2022 25th International Conference on Computer and Information Technology (ICCIT)
17-19 December, Cox's Bazar, Bangladesh

InvoPotNet: Detecting Pothole from Images through Leveraging Lightweight Involutional Neural Network

Joyanta Jyoti Mondal*
School of Data and Sciences
Brac University
Dhaka, Bangladesh
joyanta.jyoti.mondal@g.bracu.ac.bd

Md. Farhadul Islam*
School of Data and Sciences
Brac University
Dhaka, Bangladesh
md.farhadul.islam@g.bracu.ac.bd

Sarah Zabeen
School of Data and Sciences
Brac University
Dhaka, Bangladesh
sarah.zabeen@g.bracu.ac.bd

Meem Arafat Manab
School of Data and Sciences
Brac University
Dhaka, Bangladesh
meem.arafat@bracu.ac.bd

***Abstract*—Potholes can be liable to endangering people's safety on the road through road accidents, thereby bringing down the road's functionality. In this research, we present a road pothole detection system, InvoPotNet, that uses Involution Neural Network (INN) approach to automatically identify potholes on the road which is 25 times smaller than the Deep CNN model. Five models; InceptionV3, ResNet50, VGG19, MobileNetV2, and Custom Deep Convolutional Neural Network (Deep CNN) are trained and assessed with the help of a preprocessed dataset. Initially, we collect a public dataset where pothole and non-pothole pictures are gathered and categorized. The following step involves the training and evaluation of the four models for comparison of metrics such as accuracy and loss, using the processed picture dataset. Then the performance and accuracy of these four models are evaluated. The experimental findings demonstrate that InvoPotNet and the Convolutional Neural Network (CNN) model yield very similar and the most accurate detection results. Our approach shows an 86.29% accuracy with a significantly less number of parameters, unlike other popular models.**

***Index Terms*—Pothole Detection, Involutional Neural Network, Convolutional Neural Network, Deep Learning**

I. Introduction

The recent advances in the approaches and methods of artificial intelligence provide the groundwork for the development and design of autonomous vehicle systems (AVS) in the automotive sector. Nevertheless, the characteristics of roads in the actual world demand a high degree of human safety assurance, which continues to be a difficult feat for these companies. The World Health Organization (WHO) reported a substantial death rate due to road accidents [1]. Although the reasons for these fatalities are random [2], they are often the consequence of reckless driving and an inability to grasp the driving environment. Deep Learning (DL) and Computer Vision (CV) are two methods that are being utilized to examine driving scenes [3]–[5]. These methods essentially consist of feature extraction, classification, detection, and tracking. DL and CV methodologies have been helpful in improving a variety of essential components of AVS, such as the identification of a lane, road, vehicle, or pedestrian, amongst other improvements.

A road surface that has a structural fault is known as a pothole. It is not something that can be ignored since it has the potential to result in catastrophic traffic accidents and degrade the efficiency of roadways. The Asian Development Bank (ADB) conduct an inspection in 2006 and finds that more than half of these paved roads are in deplorable condition. A problem of this kind exists in almost every country. Extreme weather and heavy traffic are the two primary contributors to the formation of potholes. If you are unable to identify potholes in the road, you might endanger the safety of your passengers as well as the mechanical components of your car. In their study, Aparna et al. [6] employ a thermal camera at night in order to assess the various attributes of potholes, but the performance of an AVS during the day is restricted and expensive. Nienaber et al. use an optical camera to pinpoint potholes through the removal of non-road regions as well as concentrating on the segments of the road that include potholes [7]. They do this by deleting non-road areas first. In their study [8], Ryu et al. adopt a methodology that consists of three significant steps to the identification of potholes. During the process of segmentation, shape-based histogram thresholding is put to use in order to turn the picture that is supplied into a binary image. After that, an enhanced histogram shape-based thresholding algorithm, also known as HST, is used in order to differentiate the expanded pothole candidate region. In the last stage, the ordered histogram intersection (OHI) method is used

*These authors contributed equally to this work.

