Optimization and sensitivity analysis of existing deep learning models for pavement surface monitoring using low-quality images

Ronald Roberts, Fabien Menant, Gaetano Di Mino, Vincent Baltazart

Title: Optimization and sensitivity analysis of existing Deep Learning models for Pavement Surface monitoring using low-quality images

Authors: Ronald Roberts[1]*, Fabien Menant[1], Gaetano Di Mino[2] and Vincent Baltazart[3]

[1]MAST-LAMES, Université Gustave Eiffel, IFSTTAR, Campus de Nantes, 44344 Bouguenais, France

[2]DIING - Department of Engineering, University of Palermo, Viale delle Scienze ed.8, Palermo, 90128, Italy;

[3]SII-COSYS, Université Gustave Eiffel, IFSTTAR, Campus de Nantes, 44344 Bouguenais, France

*Corresponding author, Tel.: +33 07 61 64 16 69

Email addresses:

Ronald Roberts - ronald.roberts@univ-eiffel.fr;

Fabien Menant - fabien.menant@univ-eiffel.fr;

Gaetano Di Mino – gaetano.dimino@unipa.it;

Vincent Baltazart – vincent.baltazart@univ-eiffel.fr.

Abstract

Automated pavement distress detection systems have become increasingly sought after by road agencies to increase the efficiency of field surveys and reduce the likelihood of insufficient road condition data. However, many modern approaches are developed without practical testing using real-world scenarios. This study addresses this by practically analysing Deep Learning models to detect pavement distresses using French Secondary road surface images, given the issues of limited available road condition data in those networks. The study specifically explores several experimental and sensitivity-testing strategies using augmentation and hyperparameter case studies to bolster practical model instrumentation and implementation. The tests achieve adequate distress detection performance and provide an understanding of how changing aspects of the workflow influence the actual engineering application, thus taking another step towards low-cost automation of aspects of the pavement management system.

Keywords

Deep Learning, Transfer Learning; Pavement distresses; Pavement Management Systems; Monitoring Pavement Surfaces

1.0 Introduction

1.1 Context and Overview

With its 1,000,000 km, France has one of the largest and most dense road networks in Europe. Like other countries, the main issue for road managers now is more to maintain the network than extend it. The preservation of road assets consists of defining and planning maintenance works from several types of technical data: inventory data (e.g. age and form of the pavement structure), traffic data and pavement monitoring data that aim to detect the presence of road defects. Pavement monitoring data typically come from measurements or surveys provided on the visible part of the road (*i.e.* surface layer) as well as the non-visible part (*i.e.* base layer). One of the most important pavement monitoring data to have is the distress survey that mainly consists of detecting and locating different types of cracks at the road surface. These defects may indeed reveal issues about the surface layer (minor issues) or about the structure as a whole, which is then more serious. This type of data is key to producing adequate pavement management plans for maintenance, rehabilitation and reconstruction activities.

For main roads (highways and national roads), distress surveys are usually done using very accurate measurement systems which are also very expensive to buy and use (only a dozen of these systems operate in France). This is acceptable because the main roads are highly strategic for economic activities (e.g. transportation of goods) and also because they represent only a small part of the network (about 2% in France). For the rest of the network and especially for the secondary roads (rural roads) that approximately represent 38% of the French network (380,000 km)[1], these systems are financially and operationally less adapted. Therefore, secondary road managers look out for adaptive measurement systems that can solve both the economic constraints and the need for data collection. For now, such a system does not really exist so managers have either no data or data that are often too inaccurate/unreliable to be used as they come from visual inspections.

1.2 Goals of the study

This study aims to test, adapt and evaluate some Deep Learning (DL) techniques applied to "low-quality" images for carrying out distress surveys on road networks. With "low-quality images" defined as images coming from low-cost devices such as webcams and camera phones with low photo resolutions (images where 1 pixel represents more than 1 mm of actual detail) and other adverse factors such as the presence of shadows, altering brightness conditions and blurs. In this study, only a few types of distresses have been considered, specifically those, which are the most relevant for the management of secondary roads[2]. Faced with economic constraints and the need to cover the large size of the secondary road networks, images from low-cost cameras embedded into a fleet of vehicles (e.g. patrol vehicles) are used. In this way, it is possible to have many pictures of the road surface, continuously and frequently updated, on the network but with a low quality. In this context, a robust image processing technique is highly desired in order to detect and classify pavement distresses from the images. DL can help this challenge considering that it is well adapted to the use of probe vehicles, which provides numerous images and thus facilitates an important need of the learning process. Nevertheless, we do not know if DL will adequately perform for low-quality images (because of the camera characteristics) and for "noisy" images that include the effects of several factors: mechanical vibrations, brightness changes, shadows, vehicle speed, etc. This paper outlines a methodology used to test and adapt Deep Learning models from experimental data that are representative of real conditions and

provides case studies of changing different aspects of the workflow in practical scenarios. The novel aspect of the work is centered around the results of case studies testing the various available DL models that mimic practical changes that could occur in the real world. The results are meant to offer a practical understanding of how to test and understand using augmentations and hyperparameters for the specific task of detecting and understanding pavement distresses in a Deep Learning engineering application.

1.3 The rising use of Deep Learning for problem-solving

Deep Learning (DL) has seen significant rises in recent years [3] to solve many practical problems with numerous studies focused on defect detection, diagnosis and prediction [4]. This is important as authorities are continually seeking a range of technologies because they are key to instrumenting road infrastructure for detecting and monitoring damages [5]. DL generally involves using models comprised of several processing layers that learn data representations using multiple layers of generalization [6]. The key capability of a DL system is its ability to extract very complex features using large amounts of data by uncovering hidden patterns within them [7]. This allows complex models to be easily built for important tasks. The general architectures of DL models have been widely studied [6] with continual advancements to their structures. The systems can however suffer from errors, chief among them being systematic and random ones [8]. With more testing and calibration, errors can be averted by improving training using more generalizable data and better hypothesis functions [4]. To this end, this study aims to carry out a series of optimization and sensitivity tests to help improve the understanding of building models to detect road distresses. This will provide more clarity on practical model development. This is especially important because of the gap that often lies when connecting the fields of DL and practical engineering application and in particular, road asset management. When looking at DL research in pavement engineering, the primary focus has been on the computer science phases with many studies focusing on improving the structure rather than practical application and calibration. This is seen in many studies focused on generating new model structures and algorithms that can reduce errors based on generic isolated datasets.

This is often a problem with DL due to the disconnect between modelling and real-world applications where the connection of important problems to the larger world of science and society is lost [9]. This often happens because DL researchers are not always experts in the

relevant technical areas and therefore focus more on the model structure rather than their adaptation. Assisting the adaptation to real-world implementation is this study's primary goal to help deliver models that are context-driven and understood by practitioners and not just researchers. This study, therefore, develops distress detection models using road images from secondary roads in France to build models and then carry out experimental testing to improve performance and understand the effects of different practical workflow changes. Models developed are done so directly from a civil engineering practitioner's point of view and not with the typical idea of generating a mathematically advanced model that may suffer practical application issues.

2.0 Overview of applications in pavement engineering

It is important to highlight aspects of the significant research done using DL for Pavement Engineering uses [10–12]. These applications can be allotted to the following areas:

- Condition and performance predictions, where models are used to predict future pavement conditions using historical data [13]
- Pavement Management systems where models are used to optimize their functions
- Serviceability forecasting where models are used to monitor the ability of the pavement to serve its user's needs (e.g. comfort) [14]
- Structural evaluations, where models are used to evaluate the pavement's physical structure [15]
- Modelling materials, where models are used to monitor materials in the pavement structure and how they react to different conditions [16]
- Image analysis and classification, where models are used to analyse images of the pavement and wherein lies the focus of this study [17–19].

