Journal of Harbin Engineering University
ISSN: 1006-7043
Vol 44 No. 7
July 2023

DETECTION OF ROAD SURFACE DAMAGE USING CUSTOM OBJECTS THROUGH SINGLE SHOT DETECTION

R. PALANI
Research Scholar, Department of Computer and Information Science,
Annamalai University, Chidambaram, Tamil Nadu, India

Dr. N. PUVIARASAN
Professor and Head, Department of Computer and Information Science,
Annamalai University, Chidambaram, Tamil Nadu, India

Dr. A. RAMA PRASATH
Associate Professor, School of Computing Sciences,
Hindustan University, Chennai, Tamil Nadu, India

Abstract: The aim of the research is to apply Single Shot Detection methods to identify road damage in custom objects. Despite the existence of various established techniques, the researchers are determined to continue using them because newer and improved approaches have become available. Additionally, it is important to delve into the underlying causes of road damage and introduce fresh ideas and reasoning into the detection process. The research paper encompasses an exploration of techniques for data preparation, annotation of data labels, and the creation of a road damage dataset for training models.

Keywords: Single Shot Detection, Road Damage Identification, custom object detection, Data annotation, road maintenance, SSD - MobileNet, SSD - ResNet

1. Introduction

Roads are vital infrastructural components that enable transportation and connectivity in any country. In India, a rapidly developing nation with a vast road network, maintaining road quality is crucial for ensuring efficient and safe movement of people and goods. However, the wear and tear experienced by roads due to factors like heavy traffic, adverse weather conditions, and inadequate maintenance can lead to road damage, posing significant challenges to transportation systems.

Road damage detection plays a pivotal role in identifying and assessing the deterioration of roads. Traditional methods of road damage detection rely on visual inspections carried out by human experts, which are time-consuming, costly, and subjective. The advent of advanced technologies, such as artificial intelligence (AI) and computer vision, has revolutionized the field of road damage detection by providing automated and accurate solutions.

In recent years, India has witnessed a growing interest in leveraging AI-based techniques for road damage detection. Researchers, engineers, and government agencies are exploring innovative approaches to monitor and assess road conditions, aiming to improve maintenance strategies, enhance road safety, and optimize resource allocation. The application of AI in road damage detection has the potential to transform the way roads are monitored and managed in India, leading to more efficient and sustainable transportation networks.

This article aims to provide an overview of road damage detection in India, focusing on the advancements in AI-based techniques and their implications. By examining the current state of road infrastructure, discussing the challenges faced, and highlighting notable initiatives, this

article aims to shed light on the potential benefits and opportunities that arise from adopting AI-driven approaches in road damage detection. Furthermore, this article will explore the impact of such advancements on road maintenance practices, cost-effectiveness, and overall road network performance.

Through a comprehensive analysis of the existing literature, case studies, and practical implementations, this article aims to provide insights into the current state-of-the-art methods, technologies, and frameworks employed for road damage detection in India. By understanding the strengths and limitations of these approaches, stakeholders can make informed decisions regarding the implementation of AI-driven solutions, thereby paving the way for more sustainable and resilient road networks in the country.

1.2 Research aim

The aim of this research is to apply single shot detection (SSD) techniques for the purpose of road damage detection. This approach is renowned for its simplicity, high accuracy, and overall performance. In the realm of object detection, there are numerous techniques currently available, and researchers are actively exploring them to keep up with evolving technological advancements. Additionally, it is important to note that environmental disturbances can impact the underlying causes of road damage, which necessitates continuous research and adaptation.

Furthermore, it is worth mentioning that the detection application is currently facing challenges due to compiler version changes. This ongoing application development is driven by multiple factors, including cost-effectiveness, accuracy, and ease of handling. Additionally, the application's compatibility and integration with other applications are also important considerations.

1.3 Custom Object Detection

Custom object detection involves training a machine learning model to detect specific objects in images or videos. This process requires creating a dataset with labelled images or videos containing the objects of interest, followed by training the model to recognize and locate these objects in new data. Custom object detection has various applications, including self-driving cars, security cameras, retail stores, and medical imaging. However, it also presents challenges, such as the time-consuming and expensive nature of data collection to gather a sufficiently large and labelled dataset. Overcoming these challenges is crucial for developing accurate and reliable custom object detection systems.

1.4 Stage Detectors

Stage detectors are a type of object detection algorithm that breaks the task into multiple stages.

Two-stage detectors first generate a set of region proposals, and then classify and refine the proposals. Single-stage detectors directly classify and regress object bounding boxes from a single convolutional neural network. Stage detectors have different strengths and weaknesses. Two-stage detectors are typically more accurate, but they are also slower. Single-stage detectors are faster, but they are not as accurate. The choice of stage detector depends on the specific application. For example, two-stage detectors are often used in self-driving cars, where accuracy is critical. Single-stage detectors are often used in mobile devices, where speed is critical.

- Two-stage detectors: R-CNN, Fast R-CNN, Faster R-CNN, Mask R-CNN
- Single-stage detectors: YOLO, SSD, RetinaNet

2. Related Works

2.1 Road Damage Detection

Road surface inspection is primarily based on visual observations by humans and quantitative analysis using expensive machines. Among these, the visual inspection approach not only requires experienced road managers, but also is time consuming and expensive. Furthermore, visual inspection tends to be inconsistent and unsustainable, which increases the risk associated with aging road infrastructure. Considering these issues, municipalities lacking the required resources do not conduct infrastructure inspections appropriately and frequently, increasing the risk posed by deteriorating structures.