979-8-3503-4602-2/22/$31.00 ©2022 IEEE

to find potholes by determining the degree of similarity that exists across different areas. In a similar manner, the research presented in [9] examines the structure of the road surface by utilizing the accelerometers found in smartphones in order to evaluate the stability of a moving car while it is on the road. The motion of a car as it goes over potholes depicts fluctuations in the motion of the vehicle until it reaches a point where it is stable on a smooth road. In order to differentiate the various road objects according to the stability patterns they exhibit, these sequences of events are recorded and modeled as a multivariate time series.

In this scenario, DL is implemented as a long short-term memory (LSTM), a Convolutional Neural Network (CNN), and a reservoir computing (RC) system to differentiate potholes from other objects on the road. In order to limit the amount of processing that is required, regions of interest (ROIs) are established and trained to evaluate the selected frames using a deep CNN model. The fourth iteration of Inception Potholes can be located using ResNetV2 and MobileNetV1, respectively. In addition, LS-SVM and ANN (Artificial Neural Network) are implemented to detect potholes, where both methods yield a rate of around 89% classification accuracy [10]. The performance of the currently available methods for pothole detection is restricted, but it is possible to enhance these methods by developing a software and hardware architecture that is both easier to use and more cost-effective. Additionally, the behavior of AVS (Autonomous Vehicle System) with regard to intelligent decision-making has not yet been researched. Also, in the last few years, researchers are focusing on decreasing the reliance upon more computational power to make the model more usable in mass level [11]. So that, their models can be more available regardless of the processing power of a computer, and can be easily implemented in low-cost online servers.

Kumar et al. [12] proposes a fine-tuned model named "F-RCNN Inception v2" to solve this problem. Ping et al. [13] also examine to solve this problem in four different models which are YOLOv3, SSD (Single Shot Detector), HOG (Histogram of Oriented Gradients) with SVM (Support Vector Machine) and Faster R-CNN and they report YOLO model to perform the best with an accuracy of 82%.

According to our study, we are able to make the following set of contributions to this paper:

- We present a deep learning based approach using Involutional Neural Network (INN) algorithm which takes a very low amount of computational power.
- We evaluate existing transfer learning based deep learning approaches to compare their efficiency rate along with their parameters (computational power) with our proposed approach.

The rest of the study is organized as follows: We set out with illustrating the approach we take to solve this problem in Section II. Then, in Section III, we show experiments taken for the issue and evaluate the experimental results. Afterwards, the evaluation of our investigation is discussed in Section IV. Lastly, in Section V, we explore and demonstrate possible future applications.

II. Research Methodology

In this part, we present an INN architecture for detecting road potholes. Initially, we create a variation of the standard INN framework by altering architectural design and functions. The main reason of using INN model is, it captures area or spatial information efficiently. A potholes can be considered as a segment of an image. With the power of INN, we can classify the images with very low computational cost.

A. Involutional Neural Network (INN)

Convolution is the foundation of the majority of contemporary neural networks for computer vision. A convolution kernel is channel-specific and spatially neutral. This prevents it from adapting to diverse visual patterns in relation to distinct spatial places. In addition to location-related issues, the receptive area of convolution makes it difficult to capture protracted spatial connections.

Li et al. [14] reconsider the features of convolution in order to overcome the aforementioned problems. The authors propose the location-specific and channel-independent "Involution kernel." Due to the operation's location-specific character, the authors assert that self-attention comes under the Involution design paradigm.

To infer the concept of Involutions properly, we have to look at the process of convolution. Consider a tensor X with dimensions H, W, and C_{in} as an input. We take a collection of C_{out} convolution kernels with K, K, C_{in} shapes. With the multiply-add operation between the input tensor and the kernels, the output tensor Y possesses the dimensions H, W, and C_{out}.

In Figure 1, C_{out} = 3 which produces an output tensor with the shapes H, W, and 3. The convolution kernel is independent of the spatial position of the input tensor, rendering it location-independent. Alternatively, each channel in the output tensor is based on a distinct convolution filter, making it channel-specific.

The goal is to establish an operation that is both channel-agnostic and location-specific. It is difficult to implement these precise qualities. With a fixed number of Involution kernels (for each spatial point), variable-resolution input tensors cannot be processed.