The most researched area is Pavement Image analysis in which the focus is split between image classifications, where images are classified based on a distress in the image; and object detection, where distresses are detected using bounding boxes or masks in the image. Image classification has issues as multiple distresses frequently occur in an image making it impossible to label a particular image with one label. With object detection, multiple overlapping objects within an image can be identified, making it is possible to detect multiple overlapping pavement distresses. This is key for road asset databases. Many studies have focused on developing models to indicate

the presence or lack of a distress [20]. Crack detection has largely been seen as the most researched category. They are seen as the most common distress [21] and are easier to measure, with the simple required metrics such as the crack's width and length. Many studies have developed specific DL models for crack detection and analysis using 2D and 3D imagery [20,22–27]. Comparisons are often made to results from image-based systems for crack detection such as CrackIT [28] and other available crack databases [29,30]. CrackIT is a Matlab based system created to detect and characterize cracks. It must be noted, however, that whilst the major category of focus has been cracks; this is only one distress category. Other studies have considered multiple distress categories and datasets of multiple types. A German team developed a model for pavement distress application based on imagery across the German road network using a mobile mapping system [31]. The German Asphalt Pavement Distress (GAPS) dataset was developed by this team, which was utilized to detect and analyse six different distress categories based on German road manuals considering their developed neural network. Recently, the dataset was further updated with better annotations and a larger database [32]. Other teams have considered the use of street images where DL models are used to detect distresses, mainly cracks [33]. Using Google street view images, others have developed models to detect multiple distresses using the You Only Look Once (YOLO) model and network [34] and subsequently used image segmentation to detect the crack size [35]. Studies have also been carried out considering fourteen distress categories with the application of semantic segmentation and object detection algorithms on a dataset in Naples, Italy [36]. This was used to create a decision support system based on the occurrence of predicted distresses. There is also a large dataset of road surfaces called the KITTI dataset [37], however, this dataset was created primarily for assisting with automated driving research. Other studies have considered the use of transfer learning using models initially trained in the ImageNet database. Using these available models, models were developed to detect cracks [20]. The use of smartphones for capturing datasets has also been considered with one of the applications being the use of a DL model on image patches collected from a smartphone to detect cracks [38]. There has also been the development of a Japanese road distress database using smartphones [39]. This has provided the basis for competitions such as the Global Road Damage Detection Challenge [40], where different models are submitted to obtain higher performances using different configurations. This has led to different configurations using different base networks and models having the same goal to detect distresses [41,42]. Whilst

these developed models represent a significant step forward in distress detection and analysis, changes to model training and testing due to augmentation and hyperparameter testing has largely been missing. This study further explores this by using models built around similar bases but focusing on highlighting the effects of different changes to the model workflow.

3.0 Methodology

3.1 Data collection

A database containing 56,783 images captured across approximately 100km of secondary roads in France was used. The images were taken during road surveys using a webcam affixed to a survey vehicle and aimed directly at the pavement's surface in a top-down view as shown in Figure 1.

Figure 1. Example of instrumentation used to collect images

The webcam used was a Logitech C920 Pro with a max resolution of 1080p equipped at a vertical distance of 1.8m from the surface. This means the road surface represents 2mm/pixel on an image with the actual space taken being 1.5m in length and 2m in width. This configuration offers a good compromise between the minimum image resolution and ground measurement area and importantly accordingly with the needs of road managers for rural French roads. After an overview of the database, it was determined that five distress categories would be considered. These categories are in line with category descriptions of the French Manual for distress collection and identification [43], which is critical as this is what is used by the industry.

The categories are:

1. Ravelling (denoted as RV)
2. Transverse Cracking (denoted as TC)

3. Longitudinal cracking out of wheel path (denoted as LC1)
4. Longitudinal cracking within wheel path (denoted as LC2)
5. Alligator/Block cracking (denoted as BC1)

These are the most relevant to the road managers and given the type of roads (secondary), managers only need to know the mere presence of the distress in the road sections being surveyed, making object detection a perfect tool. The distinction between the longitudinal cracks in and out of the wheel path is relevant to determining whether the cracking is due to pavement fatigue or not due to the direct loads of vehicles. Examples of these are shown in Figure 2.

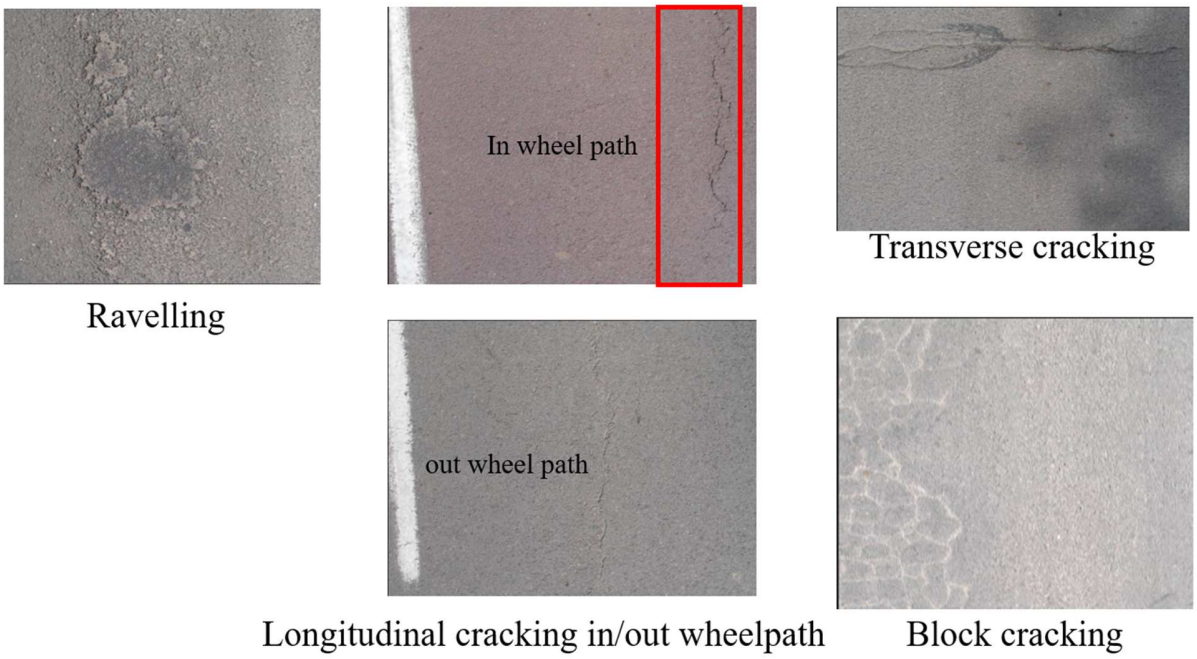

Figure 2. Examples of distress categories used in modelling

The next step was analysing the complete dataset to pinpoint images containing the relevant distresses. This was done manually and the distresses were identified and recorded using the *LabelImg* open-source software [44] in the Pascal VOC [45] xml format. For annotations Complex cases (such as alligator cracking inside block cracking) are not labelled as it is considered as the same in the French methodology for secondary roads The distresses identified are detailed in Table 1. This dataset amounted to 2325 images with 3280 annotations (1.4 per image) out of the complete set (4.09% of images).

Table 1. Annotated distresses

Distress	Number of occurrences	% of occurrences of the total dataset

RV	904	1.59
LC1	676	1.19
LC2	545	0.96
TC	565	1.00
BC1	590	1.04

3.2 Development using pre-existing DL models

For model development, the TensorFlow (tf) [46] library was used. Within the library, there is a vast object detection model zoo [47], which provides base models, trained on millions of images that can be utilized for transfer learning for new tasks. Transfer learning enables training models based on the 'backbone' of preexisting models and therefore does not require training from scratch. It has been shown to be efficient for developing new models [48,49]. The models used were:

- ssd_mobilenet_v2
- ssd_inception_v2
- ssdlite_mobilenet_v2
- ssd_mobilenet_v1_ppn
- ssd_mobilenet_v1_fpn
- faster_rcnn_inception_v2

These models were chosen based on earlier work on the detection of pavement distresses [50]. Besides these models, one other widely used model that should also be mentioned is the Yolo – You only look once [51] model which uses a single convolutional network for predictions and is the backbone of many applications. The network does not look at the entire image. It splits the model into a grid and then bounding boxes are generated within that image. For each box, the network then outputs the class probabilities and offset values. Boxes that have a class probability above a set threshold value are identified and used to pinpoint the location of the object in the image. It does have issues detecting small objects with noted issues such as missed and false detections [52,53]. As pavement distresses can be very small, the decision was made not to use it in this particular study but it can be considered for future research. Furthermore, the model is not included in the object detection model zoo [47] that was used for this study.

The architectures of these models will not be considered here as they have widely been studied and the importance of this work is on the workflow and sensitivity testing using the specific case study applications. The largest difference between the models is that the first three take a single-stage approach using only one snapshot of the image to detect multiple objects within the image, whilst the final two use a region proposal approach wherein two steps are employed, the first being to generate region proposals and then the second where the predictions are made. For the training, Figure 3 highlights the key stages of the process, wherein the files necessary for the training are developed using various scripts to be used in the TensorFlow environment.