In contrast, quantitative determination based on large-scale inspection, such as using a mobile measurement system (MMS) (KOKUSAI KOGYO CO., 2016) or laser-scanning method (Yu and Salari, 2011) is also widely conducted. An MMS obtains highly accurate geospatial information using a moving vehicle; this system comprises a global positioning system (GPS) unit, an internal measurement unit, digital measurable images, a digital camera, a laser scanner, and an omnidirectional video recorder. Though quantitative inspection is highly accurate, it is considerably expensive to conduct such comprehensive inspections especially for small municipalities that lack the required financial resources.

Therefore, considering the abovementioned issues, several attempts have been made to develop a method for analyzing road properties by using a combination of recordings by in-vehicle cameras and image processing technology to more efficiently inspect a road surface. For example, a previous study proposed an automated asphalt pavement crack detection method using image processing techniques and a naive Bayes-based machine-learning approach (Chun et al., 2015). In addition, a pothole-detection system using a commercial black-box camera has been previously proposed (Jo and Ryu, 2015). In recent times, it has become possible to quite accurately analyze the damage to road surfaces using deep neural networks (Zhang et al., 2016; Maeda et al., 2016; Zhang et al., 2017). For instance, Zhang et al. (Zhang et al., 2017) introduced CrackNet, which predicts class scores for all pixels. However, such road damage detection methods focus only on the determination of the existence of damage. Though some studies do classify the damage based on types— for example, Zalama et al. (Zalama et al., 2014) classified damage types vertically and horizontally, and Akarsu et al. (Akarsu et al., 2016) categorized damage into three types, namely, vertical, horizontal, and crocodile—most studies primarily focus on classifying damages between a few types. Therefore, for a practical damage detection model for use by municipalities, it is necessary to clearly distinguish and detect different types of road damage; this is because, depending on the type of damage, the road administrator needs to follow different approaches to rectify the damage.

Furthermore, the application of deep learning for road surface damage identification has been proposed by few studies, for example, studies by Maeda et al. (Maeda et al., 2016) and Zhang et al. (Zhang et al.,2016). However, the method proposed by Maeda et al. (Maeda et al., 2016), which uses 256 256 pixel images, identifies the damaged road surfaces, but does not classify them into different types. In addition, the method of Zhang et al. (Zhang et al., 2016) identifies whether damage occurred exclusively using a 99 x99 patch obtained from a 3264 2448 pixel image. Further, a 256 256 pixel damage classifier is applied using a sliding window approach (Felzenszwalb et al., 2010) for 5,888 3,584 pixel images in order to detect cracks on the concrete surface (Cha et al., 2017). In these studies, classification methods are applied to input images and damage is detected. Recently, it has been reported that object detection using end-to-end deep learning is more accurate and has a faster processing speed than using a combination of classification methods; this will be discussed in detail in 2.3. As an example of a method using end-to-end deep learning performing better than tradition methods, white line detection based on end-to-end deep learning using Over- Feat (Sermanet et al., 2013) outperformed a previously proposed empirical method (Huval et al., 2015). However, to the best of our knowledge, no example of the application of end-to-end deep learning method for road damage detection exists. It is important to note that classification refers to labeling an image rather than an object, whereas detection means assigning an image a label and identifying the objects coordinates as exemplified by the ImageNet competition (Deng et al., 2009).

2.2 Image Dataset of Road Surface Damage

Though an image dataset of the road surface exists, called the kitti dataset (Geiger et al., 2013), it is primarily used for applications related to automatic driving. However, to the best of our knowledge, no dataset tagged for road damage exists in the field. In all the studies focusing on road damage detection described in 2.1, in each

study, the researchers independently propose unique methods using acquired road images. Therefore, a comparison between the methods presented in these studies is difficult.

Furthermore, according to Mohan et al. (Mohan and Poobal, 2017), there are few studies that construct damage detection models using real data, and 20 of these studies use road images taken directly from above the road. In fact, it is difficult to reproduce the road images taken directly from above the roads, because doing so involves installing a camera outside the car body, which, in many countries, is a violation of the law; in addition, it is costly to maintain a dedicated car solely for road images. Therefore, we have developed a dataset of road damage images using the road images captured using a smartphone on the dashboard of a general passenger car; in addition, we made this dataset publicly available. Moreover, we show that road surface damage can be detected with considerably high accuracy even with images acquired by employing such a simple method.

2.3 Object Detection System

In general, for object detection, methods that apply an image classifier to an object detection task have become mainstream; these methods entail varying the size and position of the object in the test image, and then using the classifier to identify the object. The sliding window approach is a well-known example (Felzenszwalb et al., 2010). In the past few years, an approach involving the extraction of multiple candidate regions of objects using region proposals as typified by R-CNN, then making a classification decision with candidate regions using classifiers has also been reported (Girshick et al., 2014). However, the R-CNN approach can be time consuming because it requires more crops, leading to significant duplicate computation from overlapping crops. This calculation redundancy was solved using a Fast R-CNN (Girshick, 2015), which inputs the entire image once through a feature extractor so that crops share the computation load of feature extraction. As described above, image processing methods have historically developed at a considerable pace. In our study, we primarily focus on four recent object detection systems: the Faster R-CNN (Ren et al., 2015), the You Look Only Once (YOLO) (Redmon et al., 2016; Redmon and Farhadi, 2016) system, the Region-based Fully Convolutional Networks (R-FCN) system (Dai et al., 2016), and the Single Shot Multibox Detector (SSD) system (Liu et al., 2016).