The authors have proposed constructing each kernel based on specified spatial positions in order to resolve this issue. This technique should facilitate the processing of variable-resolution input tensors. Figure 2 illustrates this method of kernel creation. Here, $K \times K \times C$ filters are generated, where C is the number of channel groups. Instead of employing a single filter and broadcasting it overall C input channels, we generate C filters and broadcast them into each of the C input channels.

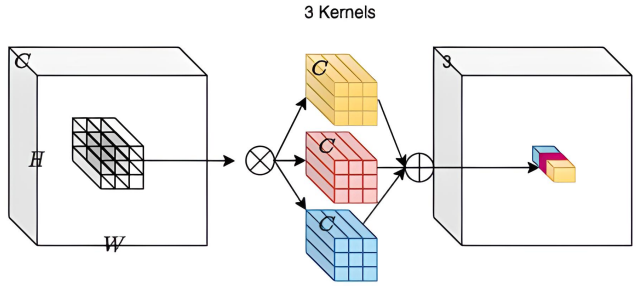

Fig. 1. Convolution Process

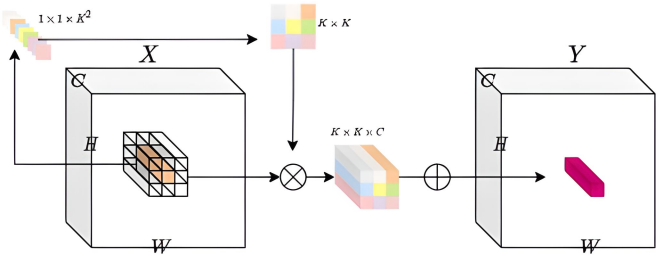

Fig. 2. Involution Process

B. Proposed Methodology

The key purpose of this study is to have as few parameters as feasible by making the input shape as small as possible while still producing a desirable outcome. An architecture with fewer parameters achieves greater precision and computing speed. Our INN model begins with a 75×75 pixel image with three channels extracted from the dataset. Then, we employ three Involution layers with kernel sizes of (3,3). This is because we want to benefit from weight sharing and reduction in computational costs. In addition, we use MaxPooling layers with a pool size of (2,2) to reduce the computational expense of the layers. Furthermore, all Involution layers have default strides of 1.

As opposed to other activation functions like the Tanh/Sigmoid function, we use ReLU as the activation function because its gradient is not saturated, which significantly speeds up the development of stochastic gradient descent (SGD).

The next step is to turn the values into a one-dimensional array, after which the INN's initial 64 nodes will be replaced with fully connected (FC) layers. We use Adam as the optimizer of the proposed model compilation, and it has a learning rate of 0.001, so it helps us get the most out of our production processes. The maximum number of epochs the model can run for is 30, and the maximum batch size is 32. With fewer parameters and hence fewer computational requirements, this architecture delivers optimal performance.

III. Experimental Evaluation

After the creation of our proposed architecture, we put our model through its tests utilizing metrics that are routinely employed in classification-based tasks. Our number one priority is to develop a model that, regardless of the values of its parameters or other hyperparameters, always produces the highest quality output.

TABLE I
Output Shape and Parameter Size of Each Layer of The Proposed Model

Layers	Output Shape	Parameters
Involutional Layer	(None, 75, 75, 3)	26
2D Max Pooling	(None, 37, 37, 3)	0
Involutional Layer	(None, 37, 37, 3)	26
2D Max Pooling	(None, 18, 18, 3)	0
Involutional Layer	(None, 18, 18, 3)	26
Flatten	(None, 972)	0
Dense	(None, 64)	62,272
Dense	(None, 10)	65
Total parameters		62,415
Trainable parameters		62,409
Non-trainable parameters		6

TABLE II
Hyperparameter of Our Proposed Model

Hyperparameter				
Image Input Size	Epoch	Batch Size	Learning Rate	Parameters
75 x 75	30	32	0.001	62,415

A. Experimental Setup

Tensorflow, Keras, Pillow, and OpenCV Python libraries are used to make the training and testing protocols for this INN model, as well as for all other transfer learning models. The models are trained and evaluated on two different devices; one with an NVIDIA RTX 2070 with 7.5 TeraFLOPs of performance and another with an NVIDIA RTX 3080TI GPU which has 34.1 TeraFLOPs of performance.