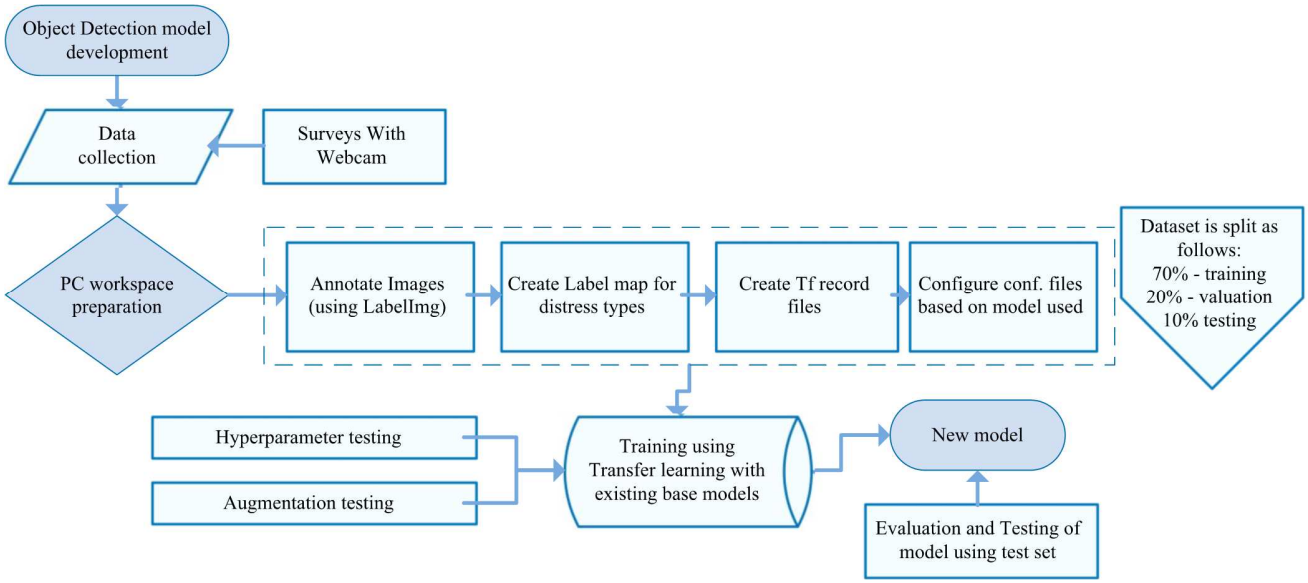

Figure 3. Workflow for developing TensorFlow pavement distress detection model

The tf record format which is used for the model development is key as the information on the annotations and their locations are embedded within these files. Additionally, it is important to point out that the dataset was split for training, validation and testing in the ratio of 70:20:10 which is typical for this type of model development [50]. The split was made ensuring that a similar distribution of the distresses was in each dataset to avoid overfitting or inflated results at any point. The percentage image split of each dataset is provided in Table 2. From this, it can be seen that the distributions are done to ensure no set is inflated with any particular category. The distributions cannot be exact, because of multiple distresses in images and overlaps, which is how they appear in the real world.

Table 2. Analysis of commonly used metrics

Distress	% of distress occurrence in the training set	% of distress occurrence in the validation set	% of distress occurrence in the testing set

BC1	19%	18%	20%
LC1	23%	19%	22%
LC2	18%	19%	15%
RV	22%	21%	25%
TC	18%	23%	18%

The critical stage in the process is the tuning, which occurs, based on the evaluation results of the model during the training process. This is monitored using Tensorboard, which allows the user to check the losses and other metrics of evaluation during the training and evaluation stages of the model. The workstations utilized for modelling was a Windows 10 PC with an NVIDIA Quadro P4000 GPU (8 GB ram) and total CPU memory of 32 GB @ 3.7GHz and 64bit processor. The general speed of the models on the workstation is approximately 0.6 seconds per training step with an average of 200,000 steps utilized in the modelling process. The faster of the models, as expected, was the Mobilenet one, which was developed for quick mobile deployment.

3.3 Determination of optimization tests for models

For the optimization tests, the focus was on augmenting images with different alterations that could mimic practical changes such as light exposure and blurs, factors that are typically linked to the use of low-cost cameras. Additionally, it became apparent that the size of the object being detected was critical so considerations of the bounding box in the actual distribution were also done. The model hyperparameters were varied to obtain the best performance of the model. The hyperparameters considered were the learning rate, the decay rate of the learning rate, intersection threshold, anchor-box sizes and scale. Throughout the training, these parameters were monitored and changed with learning rates varying between 0.001 and 0.002. The former was the more utilized version that saw quicker losses in the training and was in line with previous models run in past experiments [50]. A decay of 0.95 in the learning rate was generally employed after 50,000 steps and again was in line with previous work done [50]. It should be noted that these are not the only available parameters that could be changed but represent key ones that contribute to the effectiveness of the model in practical applications.

3.4 Performance metrics

Model performance is typically evaluated based on the metrics of precision and recall. Precision can be classified as the results within a test that are relevant to the detection problem whilst

Recall refers to the percentage of relevant results which are in turn correctly identified by the model. These metrics are computed from the confusion matrix values, which detail the occurrences of true and false positives (TP, FP) and negatives, (TN, FN). Table 3 shows the metrics and offers a practical explanation regarding detecting pavement distresses.

Table 3. Analysis of commonly used metrics

Metric	Description	Formula
Precision	The proportion of accurate distresses that are detected against the total number of predicted distresses	TP/(TP+FP)
Recall	The proportion of accurate distresses that are detected against the total number of actual distresses	TP/(TP+FN)
f1 score	The harmonic mean between precision and recall where the two are treated equally	2((1/Precision) + (1/Recall))

Whilst generally, it is common to have information on both precision and recall as they represent the key efficiencies of the DL model, the use of these metrics largely depends on the motivation of the study and model. The critical element that must be considered is what is important to the designated stakeholder. It should be further stated that if a stakeholder emphasizes that there should be limited false positives then precision is considered a good metric for performance evaluation. With regards to pavement distresses, this would mean that the stakeholder does not want a large number of falsely detected pavement distresses which is the case for a great number of French road managers, therefore stressing the importance of this metric for the application on French roads. However, should the stakeholder place greater emphasis on simply identifying the distress and missing a distress is not acceptable then recall is a better metric. If both of these are important then the harmonized f1 score should be the preferred metric. These considerations are key to the practical implementation and use of the DL models. For engineering applications, the use of these metrics may also heavily depend on financial resources and the importance of not missing a distress during a survey. Therefore, this is a critical consideration for practical evaluations.

3.5 Anchor-box considerations

As the size and perspective of images obtained in practical situations may differ, the sizes of distresses in an image can also change. A significant change could be from a panoramic view to a

top-down view as used in this study. In these two situations, the number of pixels in the image where the distress occurs would change significantly and this needs to be considered in hyperparameter selection. One of the key considerations while developing many detection models is the anchor-box configuration. During the modelling process, models make numerous predictions on where an object could be within the image space. Typically, thousands of boxes are generated on the image to assist in this process. These boxes, named anchor-boxes, are defined before the training begins and are typically rectangular in size and used by the model to pinpoint the location of the detection by comparing their location and size to the location of the ground truth boxes and using the differences to train. These boxes should be consistent with the location, shape and size of the objects that the model is attempting to detect. This is important as the overlap between the boxes generated and the annotated training set is used for the model to understand the particular features of the annotations. As a result, it is important to establish a clear understanding of the space occupied by the distresses to produce effective models. The traffic direction may for example be different in different countries causing distresses to be localized in different lanes. This is another clear justification for the inclusion of this study on anchor-box location for practical applications. For this, the images within the used database were analyzed. To this end, the CSV files generated from the annotated images containing the information on the bounding box location and sizes were analysed. Their sizes were normalized using the resizing function ‘compute_new_static_size’ from the TensorFlow object detection API [47]. The distribution of the sizes and location of the bounding boxes' width and height were calculated and are displayed in Figure 4.

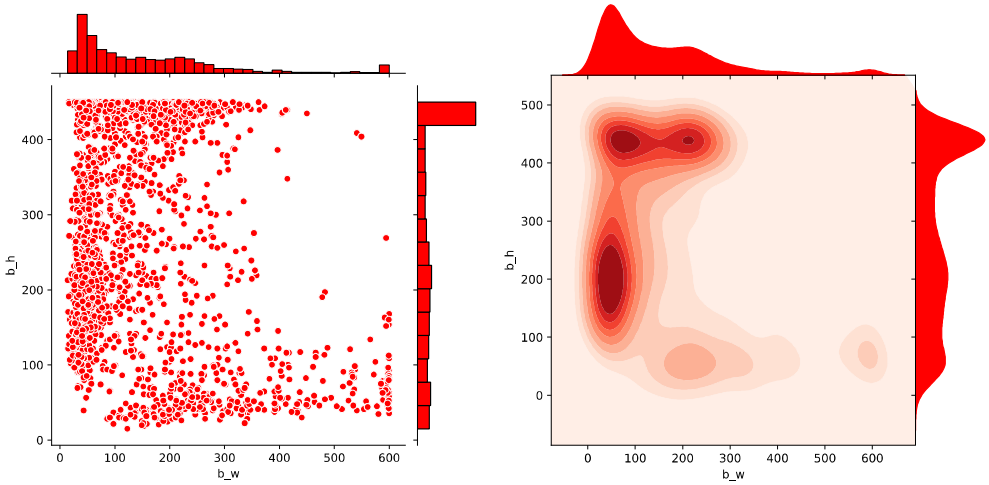

Figure 4. Distribution of the bounding boxes of the annotated dataset (dots in the graph on the left represent the location of the distress detections and the graph on the right shows a heatmap of these points across the dataset)

To further visualize this, generic examples of anchor-box configurations used on the images are illustrated in Figure 5, where the colored boxes represent the model's default anchor-boxes.