2.3.1 Faster R-CNN

The Faster R-CNN (Ren et al., 2015) has two stages for detection. In the first stage, images are processed using a feature extractor (e.g., VGG, MobileNet) called the Region Proposal Network (RPN) and simultaneously, some intermediate level layers (e.g., ”conv5”) are used to predict class bounding box proposals. In the second stage, these box proposals are used to crop features from the same intermediate feature map, which are subsequently input to the remainder of the feature extractor in order to predict a class label and its bounding box refinement for each proposal. It is important to note that Faster R-CNN does not crop proposals directly from the image and re-runs crops through the feature extractor, which would lead to duplicated computations.

2.3.2 YOLO

YOLO is an object detection framework that can achieve high mean average precision (mAP) and speed (Redmon et al., 2016; Redmon and Farhadi, 2016). In addition, YOLO can predict the region and class of objects with a single CNN. An advantageous feature of YOLO is that its processing speed is considerably fast because it solves the problem as a mere regression, detecting objects by considering background information. The YOLO algorithm outputs the coordinates of the bounding box of the object candidate and the confidence of the inference after receiving an image as input.

2.3.3 R-FCN

R-FCN is another object detection framework, which was proposed by Dai et al. (Dai et al., 2016). Its architecture is that of a region-based, fully convolutional network for accurate and efficient object detection. Although Faster R-CNN is several times faster than Fast R-CNN, the region-specific component must be applied several hundred times per image. Instead of

cropping features from the same layer where the region proposals are predicted like in the case of the Faster R-CNN method, in the R-FCN method, crops are taken from the last layer of the features prior to prediction. This approach of pushing cropping to the last layer minimizes the amount of per-region computation that must be performed. Dai et al. (Dai et al., 2016) showed that the R-FCN model (using Resnet 101) could achieve accuracy comparable to Faster R-CNN often at faster running speeds.

2.3.4 SSD

SSD (Liu et al., 2016) is an object detection framework that uses a single feed-forward convolutional network to directly predict classes and anchor offsets without requiring a second stage per-proposal classification operation. The key feature of this framework is the use of multi-scale convolutional bounding box outputs attached to multiple feature maps at the top of the network.

2.4 Base Network

In all these object detection systems, a convolutional feature extractor as a base network is applied to the input image in order to obtain high-level features. The selection of the feature extractor is considerably important because the number of parameters and layers, the type of layers, and other properties directly affect the performance of the detector. Selected seven representative base networks, which are explained in 2.4, and three base networks to evaluate the results in Section 5. The six feature extractors are widely used in the field of computer vision.

Darknet-19: Darknet-19 (Redmon and Farhadi, 2016) is a base model of the YOLO framework. The model has 19 convolutional layers and 5 maxpooling layers.

VGG-16: VGG 16 (Simonyan and Zisserman, 2014) is a CNN with a total of 16 layers consisting of 13 convolution layers and 3 fully connected layers proposed in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2014. This model achieved good results in ILSVRC and COCO 2015 (classification, detection, and segmentation) considering the depth of the layers.

Resnet: Resnet, which refers to Deep Residual Learning, (He et al., 2016), is a structure for deep learning, particularly for CNNs, that enables high-precision learning in a very deep network; it was released by Microsoft Research in 2015. Accuracy beyond human ability is obtained by learning images with 154 layers. Resnet achieved an error rate of 3.57% with the ImageNet test set and won the first place in ILSVRC 2015 classification task.

Inception V2: Inception V2 (Ioffe and Szegedy, 2015) and Inception V3 (Szegedy et al., 2016) enable one to increase the depth and breadth of the network without increasing the number of parameters or the computational complexity by introducing so-called inception units.

Inception Resnet: Inception Resnet V2 (Szegedy et al., 2017) improves recognition accuracy by combining both residual connections and Inception units effectively.

MobileNet: MobileNet (Howard et al., 2017) has been shown to achieve an accuracy comparable to VGG-16 on ImageNet with only 1/30th of the computational cost and model size. MobileNet is designed for efficient inference in various mobile vision applications. Its building blocks are depthwise separable convolutions that factorize a standard convolution into a depthwise convolution and a 1 x 1 convolution, effectively reducing both the computational cost and number of parameters.

3. Dataset

3.1 Data collection

The video footage was recorded on the roads located along the National Highways (NH) and Tamil Nadu State Highways (TNSH) in the districts of Cuddalore and Chengalpattu in Tamil Nadu, India. The video was shot using a Redmi5 and an iPhone12 mobile phone. To avoid any playback errors, the duration of the video file is limited to ten minutes. Additionally, some of the images used in the study were obtained from Google Images and other road datasets [43], and ethical considerations regarding filming the

video were considered. Both the video and images are stored in Google Drive.

The images were originally in JPG format with varying resolutions. They were resized to 640 by 640 pixels to create square images and maintain consistency within the dataset. A total of 1640 images were captured under different weather conditions, including sunny, rainy, and winter. The images were also taken from various angles. Out of these, only 1129 images were deemed suitable for the study, as the remaining images either had poor color quality or did not clearly depict road damage.