B. Dataset

The Electrical and Electronic Department at Stellenbosch University developed the dataset that we are working on, in 2015 [7]. The full dataset is divided into two distinct sets, of which one was regarded to be easy to understand while the other was more difficult. They gathered the dataset by placing a person's smartphone on the dashboard of a moving vehicle and having them take images on the device. The files that make up these datasets do, in fact, share parts of one another, and there are a few cases in which the names of two distinct photographs are identical. Taking this into consideration, necessary precautions need to be taken before the data is integrated into a single more comprehensive dataset. Each folder has two subfolders that hold the data for both the training and the test, respectively. In addition, the photographs of roads with potholes and roads without potholes are separated into two additional subfolders inside the training data folder. These subfolders are labeled "positive data" and "negative data," respectively.

The collection of dataset we intend to work on is Dataset 1 (or Simplex) where the images are of mainly roads that are normal and those with potholes. The dataset is partitioned into

Fig. 3. Normal Road (Left) and Road with Pothole (Right) from The Dataset

training, validation, and test datasets, each having a different number of images: 3836, 426, and 474 respectively.

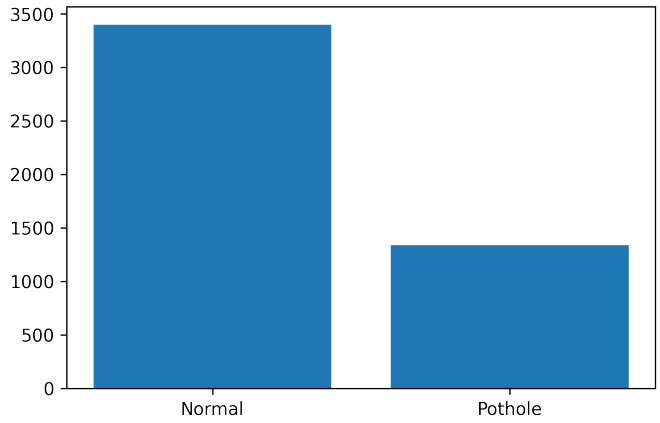

Fig. 4. Number of Images in Dataset

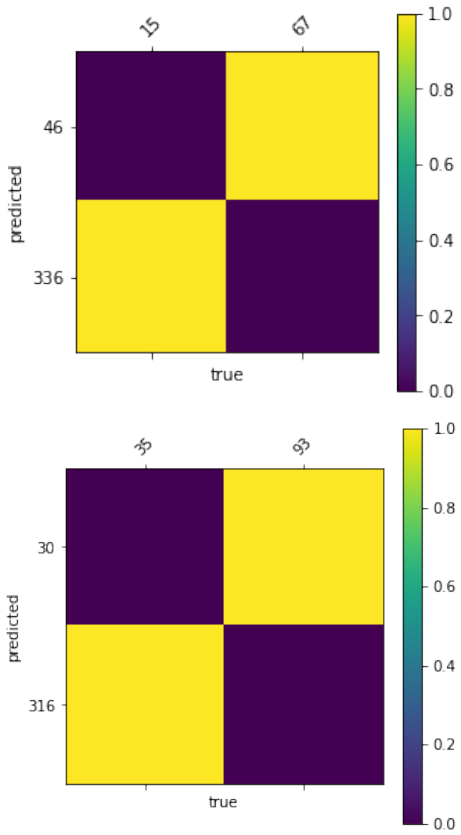

Fig. 5. Confusion Matrix of CNN (Up) and INN (Below)

1) Data Preprocessing: In most cases, data augmentation is helpful in situations in which the context of categorization between two classes is highly vivid and evident (for example, classifying a cat from a dog). When this occurs, the inclusion of noise will not have a negative impact on the greater context of the pictures that are used for categorization. The size of each picture is first shrunk down to 75 pixels along its longest dimension so that we can accelerate the learning process. It is done mainly due to the fact that the processing of images of real sizes (3680×2760 pixels) can be highly difficult and time-consuming. Moreover, the pixel reduction also reduces unnecessary information of objects which are not important for the task. Here, we require spatial area of the pothole, which helps the INN model to perform better and pixel reduction does not affect the performance. After that, we give each of the photos a flip to the horizontal position. It refers to turning all of the rows and columns of pixels in a picture horizontally in the opposite direction. After that, we do a Random Shearing. This indicates that the picture will be deformed along an axis, the primary purpose of which is to either produce or correct the perception angles. In most cases, it is used to enhance pictures so that computers can view objects from diverse perspectives in the same way that people do. And last but not least, we make use of the Random Zoom function, which arbitrarily magnifies the picture by either zooming in or adding additional pixels around the image in order to make it larger. The zoom ranged from 0 to 20% of its original size.