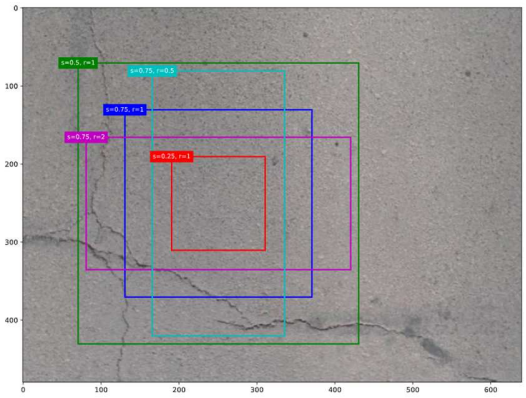

Figure 5. Generic anchor-box configuration on example image from the dataset

Considering the distributions, it was decided to use optimized anchor-boxes sizes to identify the locations of the distress annotations. This was done by carrying out calculations to determine a more appropriate aspect ratio (AR) and scale sizes to be used in the training. The AR refers to the ratio of the width to height of the box and the scale refers to the size of the box with respect to the size of the image being used. A k-means clustering approach was utilized for this with four clusters applied for the bounding boxes in an attempt for the model to produce a greater Intersection over Union (IOU) when creating the anchor-boxes and searching for the annotation location during training. The IOU is a value commonly used to measure the overlap of predictions versus the ground truth. The k-means clustering was considered suitable given the distribution shape of the boxes. The clustering approach was based on a calculation of the IOU between the boxes and the 'k' number of clusters given in equation 1.

$$IOU = \frac{\text{intersection}}{\text{box\_area} + \text{cluster\_area} - \text{intersection}} \quad (1)$$

Following this, the average IOU between the boxes and clusters was obtained and then the k-means clustering was applied using the Euclidean distance to identify the nearest clusters in the process. A cluster of four was used to reproduce four numbers for the AR and scale in line with

the previous number of scales and aspect ratios utilized for other models. Using this in Python, the clusters were returned in terms of width and height. Subsequently, the AR and scale size were calculated based on the image size as shown in equations 2 and 3:

$$\text{aspect\_ratio} = \frac{\text{clusterbox\_width}}{\text{clusterbox\_height}} \tag{2}$$

$$\text{scale} = \frac{\text{clusterbox\_height} \times \sqrt{\text{aspect\_ratio}}}{\text{height\_of\_gridanchor}} \tag{3}$$

4.0 Results and discussion

4.1 Anchor-box optimizations

In performing the anchor-box optimizations as described in section 3, the process resulted in optimized aspect ratios and sizes, which are shown along with the default values in Table 4.

Table 4. Changes to Anchor-boxes

Generic Aspect ratios	Aspect ratios after clustering	Generic scale sizes	Scale sizes after clustering
0.5	0.1953	0.25	0.4640
1.0	4.4461	0.50	0.6692
2.0	0.4893	1.00	1.5916
0.75	0.2302	2.00	0.9465

An example of the new anchor-boxes, when visualized on the image, is given in Figure 6.

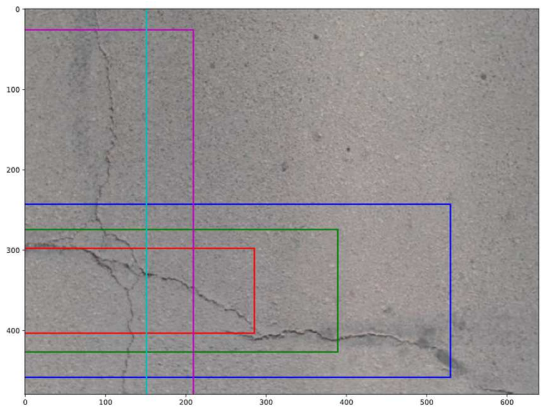

Figure 6. Optimized anchor-box configuration on an image from the dataset

From the image, it is seen that the anchor-boxes are more suitable to the shape and size of the annotated distresses as they appear more similar in size. This is key to quicker training and higher developed accuracies for practical application. Future work will carry out further testing on the

effect of using optimized anchor-box configurations. After the setup and in the first instance of testing, the results of the full dataset with the different base models and utilizing the five previously referenced classes are shown in Figure 7.

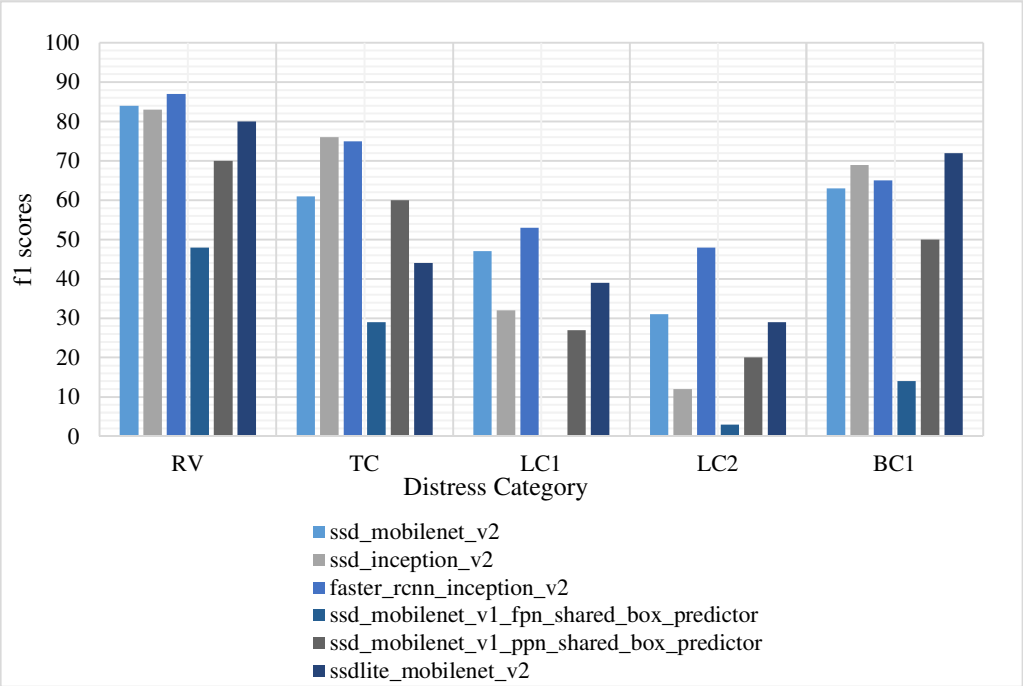

Figure 7. Level of performance on initial models for distress detection

Considering the results of these models, there were high levels of precision achieved for the majority of distresses but lower levels of recall, particularly for the longitudinal cracking and block, cracking categories. It was also noted that the base models of ssd_mobilenet_v1_fpn_shared_box_predictor and ssd_mobilenet_v1_ppn_shared_box_predictor both performed quite poorly with the former of two seeming unable to produce usable results at least under the used configurations of the models. This may be due to the models trying to speed up detection compared to the regular SSD configuration [54] and missing the smaller detections of pavement distresses contained in this dataset. It should not be ruled out that other configurations could yield usable results and therefore further test of these models could remain an option. Examples of the detections shown on actual images are given in Figure 8.