Figure 1. State Highway Road Damage Dataset Sample.

3.2 Damage Classification

The international study report [44] classifies various types of road damage. However, this research article specifically focuses on four categories of damages, as indicated in table 1, that are commonly found on Indian highways and considered significant. It should be noted that certain types of damage, including longitudinal construction joint part (D01), lateral construction joint part (D11), Cross walk blur (D43), and White line blur (D43), are not addressed in this study. The damages are identified and represented by image annotation labels, which are discussed below.

Damage Type			Detail	Class Name
Crack	Linear Crack	Longitudinal	Wheel mark part	D00
		Lateral	Equal interval	D10
	Allegator		Partial Pavement, All Pavement'	D20
Other Corruption			Rutting, bump, pothole, Separation	D40

Table 1: Classification of Road surface Damages

3.3 Data Annotation

Data annotation involves the task of augmenting a dataset with metadata or labels to enhance its comprehensibility and usefulness for machine learning algorithms. Before the annotation process could commence, the dataset underwent a thorough examination to ensure the validity of all images, eliminate errors, and establish proper naming conventions, which consumed a substantial amount of time. The images were then annotated, with the exception of ambiguous ones, and assigned to their respective categories, as outlined in Table 1, which enumerates four distinct damage categories. To improve accessibility and usability, the images were systematically organized into folders and stored on Google Drive.

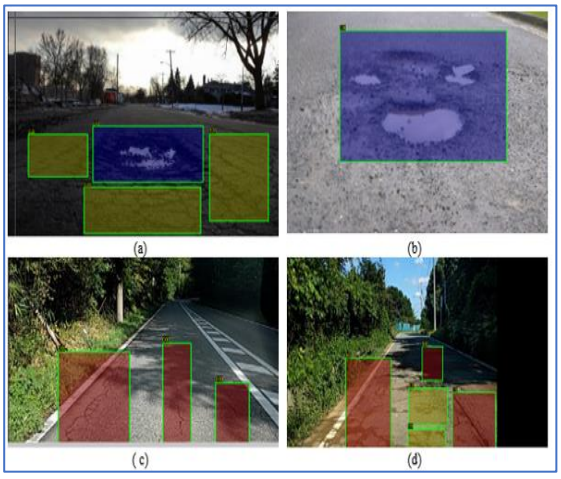

Figure 2. Damage class are marked with different colours.

3.4 Data Statistics

The statistics for the Tamil Nadu Road datasets are shown in Figure 5. It should be noticed that datasets have an uneven distribution of occurrences for different damage classes. image augmentation techniques were used to create a

balanced representation that can be used to train deep learning models.

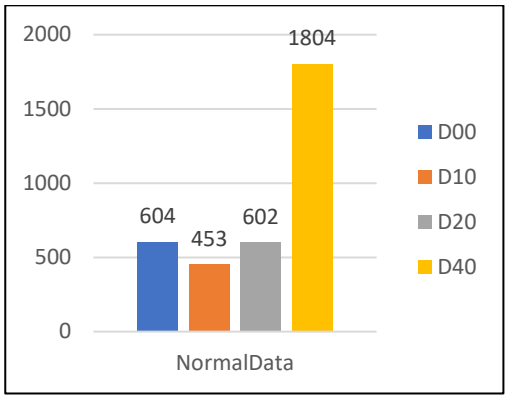

Figure 3. Statistics about the total of damage cases inside the underlying datasets.

Methodology

The data sources for this research are images or videos, which are discussed in Section 3. The research involves several high-level stages, including data gathering (images), data pre-processing, data augmentation, model preparation and validation, and real-time object detection. The precision, accuracy, and performance of the model are analysed and reported in Section 6. The steps are illustrated Figure 6. The stage data collection is explained in Section 3.

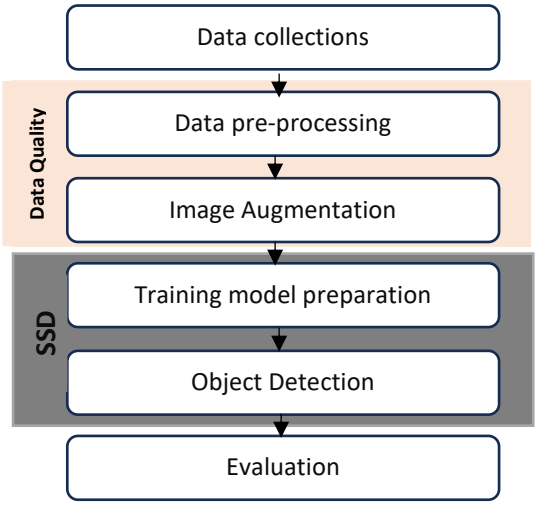

Figure 4 Steps for proposed methods.

3.3 Data Pre-processing

In this research, some of the key steps involved in data pre-processing for custom object detection:

- Data cleaning: Once the data has been gathered, it is important to perform data cleaning to remove any images that are of poor quality or contain incorrect annotations. This helps to ensure that the model is trained on high-quality data that is representative of the real-world scenarios.

- Resizing and normalization: The images in the dataset should be resized to a consistent size and normalized to ensure that the model receives consistent input. This helps to improve the training speed and accuracy of the model.