C. Experimental Findings

The accuracy of a model can be conceived of as the fraction of true predictions made by the model relative to the total number of predictions.

In comparison to other methodologies, our model delivers highly promising testing accuracy and other metrics. Table III details the comparison. From Table III, we can see that the majority of methods require a significant number of parameters, whereas our model requires fewer parameters. From Fig. 5, we can see that INN is more balanced in terms of predicting both classes. CNN is more biased towards predicitng the ”Nomral” class.

From Fig. 7, we see the proposed INN model has a better fit than the custom CNN model which has comparatively good results. But the INN predictions are much more stable. Even though the proposed INN could not get the highest accuracy but the difference is only 0.63, which is negligible since the INN model is more stable and fits the model better than others.

IV. Discussion

In this task, Involution can summarize the situation in a more extensive spatial configuration which leads to a better performance than convolution-based models. The lightweight architecture and the relation to the attention-based networks

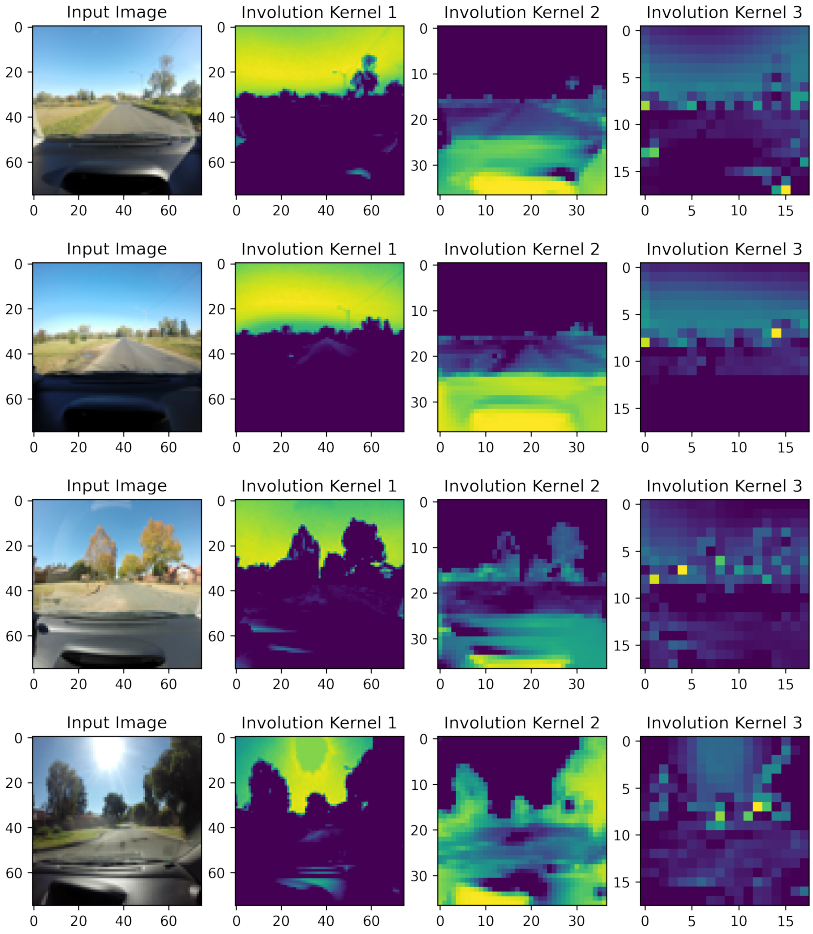

Fig. 6. Visualization of Involutional Kernel

TABLE III
COMPARISON AFTER TRAINING IN DIFFERENT MODELS

Models	Parameters (in Million)	Loss	Accuracy	Recall	Precision	F1 Score
InceptionV3	21.8	0.52	71.73%	72.5%	72.5%	72.5%
ResNet50	23.6	0.56	74.05%	78.33%	79.17%	78.74%
VGG19	20	0.55	74.05%	78%	79.5%	78.74%
MobileNetV2	2.2	0.37	85.44%	87.44%	90.44%	88.91%
Ping et al. [13]	-	-	82%	-	-	-
Custom Deep CNN	**1.2**	**0.35**	**86.85%**	**95.73%**	**87.96%**	**91.68%**
Ours	**0.06**	**0.326**	**86.29%**	**90.03%**	**91.33%**	**90.67%**