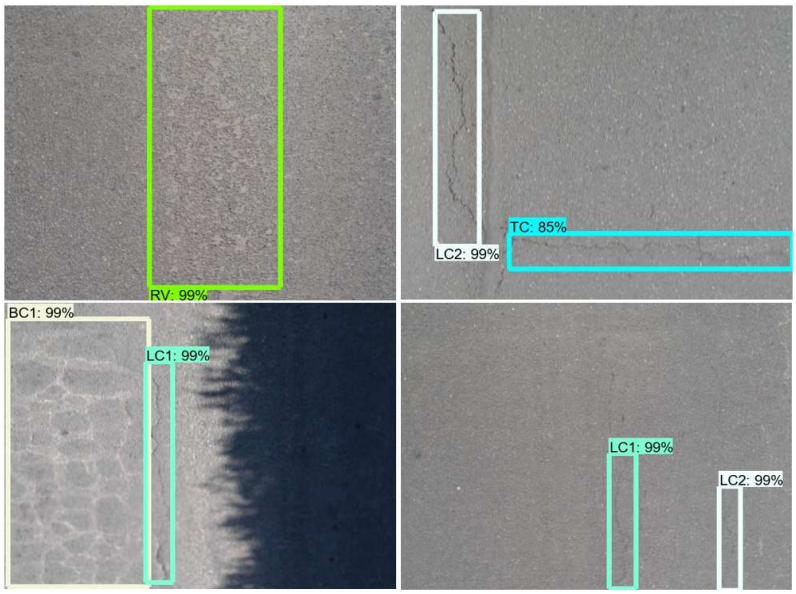

Figure 8. Example of detections given by the DL models

4.2 Case studies to test the use of Augmentations

In an attempt to increase the dataset size and produce better results (given the tendency of DL to function better with more data [55]), augmentations were considered and applied to the images before training. Augmentations are variations of an image, wherein particular image characteristics are modified. These changes can vary from the color, size or rotation of the images and this process is used to increase the size of the dataset. Augmentation is generally assumed to increase the variability and diversity of the data which will enable producing a more robust DL model, considering the many variances that can exist in the specific data [56]. This study aimed to explore this and understand the effects of using various augmentations for the given dataset. The purpose of this was not to declare a specific combination of augmentations as the winner but rather to see if there was indeed a boost to the model performance and understand whether this process should be carried out. To carry this out tests were run without any augmentations, and this model was used as a base case scenario. Subsequently, test sets using the same pipeline were run changing different augmentations. These augmentations included several changes to the image appearance, which could practically happen, with changes to light and movement during a survey such as vibrations of a camera, causing image blurs. An example of the augmentations used is shown in Figure 9.

Figure 9. Example of an augmentation where the lighting was artificially changed, (middle image is the original one)

In the figure, the original image appears in the middle and brightened and darkened versions are shown to the left and right, representing differing lighting conditions that can occur during surveys. The images show how light can make a significant difference in identifying the presence of distresses. In total, nine different situations were considered for the case study with results shown in Figure 10 and which were done using Python scripts to add the changes to the images. It should be noted that horizontal and vertical flips were performed but these flips are mirror flips wherein the orientation of the transverse or longitudinal cracks are not interfered with and instead the crack would simply appear in the other half of the image. This is important as the use of an incorrect flip or rotation could result in a transverse crack noted as a longitudinal one or vice versa.

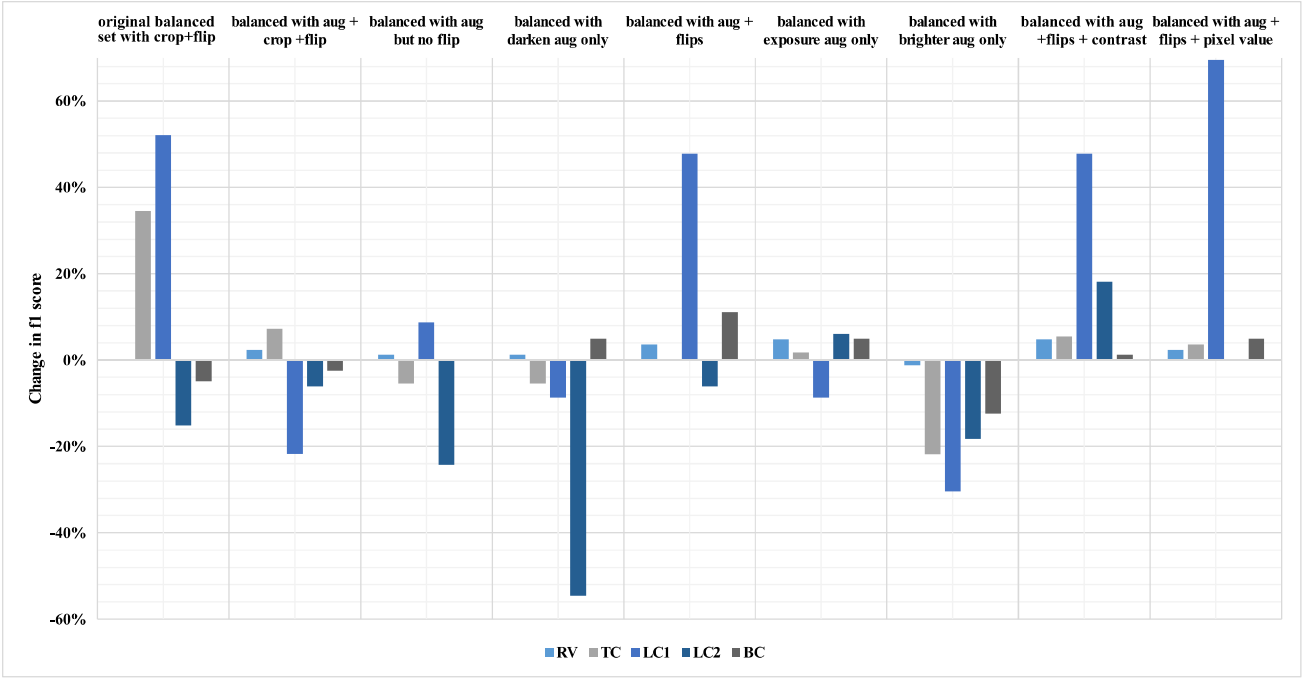

Figure 10. Evolution of the f1 scores of the model due to augmentations for each distress type

When analyzing the results and changes of the augmentations, the changes varied across the distress types and the results are shown in Figure 9. In the figure, it is shown that some augmentations had very marginal effects on the model's performance with some even producing negative effects in particular instances. The following important points were ascertained:

i. The augmentation of a random crop had negative effects, particularly on distresses where the location was important (Longitudinal cracking within the wheel path – LC2). This is rational, as a crop would allow the distress not to appear in its precise location. However, it could be considered as a positive for models where the position of the distress is not important. This underscores the importance of establishing criteria based on specific site information and exact distresses in practical engineering applications.

ii. The effect of brightening appears more effective than that of darkening which caused significant drops in performance. When examining photos, it was seen that this augmentation showed mixed performance as some images appeared better when brightened whereas some in their original instance suffered from too much light and therefore darkening helped. This led to the consideration of exposure, contract and pixel value changes as these alterations focus on the particular allocation of pixels within the image and not a generic application to the entire dataset. These augmentations are however not natural and therefore particular attention has to be paid to their use.

iii. The use of horizontal and vertical flips appeared to provide a marginally positive response in performance.

iv. A combination of flips with random changes to the contrast proved to be the best combination of augmentation for this test.

v. The augmentations generally had little positive effect on increasing the performance of the models to detect longitudinal cracking.

vi. It is simply not good enough to assume the positive effect of an augmentation and for practical implementation, testing must be done to determine the best combinations.

441 After testing the best three models as defined by the combined metrics are given in Table 5.

Table 5. Performance of Models

		Base models		
Distress	Metric	ssd_mobilenet_v2	ssd_inception_v2	faster_rcnn_inception_v2
RV	Precision	98%	95%	88%
	Recall	68%	79%	86%
	F1 score	81%	86%	87%
TC	Precision	95%	96%	58%
	Recall	29%	39%	86%
	F1 score	44%	55%	70%
LC1	Precision	91%	93%	60%
	Recall	33%	21%	81%
	F1 score	49%	34%	69%
LC2	Precision	79%	82%	53%
	Recall	28%	19%	52%
	F1 score	42%	31%	53%
BC1	Precision	92%	90%	76%
	Recall	59%	90%	92%
	F1 score	72%	90%	83%

From Table 5, the models based on the SSD network with the inception and mobilenet configurations show greater levels of precision but lower levels of recall as compared to the faster-rcnn based one. It is also useful to note that the first two models take a 'one-shot' look approach to determine the object in the images [57], whereas the faster-rcnn one takes a region proposal approach [58] making the case that for distress detection, this plays a significant role considering the sizes of pavement distresses. The models' f1 scores are shown in Figure 11 showing the best overall model as the faster-rcnn based one with values similar to the many studies done on DL models in the field as shown by [17]. A comparison of precision to recall for the models is also shown in Figure 12 to highlight the strengths and weaknesses of the models.