- Data splitting: Splitting data is an essential element of data science, especially for building accurate models based on machine learning to avoid overfitting. The dataset split into two parts, one part is used to evaluate or test the models and the other is used to train the models. Therefore, to evaluate the performance of a deep neural network model in an unbiased manner, a dataset properly divided into training and testing sets is required. In this proposed work, the data was split according to the standard 80:20 ratio, where 80% of the dataset was used for training and 20% for evaluation (testing and validation), using python script. Consequently, the performance of the model was measured based on 20% of the data that was neither used in training nor previously seen by the model to ensure that the analysis was fair. The training data statistical show in Figure 5.

- Converting data to model input format: The data should be converted to a format that can be input into the object detection model. This usually involves converting the images and annotations to a format that the model can read, such as Pascal VOC format.

Here, sources images are various dimension which are mentioned below. Image size refers to the physical dimensions of an image. Scaling refers to the process of changing the size of an image while preserving its aspect ratio. Scaling can be done by either increasing or decreasing the number of pixels in the image. This paper

includes a dataset of images of various sizes, as shown in Table 2.

Image sources	Dimension
RSDD2023	1280 x 870
Google Images	194 x 259
GRDD2020	600 x 600
Pothel_Dataset_V3	1280 x 780

Table 2. Dataset video file / image size

3.3.1 Data Augmentation

Image data augmentation is a technique commonly used in computer vision tasks, such as image classification, object detection, and semantic segmentation. It involves applying a set of predefined transformations to existing images to create new variations of the original dataset. The purpose of data augmentation is to increase the size and diversity of the training data, which can lead to better model generalization and improved performance.

Here are some commonly used image data augmentation techniques are Horizontal/Vertical Flipping, Rotation, Scaling and Cropping, Translation, Shearing, Zooming, Adding Noise and Color Jittering. These techniques, along with others, can be applied individually or in combination to generate augmented images. It's important to strike a balance between introducing enough diversity to the dataset without distorting the original content too much, as excessively transformed images might confuse the model during training.

This research has challenges during the augmentation, for classes D00 & D10, some augmentations, such as flip rotation and left-to-right rotation, are invalid. The results are distorted when class D00 is turned to become class D10. Class D20 and D40 are exempt from image augmentation, nevertheless. All classes can benefit from contrast augmentation, which is a component of image augmentation. By enhancing 1129 base images, 2159 augmented images with 1791 bounding boxes are generated. As illustrated in Figure 4, these techniques were applied to each image to obtain a new training and testing sample. The training and testing portions of this dataset had a ratio of 80:20.

Classes	Augmented Image Frame Counts	
	Before	After
D00	726	1316
D10	526	1013
D20	707	1450
D40	1908	3382
No. of Overall Images	1129	2159
No. of Rejected Images	538	711

Table 3. Statistical of Normal and Augmented Images

3.4 Single Stage Detection (SSD)

Single Shot Detection (SSD) is a popular object detection algorithm that aims to achieve high detection accuracy while maintaining real-time performance. Unlike some other object detection methods that require multiple stages and multiple network passes, SSD performs detection in a single shot. Some of the key features are highlighted below.

- Multi-scale feature maps
- Default anchor boxes
- Predictions at multiple scales
- Training with hard negatives
- Efficient inference

SSD has been widely used in various applications, including pedestrian detection, vehicle detection, and general object detection tasks. It strikes a balance between accuracy and efficiency, making it suitable for real-time applications that require both speed and accuracy.

3.5 Architecture

Single Shot Detection (SSD) can be implemented using different backbone architectures such as MobileNet and ResNet. Let's look at SSD with MobileNet and SSD with ResNet:

3.5.1 MobileNet

SSD with MobileNet combines the efficiency of MobileNet, a lightweight convolutional neural network architecture, with the object detection capabilities of SSD. The MobileNet backbone network is used to extract feature maps from the input image. These feature maps capture

semantic information at different resolutions. SSD-specific convolutional layers are added on top of the MobileNet backbone to predict class probabilities and bounding box offsets for each anchor box. Non-maximum suppression is then applied to filter out redundant detections. SSD with MobileNet is known for its excellent speed and accuracy trade-off, making it suitable for real-time and resource-constrained applications.

The SSD detection layers typically include multiple feature maps at different scales, allowing the detection of objects at various sizes. These feature maps are progressively down sampled to capture objects at different scales and improve localization accuracy.

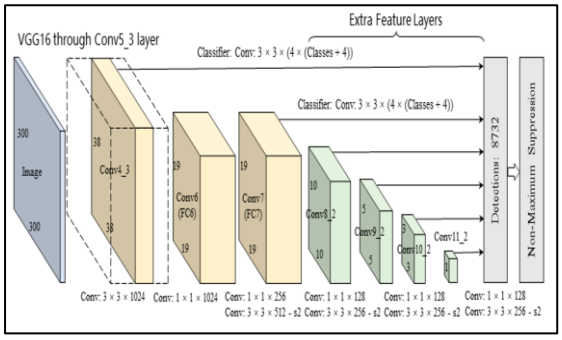

Figure 5 Architecture of a convolutional neural network with an SSD detector

4.3.2 Improved SSD with Resnet(R-SSD)

SSD with ResNet integrates the powerful feature extraction capabilities of ResNet into the SSD framework. ResNet is a deeper and more complex convolutional neural network architecture known for its ability to handle deeper networks without suffering from the degradation problem. The ResNet backbone network is utilized to extract feature maps, which capture both low-level and high-level semantic information. Additional convolutional layers specific to SSD are added on top of the ResNet backbone to predict class probabilities and bounding box offsets. Non-maximum suppression is then applied to refine the final set of detections. SSD with ResNet offers improved detection accuracy, especially for objects with fine-grained details and complex visual patterns.

