen, M. Zhang, Y. Ren, X. Zhang, "A multi-scale gated multi-head attention depthwise separable CNN model for recognizing COVID-19", Scientific Reports, Vol. 11, No. 1, 2021, p. 18048.

[42] C. Shorten, T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning", Journal of Big Data, Vol. 6, No. 1, 2019, p. 60.

[43] X. Dong, Y. Liu, J. Dai, "Concrete Surface Crack Detection Algorithm Based on Improved YOLOv8", Sensors, Vol. 24, No. 16, 2024, p. 5252.