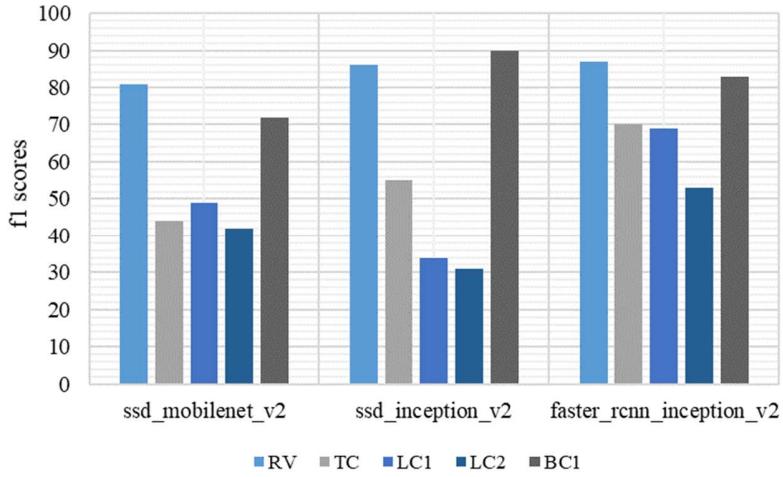

Figure 11. Level of performance of the best three models

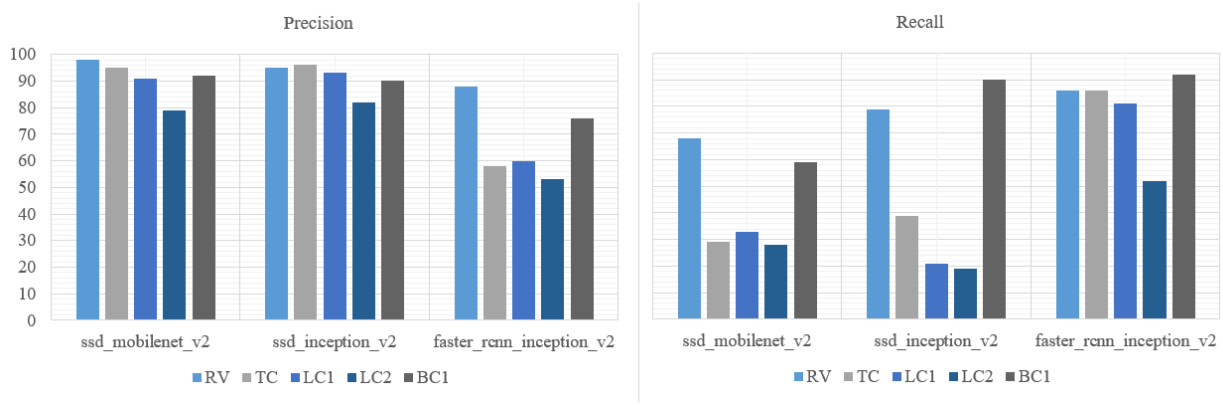

Figure 12. Comparison of Precision to Recall for the best three models

From Figure 11, it can be clearly shown that the first two models (one-shot approach types) produce outstanding levels of precision but suffer from poor levels of recall for this type of application. Therefore, it should be noted that the first two models, therefore, produce lower levels of false positives – as high levels of precision imply – but in the same instance produce a larger number of false negatives (missed detections). The importance of this has to be decided by the stakeholder that the model is directed at, as some companies/authorities may value false detections higher than missed detections. Nevertheless, it is an important result to be noted for practical application.

4.3 Case studies to test Confidence and IOU thresholds

Case studies were also carried out to determine the effects of changing the Intersection over Union (IOU) and Confidence thresholds. The idea behind this was to boost the detection rate in order to reduce the false-negative rate whilst still maintaining an adequate performance level for the models. The IOU threshold has a range of 0 to 1, where one represents a perfect overlap of detection and the ground truth. Confidence meanwhile refers to the model's score in predicting the class, which is the 'certainty' the model has when making a prediction. The confidence threshold, therefore, sets the minimum value that the model score should be before returning a result of a detection. For this test, a range of IOU thresholds was utilized from 0.10 to 0.95 with a fixed confidence threshold score of 0.50. Additionally, the confidence score was also altered with a range of 0.10 to 0.95 with a fixed IOU Threshold of 0.50. This process was tested on the best model iterations of the inception and faster_rcnn based models. Each case study's results are given below.

4.3.1 Effects on ssd_inceptionv2 based model

In the first case, the confidence level was controlled with varying levels of threshold and there was no effect on the general model performance. After a threshold of 75% was applied, however, the f1 scores began to fall. In the second test, the IOU threshold was maintained and confidence levels were changed. With this test, the performance of the models in detecting distresses dropped across categories incrementally. Both tests are shown in Figure 13.

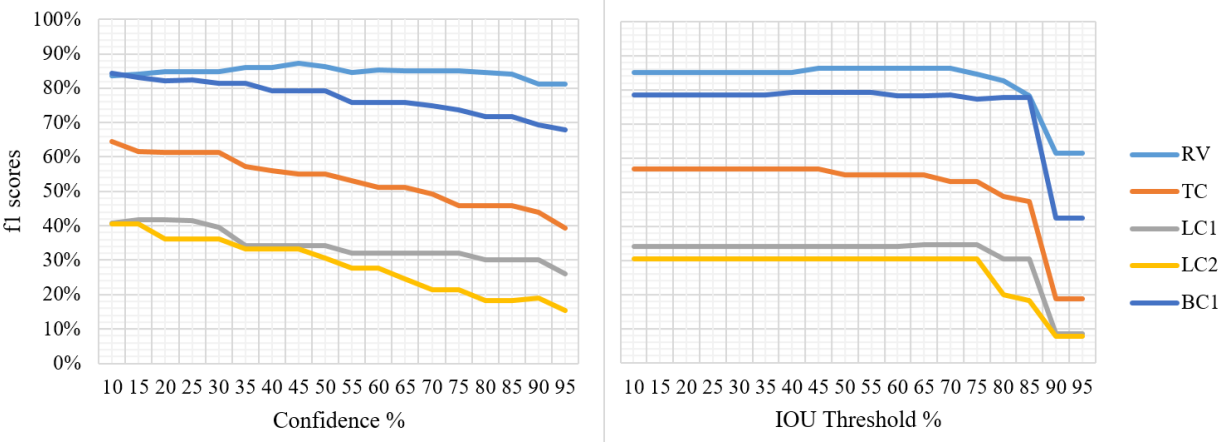

Figure 13. Variation of f1 scores when altering the thresholds

4.3.2 Effects on Faster-RCNN based model

For this test, the model's performance again was maintained until approximately 75% IOU threshold where its performance declined significantly and quickly as shown in Figure 14.

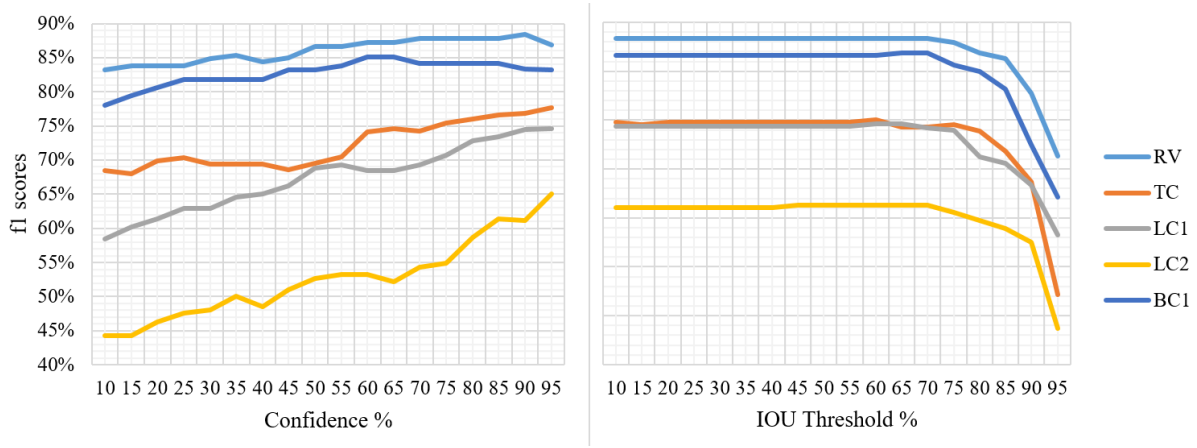

Figure 14. Variation of f1 scores when altering the thresholds

However, in the confidence threshold test, the performance of the model increased incrementally as the confidence threshold was increased with increases of up to 30% from initial results across all distress categories. Given these results, it was concluded that the tuning of the confidence level does affect the models used. However, it appeared with the model based on inception, the overall performance of the models could not be increased even with alterations to the confidence levels. To understand what could be causing these changes, the numbers of positives and negatives detected were similarly observed across the models with varying confidence levels. In analyzing these parameters, the changes in the number of false positives were identified as a cause for the change. These graphs are shown in Figure 15.