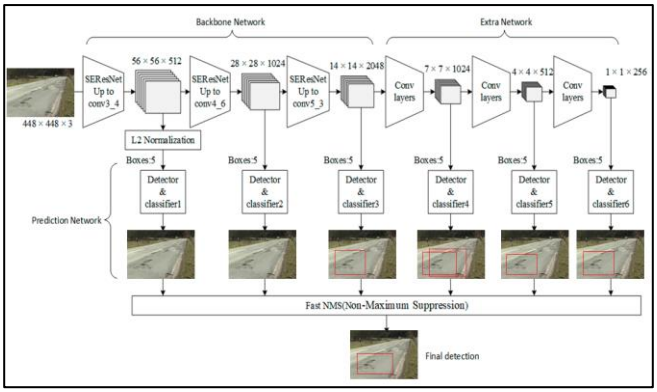

Figure 6 Architecture of SSD with ResNet

Both SSD with MobileNet and SSD with ResNet provide efficient and effective solutions for object detection. The choice between them depends on the specific requirements of the application, such as the available computational resources, desired detection accuracy, and real-time performance constraints.

3.6 Additional Fields Multibox Detector

The traditional SSD design consists of a fully convolutional neural network that generates bounding boxes of a defined size and class scores for the existence of objects inside the boxes. This is followed by a non-maximum suppression phase that selects one or more boxes with the highest-class scores. The output collection of boxes is expressed as offsets to a set of arbitrarily selected default boxes with various sizes and aspect ratios.

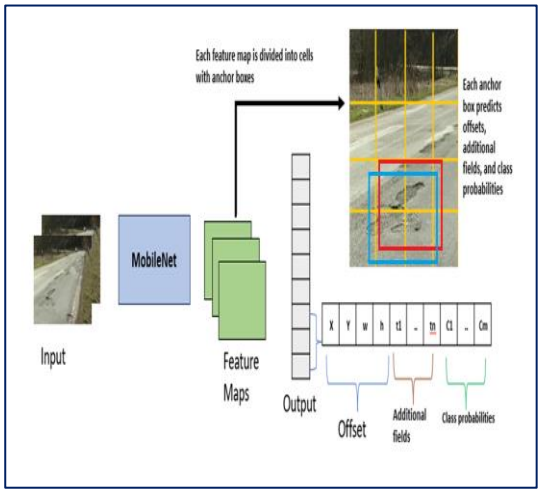

Figure 7 Object Detection approaches using MobileNet.

3.7 Training model

created two training models, both of which are presented here.

- MobileNet
- ResNet

These models were developed in a CUDA environment for research conducted in Colab research environment. Detailed information about the environment configuration can be found in section 5. The logs were saved in a repository, and the models were thoroughly examined and monitored. Different thresholds were employed to validate Road Damage Detection. Based on the research findings, ResNet with demonstrates superior predictive capabilities compared to MobileNet. The training process for this specific research objective is currently underway, and the historical records of object predictions are stored in Google Drive to substantiate the claims.

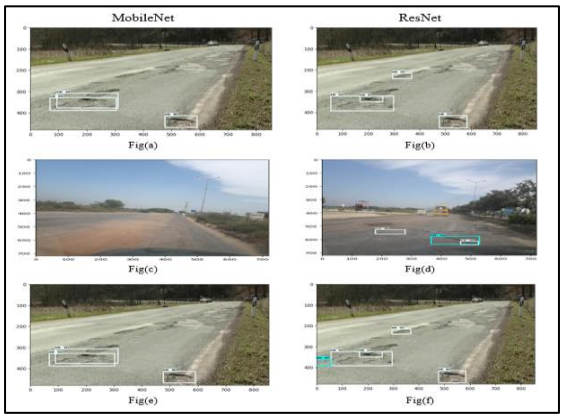

Figure 8 Detected samples using MobileNet Vs ResNet

4. Experimental setup

This research's data was analysed using the GPU environment of the NVIDIA Jetson nano hardware device. Since the CPU was unable to process the image data, the GPU environment was used. Most of the time training model prepared on Colab Research environment, which is another environment that is also available. Depending on availability, Google Colab gives users access to one of several different NVIDIA GPUs. This study's processing was done on a Tesla T4 processor running Torch 1.10.2+cu111 CUDA:0 with 12 GB of RAM. used Google Drive, which has a default 15 GB storage restriction, for storing files.

5.1 Evaluation Measures

For the experiments evaluated, we used several metrics to evaluate the models' performance. The formulae are defined as follows:

$$\text{Accuracy} = \frac{TP+TN}{\text{TP+FP+FN+TN}} \quad (1)$$

$$\text{TPR} = \frac{TP}{\text{TP+FN}} \quad (2)$$

$$\text{FPR} = \frac{FP}{\text{FP+TN}} \quad (3)$$

$$\text{Recall} = \frac{TP}{\text{TP+FN}} \quad (4)$$

$$\text{Precision} = \frac{TP}{\text{TP+FP}} \quad (5)$$

$$\text{F1 Score} = \frac{2\ x\ Precision\ x\ Recall}{\text{Precision + Recall}} \quad (6)$$

where TP, FN, TN, and FP represent the true positive prediction, false negative prediction, true negative prediction, and false positive prediction, respectively, as shown in Equations (1)–(6).