help to classify the images more efficiently than convolution-based models. The compact nature of the model, in conjunction with its high level of efficiency, has the potential to pave the way for cutting-edge real-world applications in the areas of autonomous driving and safety measures.

V. CONCLUSION

In this research, we offer an effective pothole detection system that makes use of of a INN based algorithm. There are a total of four models that are trained and evaluated using the preprocessed dataset which are InceptionV3, ResNet50, VGG19, and MobileNetV2. For more precise detection results, the hyper parameters of all four models have been modified, and the size of potholes has been taken into consideration. When the results of all four models were compared, it was determined that our proposed model performed the best, with a testing accuracy of 86%. The work that will be done in the future will include things like expanding the detecting item to include damaged drains and manhole covers and making use of photographs captured from moving vehicles in a realistic

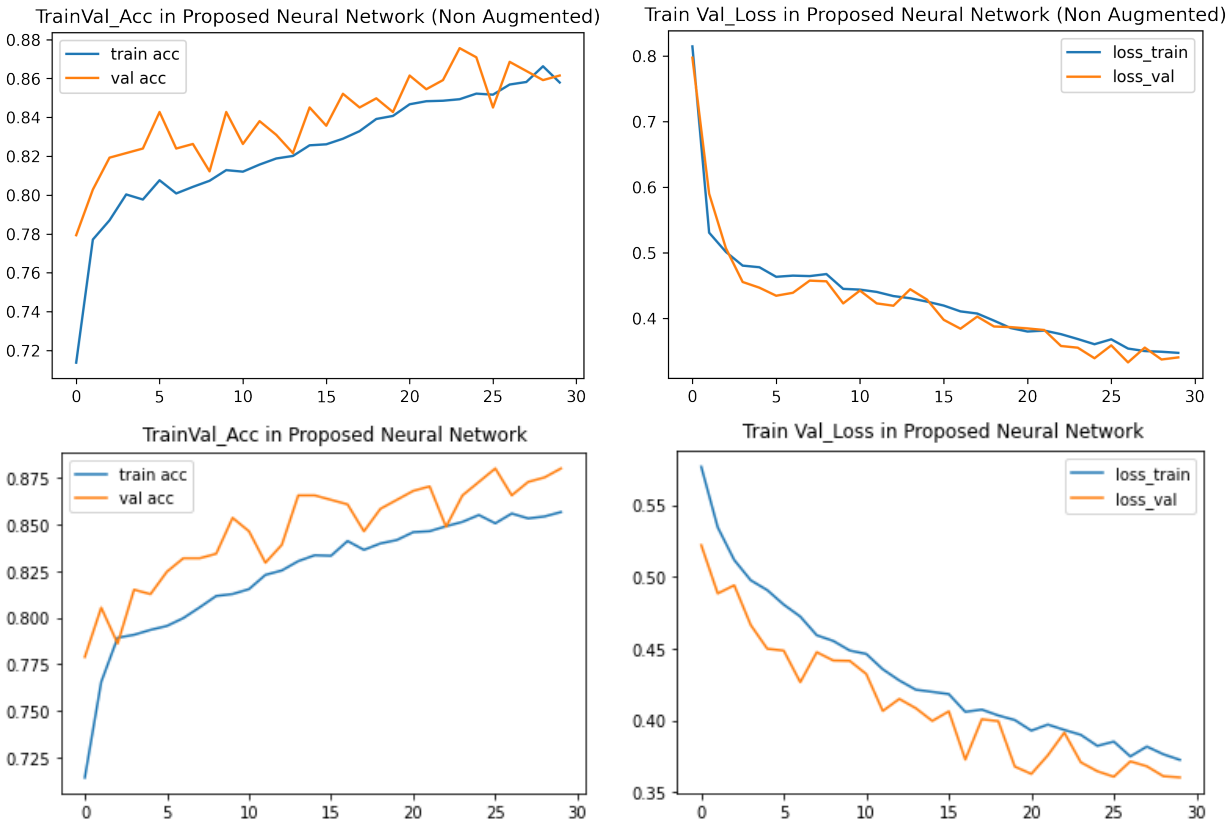

Fig. 7. Accuracy and Loss Curves of INN and CNN

setting. We can also grade the pothole size using INNs, which are good for segmentation tasks. So, INNs may be good approach for the task.

REFERENCES

[1] World Health Organization (WHO), "Road traffic deaths, global health observatory data," https://www.who.int/data/gho/data/themes/road-safety, accessed: 2022-8-27.

[2] D. K. Dewangan and S. P. Sahu, "Real time object tracking for intelligent vehicle," in *2020 First International Conference on Power, Control and Computing Technologies (ICPC2T)*. IEEE, 2020.

[3] Z. Han, J. Liang, and J. Li, "Design of intelligent road recognition and warning system for vehicles based on binocular vision," *IEEE Access*, vol. 6, pp. 62 880–62 889, 2018.

[4] L. Huang, T. Zhe, J. Wu, Q. Wu, C. Pei, and D. Chen, "Robust inter-vehicle distance estimation method based on monocular vision," *IEEE Access*, vol. 7, pp. 46 059–46 070, 2019.