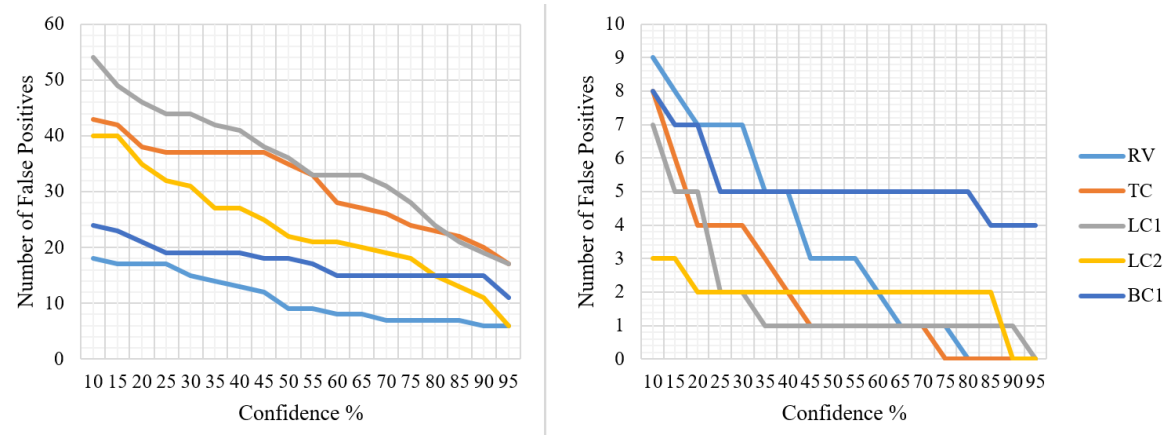

Figure 15. Variation of f1 scores when altering the thresholds

Concerning false positives, both had a similar decreasing pattern but because the numbers of false positives were substantially less for the inception based model, the effect of the confidence threshold did not produce better results. Therefore, it is clear that when altering these values, the trained model itself must have the flexibility for any altering to be effective. Essentially, there must be a higher number of false positives in the results which could be filtered out with higher confidence levels. Whilst the inception model initially produced higher precision, it also had a smaller number of false positives and therefore did not have much scope to be changed by the hypothesis of altering confidence. Conversely, the faster-rcnn based model had more false positives that could be removed and therefore increasing the confidence allowed the model to reduce the errors. It must be restated here that the models were all tested using the same test dataset, which was not revealed to the model until the testing phase and therefore represented a test akin to what could happen in practice. This is stated to establish that the effect was directly related to the models' performance on these types of images and detections.

4.4 Case studies to validate models using blank images with no distresses

As a further test on the models' robustness, a case study was done using blank images where there were no distresses were present. This was done as the initial performance of the models developed were based on test images in which distresses were present, which is typical of this workflow type. To analyze what would happen with images with no distress another test set was used. The images within this test set had images of pavements with no distresses and involving situations where other methods would likely produce a false-positive result and were images previously never used in the training or testing phases. Examples are given in Figure 16.

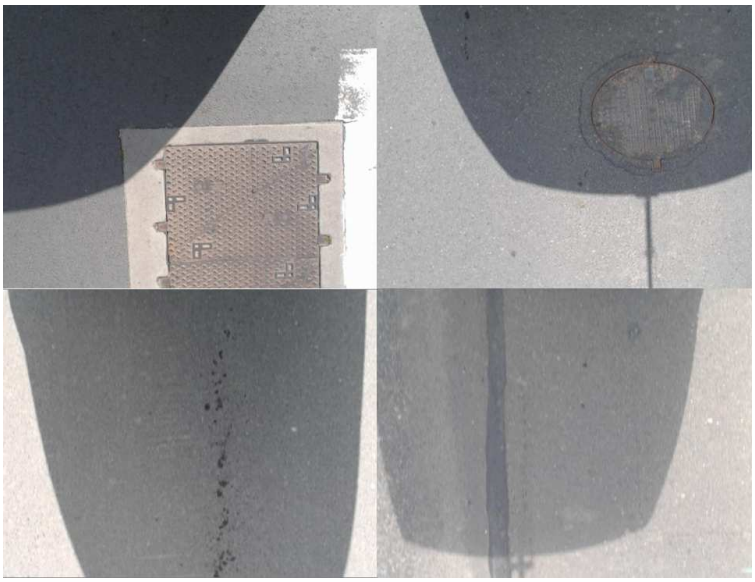

Figure 16. Examples of blank images used in the case study

These included images which were particularly chosen having characteristics likely to produce a false positive, such as shadow projections, skid marks, manhole covers and with high contrast and brightness issues as shown. The test case tried to utilize images as shown where there were marks similar to cracks, manhole covers, discolored sections and shadows created by the rod of the surveillance vehicle and marks from patches. A total of 210 images were used. The models were run over these images resulting in six cases of false detections representing a false rate of 2.86%. Given the small number of false detections, it should be considered as a positive outcome underlying the performance of the model. Additionally, the model did not suffer from the issues associated with other segmentation tools, which routinely produced false detections associated with shadows and markings on the road. None of the images showing the shadow of the surveillance vehicle were falsely detected as a pavement distress during the exercise. The images with false detections were ones with marks resembling a crack and of slightly discolored roads where it appears that ravelling has not occurred but could appear in the future.

4.5 Transferability of models

Finally, it became important to consider the transferability of detection models in other situations. In many studies, this is not considered and models are assumed to have similar results in other applications, which is not necessarily the case. As a result, the best model was tested with a different view (panoramic view) on images from the same corridor with secondary roads.

This was done using images that were simultaneously captured at panoramic view whilst the main top-down images were captured so that detections could be compared. In the first test, the images used were captured from a panoramic view using a camera installed within the survey vehicle. This resulted in images as shown in Figure 17.

Figure 17. Example of view of panoramic images used in modelling

The other images were captured as shown in the setup in section 3. The change of camera perspective however meant that the size of the distress is different and also surrounding items such as trees, electric posts and other vehicles are now captured during a survey. None of these elements appear in surveys when using a top-down approach. Therefore, it was assumed that the results would be poor when tested using the model developed and the actual results validated this. There were still several instances where the model correctly identified the distress as shown in Figure 18.

Figure 18: Examples of images where the model correctly identified distress in a panoramic view

The most commonly correctly identified distresses were that of longitudinal and transverse cracking. However, the model also produced a series of false and missed detections as shown in Figure 19 leading to an ineffective survey.

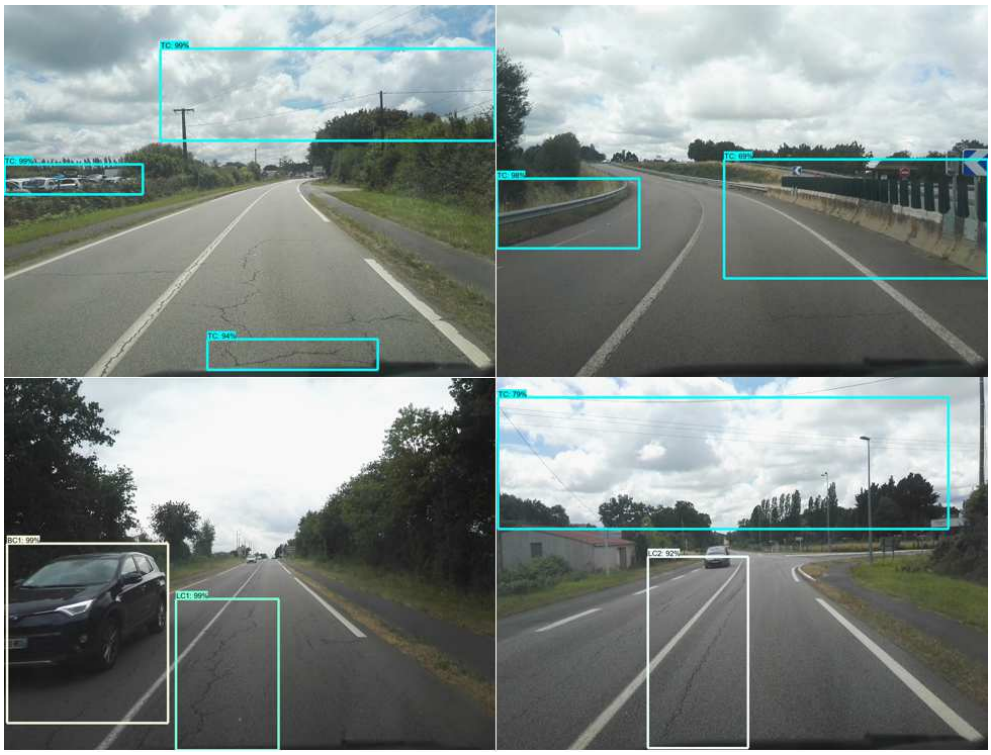

Figure 19: Examples of incorrect detections by model on panoramic images

The model was shown to incorrectly detect items such as electric wires and rails as cracks. This can be expected given in the training none of these elements appeared and the general shape of these objects are similar to that of cracks. Additionally, the models used in this work were adapted to the size of distress in the top-down view, which would be significantly different in these views.