5. Result

There are various parameters to increase the object model training performance and Object detection counts.

below parameters make changes in object training model

- Image capture Frame Per Second (FPS)
- Image-capturing device speed of travel (device often maintained on car).
- Capturing the brightness of an image.
- Distance between the road surface and the image-capture device,
- The angle of the image-capture project.
- During the image capture sun light direction
- Environmental weather and Geo-Location
- Image capturing instrument shake during the image capturing.
- training image quality
- training Image counts and etc.,

below parameter make changes in Object detection

- Label smoothing

- Non-max suppression threshold
- Enhanced learning model
- Object detection confidence
- Image sizes etc.,

The parameters mentioned above have a practical impact on both the training and prediction phases of the object training model. It should be noted that the prediction scores of different models may vary, and this phenomenon is not applicable to all cases. In certain instances, abnormal discrepancies can occur due to the parameter adjustments made during the preparation of the training model. In this research, the prediction score values for a set of photos that have been validated (object detection) by ResNet and MobileNet models are evaluated. In comparison to the MobileNet model, the ResNet model has a greater number of predictions and higher score values. According to the current research, the ResNet technique is more suited for detecting road surface damage.

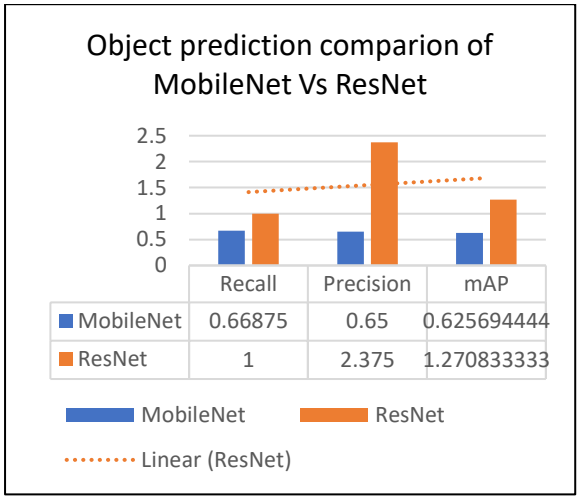

Figure 9 Prediction comparison of MobileNet Vs ResNet

6. Conclusion

The method outlined in this research can be utilized by municipal government officials to generate images of road surface damage, enabling them to prioritize and address specific areas in need of repair. Additionally, the research findings indicate that the dataset created, TNRSD2023, as well as the training models, are publicly available through a git repository, facilitating further examination and utilization by interested parties.

References

[1] AAoSHaT, O. (2008). Bridging the gap–restoring and rebuilding the nations bridges. Washington (DC): American Association of State Highway and Transportation Officials.

[2] Adeli, H. (2001). Neural networks in civil engineering: 1989–2000. Computer-Aided Civil and Infrastructure Engineering, 16(2):126–142.

[3 Akarsu, B., KARAKO¨ SE,M., PARLAK, K., Erhan, A., and SARIMADEN, A. (2016). A fast and adaptive road defect detection approach using computer vision with real time implementation.

[4] Cha, Y.-J., Choi,W., and B¨uy¨uk¨ozt¨urk, O. (2017). Deep learning-based crack damage detection using convolutional neural networks. Computer-Aided Civil and Infrastructure Engineering, 32(5):361–378.

[5] Chun, P.-j., Hashimoto, K., Kataoka, N., Kuramoto, N., and Ohga, M. (2015). Asphalt pavement crack detection using image processing and na¨ıve bayes based machine learning approach. Journal of Japan Society of Civil Engineers, Ser. E1 (Pavement Engineering), 70(3).

[6] Dai, J., Li, Y., He, K., and Sun, J. (2016). R-fcn: Object detection via region-based fully convolutional networks. In Advances in neural information processing systems, pages 379–387.

[7] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, pages 248–255. IEEE.

[8] Everingham, M., Eslami, S. A., Van Gool, L.,Williams, C. K., Winn, J., and Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. International journal of computer vision, 111(1):98– 136.

[9] Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. International journal of computer vision, 88(2):303–338.

[10] Felzenszwalb, P. F., Girshick, R. B., McAllester, D., and Ramanan, D. (2010). Object detection with discriminatively trained part-based models. IEEE transactions on pattern analysis and machine intelligence, 32(9):1627–1645.

[11] Geiger, A., Lenz, P., Stiller, C., and Urtasun, R. (2013). Vision meets robotics: The kitti dataset. The International Journal of Robotics Research, 32(11):1231–1237.

[12] Girshick, R. (2015). Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, pages 1440–1448.

[13] Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 580–587.

[14] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778.

[15] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.

[16] Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., Fischer, I., Wojna, Z., Song, Y., Guadarrama, S., et al. (2016). Speed/accuracy trade-offs for modern convolutional object detectors. arXiv preprint arXiv:1611.10012.