[5] M. Junaid, M. Ghafoor, A. Hassan, S. Khalid, S. A. Tariq, G. Ahmed, and T. Zia, "Multi-feature view-based shallow convolutional neural network for road segmentation," *IEEE Access*, vol. 8, pp. 36 612–36 623, 2020.

[6] Aparna, Y. Bhatia, R. Rai, V. Gupta, N. Aggarwal, and A. Akula, "Convolutional neural networks based potholes detection using thermal imaging," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 3, pp. 578–588, 2022.

[7] S. Nienaber, R. S. Kroon, and M. J. Booysen, "A comparison of low-cost monocular vision techniques for pothole distance estimation," in *2015 IEEE Symposium Series on Computational Intelligence*. IEEE, 2015.

[8] S.-K. Ryu, T. Kim, and Y.-R. Kim, "Image-based pothole detection system for ITS service and road management system," *Math. Probl. Eng.*, vol. 2015, pp. 1–10, 2015.

[9] B. Varona, A. Monteserin, and A. Teyseyre, "A deep learning approach to automatic road surface monitoring and pothole detection," *Pers. Ubiquitous Comput.*, vol. 24, no. 4, pp. 519–534, 2020.

[10] N.-D. Hoang, "An artificial intelligence method for asphalt pavement pothole detection using least squares support vector machine and neural network with steerable filter-based feature extraction," *Adv. Civ. Eng.*, vol. 2018, pp. 1–12, 2018.

[11] J. J. Mondal, M. F. Islam, S. Zabeen, A. B. M. A. A. Islam, and J. Noor, "Note: Plant leaf disease network (PLeaD-net): Identifying plant leaf diseases through leveraging limited-resource deep convolutional neural network," in *ACM SIGCAS/SIGCHI Conference on Computing and Sustainable Societies (COMPASS)*. New York, NY, USA: ACM, 2022.

[12] A. Kumar, Chakrapani, D. J. Kalita, and V. P. Singh, "A modern pothole detection technique using deep learning," in *2nd International Conference on Data, Engineering and Applications (IDEA)*. IEEE, 2020, pp. 1–5.

[13] P. Ping, X. Yang, and Z. Gao, "A deep learning approach for street pothole detection," in *2020 IEEE Sixth International Conference on Big Data Computing Service and Applications (BigDataService)*. IEEE, 2020, pp. 198–204.

[14] D. Li, J. Hu, C. Wang, X. Li, Q. She, L. Zhu, T. Zhang, and Q. Chen, "Involution: Inverting the inherence of convolution for visual recognition," 2021. [Online]. Available: https://arxiv.org/abs/2103.06255