[17] Huval, B., Wang, T., Tandon, S., Kiske, J., Song, W., Pazhayampallil, J., Andriluka, M., Rajpurkar, P., Migimatsu, T., Cheng-Yue, R., et al. (2015). An empirical evaluation of deep learning on highway driving. arXiv preprint arXiv:1504.01716.

[18] Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International Conference on Machine Learning, pages 448–456.

[19] Jo, Y. and Ryu, S. (2015). Pothole detection system using a black-box camera. Sensors, 15(11):29316–29331.

[20] JRA (2013). Maintenance and repair guide book of the pavement 2013. Japan Road Association, 1st. edition.

[21] Kazuya, T., Akira, K., Shun, F., and Takeki, I. (2013).An effective surface inspection method of urban roads according to the pavement management situation of local governments. Japan Scienc and Technology Information Aggregator.

[22] KOKUSAI KOGYO CO., L. (2016). Mms(mobile measurement system).

[23] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Doll´ar, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In European conference on computer vision, pages 740–755. Springer.

[24] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016). Ssd: Single shot multibox detector. In European conference on computer vision, pages 21–37. Springer.

[25] Maeda, H., Sekimoto, Y., and Seto, T. (2016). Lightweight road manager: smartphone-based automatic determination of road damage status by deep neural network. In Proceedings of the 5th ACM SIGSPATIAL International Workshop on Mobile Geographic Information Systems, pages 37–45. ACM.

[26] MLIT (2016). Present state and future of social capital aging. infrastructure maintenance information.

[27] Mohan, A. and Poobal, S. (2017). Crack detection using image processing: A critical review and analysis. Alexandria Engineering Journal.

[28] Nishikawa, T., Yoshida, J., Sugiyama, T., and Fujino, Y. (2012). Concrete crack detection by multiple sequential image filtering. Computer-Aided Civil and Infrastructure Engineering, 27(1):29–47.

[29] O'Byrne, M., Ghosh, B., Schoefs, F., and Pakrashi, V. (2014). Regionally enhanced multiphase segmentation technique for damaged surfaces. Computer-Aided Civil and Infrastructure Engineering, 29(9):644–658.

[30] Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 779–788.

[31] Redmon, J. and Farhadi, A. (2016). Yolo9000: better, faster, stronger. arXiv preprint arXiv:1612.08242.

[32] Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems, pages 91–99.

[33] Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., and LeCun, Y. (2013). Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229.

[34] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[35] Szegedy, C., Ioffe, S., Vanhoucke, V., and Alemi, A. A. (2017). Inception-v4, inception-resnet and the impact of residual connections on learning. In AAAI, pages 4278–4284.

[36] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2818–2826.

[37] Yu, X. and Salari, E. (2011). Pavement pothole detection and severity measurement using laser imaging. In Electro/Information Technology (EIT), 2011 IEEE International Conference on, pages 1–5. IEEE.

[38] Zalama, E., G´omez-Garc´ıa-Bermejo, J., Medina, R., and Llamas, J. (2014). Road crack detection using visual features extracted by gabor filters. Computer-Aided Civil and Infrastructure Engineering, 29(5):342–358.

[39] Zhang, A.,Wang, K. C., Li, B., Yang, E., Dai, X., Peng, Y., Fei, Y., Liu, Y., Li, J. Q., and Chen, C. (2017). Automated pixel-level pavement crack detection on 3d asphalt surfaces using a deep-learning network. Computer-Aided Civil and Infrastructure Engineering, 32(10):805–819.

[40] Zhang, L., Yang, F., Zhang, Y. D., and Zhu, Y. J. (2016). Road crack detection using deep convolutional neural network. In Image Processing (ICIP), 2016 IEEE International Conference on, pages 3708–3712. IEEE.

[41] R. Palani[1] , Dr. N. Puviarasan[2] and Dr. A. Rama Prasath[3], Literature Review of Road Damage Detection with Repairing Cost Estimation, International Journal of Mechanical Engineering, ISSN: 0974-58232 , Vol. 7 No. 2 February 2022

[42] R. Palani[1] , Dr. N. Puviarasan[2] and Dr. A. Rama Prasath[3] , Collection of distinct image frames using Structured similarity Index measured, Advanced Engineering Science, ISSN: 2096-3246 | Jan 2023

[43] R. Palani[1] , Dr. N. Puviarasan[2] and Dr. A. Rama Prasath[3], IMPROVE CUSTOM OBJECT DETECTION OF ROAD SURFACES BY SMOOTHING OF LABELS AND IMAGE ENHANCEMENT, Journal of Data Acquisition and Processing, ISSN: 1004-9037| Vol. 38 (1) 2023

[44] R. Palani[1] , Dr. N. Puviarasan[2] and Dr. A. Rama Prasath[3], Road Surface Damage Detection With Ensemble Techniques, Scandinavian Journal of Information Systems | ISSN: 0905-0167|1901-0990| 2023 35(1)

[45] R. Palani[1] , Dr. N. Puviarasan[2] and Dr. A. Rama Prasath[3], TECHNIQUES FOR IOU NON-MAX SUPPRESSION TO IMPROVE ROAD SURFACE DAMAGE DETECTION ACCURACY | Vol. 13 No. 01 (2023), Hipatia Press | ISSN: 2014-2862

[46] Hiroya Maeda (et al), Road Damage Detection Using Deep Neural Networks with Images Captured Through a Smartphone, arXiv:1801.09454v2 [cs.CV] 2 Feb 2018