These tests, therefore, established that for the model to work, the camera perspective should remain the same. It is a significant result as the surrounding characteristics and image orientation have to be considered. This test was done to validate that models are not transferable over different environments where there are a significant number of features not shown on the images used for training the original models. For transferability, the images used should have similar characteristics and backgrounds. Whilst all features on similar networks cannot be featured in a training set, the difference should not be significant as a change from a top-down view to that of a panoramic view. A model showing high performance metrics is not the only thing that matters in this case and the background context is critical.

5.0 Conclusions

The study included the practical assessment of different optimization and sensitivity tests and strategies for instrumenting Deep Learning models for detecting pavement distresses on secondary roads in France. For this kind of network, a summary of the detection of distresses (presence/absence) is expected and needed by road managers, unlike main networks where a more accurate survey is needed to be able to observe the evolution of degradations over time. The models are able to produce a global overview of the presence of distresses on a corridor and this can be tracked by simply rerunning the survey on the corridor without the need to track the exact evolution of every distress. This adds significant value for secondary road managers, who are the target of this study. Future work will consider models that can track the evolution of distresses to assist larger networks and road managers.

The main goals of this work are to understand the impacts of changing particular aspects of the DL workflow in a practical and effective manner, especially for these secondary road types. To do this, several DL models are tested with varying optimization and sensitivity tests to understand the effects on the performance when detecting particular distresses for secondary roads.

The experimental results from the case studies demonstrate the value of testing strategies considering a real-world approach and with real datasets, and showcase the performance under different practical situations. The case studies help to provide an understanding of how DL models react to different changes in the context of pavement engineering and application. The results can be used to create better workflows for practical implementation and they offer a new view of the typical DL implementation and workflow, showing the novelty of the research. This includes considerations of different augmentations, the optimization of anchor-boxes for models and considerations of changes to the confidence and IOU thresholds. These provide a practical examination of how DL distress models react, which can help the end-user in troubleshooting a model or deciding on which model to use. The use of augmentations in particular help demonstrate the effects different types can have on a distress detection model. Moreover, they importantly demonstrate that a broad application of them should not be done and each case should be treated differently considering that some augmentations can in fact have a negative result on performance. The consideration of the different metrics also presents a clear pipeline consideration for the final stakeholder and the results of the paper can be used to make decisions on a model type for the required application. The results of this particular study indicate that

models that utilize a two-step approach to the detection such as the faster-rcnn models seem more appropriate to pavement distress detection in the specific context of this work. These models also showed that with the particular optimization tests relating to the IOU and confidence threshold, further increases of up to 30% in the f1 scores can be possible which is an important result. However, it must be reiterated that results of this are context-specific and the more important results are the shown variations from the sensitivity and optimization tests demonstrating the value of similar exercises before a model is practically implemented. Furthermore, the test considering a different camera orientation demonstrates that models should also be only transferred across situations where the expected images that will be used remain similar as models need to understand the specific environmental context. The results also confirm that the models, using only low-cost cameras, can be accurate for the intended purpose as opposed to typical visual inspections.

The use of DL techniques is likely to take over the domain still held by common images segmentation methods in particular when the quality of images is quite low. Although the initial results in this study were interesting, better robustness will be sought through both an optimization of several parameters in the DL workflow and a further expanded sensitivity study. Future tests will consider comparisons to results using higher quality images, more datasets of similar roads conditions and the implementation of more distress categories such as glazing/bleeding, representing more visual road deformations.

Acknowledgments

The authors would like to acknowledge and thank the county council of La Manche (France) for assisting with the data collection and Mr S Todkar, of SII-COSYS, Université Gustave Eiffel, for help in proofreading the work. The research presented in this paper was carried out as part of the SMARTI ETN project under the H2020-MSCA-ETN-2016. This project has received funding from the European Union's H2020 Programme for research, technological development and demonstration under grant agreement number 721493.

References

[1] L'Observatoire National de la Route, Rapport Observatoire National de la Route (ONR) 2019, Rapport. (2019) 1–72. https://www.idrrim.com/ressources/documents/11/7102-IDRRIM_Rapport_ONR-2019.pdf (Accessed date: 12 November 2021).

[2] Laboratoire Central des Ponts et Chaussees (LCPC), Relevé des dégradations de surface des chaussées :

Méthode d'essai LPC n°38-2, Rapport. (1997) 1–44. https://www.ifsttar.fr/fileadmin/user_upload/editions/lcpc/MethodeDEssai/MethodeDEssai-LCPC-ME38_2.pdf (Accessed date: 12 November 2022).
[3] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, F.E. Alsaadi, A survey of deep neural network architectures and their applications, Neurocomputing. 234 (2017) 11–26. https://doi.org/10.1016/j.neucom.2016.12.038.
[4] M. Khanafer, S. Shirmohammadi, Applied AI in instrumentation and measurement: The deep learning revolution, Institute of Electrical and Electronics Engineers Instrumentation and Measurement Magazine. 23 (2020) 10–17. https://doi.org/10.1109/MIM.2020.9200875.
[5] R. Fedele, F.G. Praticò, R. Carotenuto, F. Giuseppe, D. Corte, Instrumented infrastructures for damage detection and management, in: Proc. 5th Institute of Electrical and Electronics Engineers International Conference on Models and Technologies for Intelligent Transportation Systems, Naples, Italy, 2017: pp. 526–531. https://doi.org/10.1109/MTITS.2017.8005729.
[6] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature. 521 (2015) 436–444. https://doi.org/doi:10.1038/nature14539.
[7] S. Pouyanfar, S. Sadiq, Y. Yan, H. Tian, Y. Tao, M.P. Reyes, M.-L. Shyu, S.-C. Chen, S.S. Iyengar, A Survey on Deep Learning: Algorithms, Techniques, and Applications, Association for Computing Machinery Computing Surveys. 51 (2019) 1–36. https://doi.org/10.1145/3234150.
[8] A. Ferrero, S. Salicone, Measurement Uncertainty, Institute of Electrical and Electronics Engineers Instrumentation and Measurement Magazine. 9 (2006) 44–51. https://doi.org/10.1109/MIM.2006.1637979.
[9] K.L. Wagstaff, Machine Learning that Matters, in: Proc. 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, 2012: pp. 529–534. https://doi.org/10.48550/arXiv.1206.4656.
[10] H. Ceylan, M.B. Bayrak, K. Gopalakrishnan, Neural networks applications in pavement engineering: A recent survey, International Journal of Pavement Research and Technology. 7 (2014) 434–444. https://doi.org/10.6135/ijprt.org.tw/2014.
[11] F. Zantalis, G. Koulouras, S. Karabetsos, D. Kandris, A Review of Machine Learning and IoT in Smart Transportation, Future Internet. 11 (2019) 94. https://doi.org/10.3390/fi11040094.
[12] C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, P. Fieguth, A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure, Advanced Engineering Informatics. 29 (2015) 196–210. https://doi.org/10.1016/j.aei.2015.01.008.
[13] A.P. Singh, A. Sharma, R. Mishra, M. Wagle, A.K. Sarkar, Pavement condition assessment using soft computing techniques, International Journal of Pavement Research and Technology. 11 (2018) 564–581. https://doi.org/10.1016/j.ijprt.2017.12.006.
[14] A. Bianchini, P. Bandini, Prediction of pavement performance through neuro-fuzzy reasoning, Computer-Aided Civil and Infrastructure Engineering. 25 (2010) 39–54. https://doi.org/10.1111/j.1467-8667.2009.00615.x.
[15] Y. Liu, J. Yao, X. Lu, R. Xie, L. Li, DeepCrack: A deep hierarchical feature learning architecture for crack segmentation, Neurocomputing. 338 (2019) 139–153. https://doi.org/10.1016/j.neucom.2019.01.036.
[16] G.H. Shafabakhsh, O.J. Ani, M. Talebsafa, Artificial neural network modeling (ANN) for predicting rutting performance of nano-modified hot-mix asphalt mixtures containing steel slag aggregates, Construction and Building Materials. 85 (2015) 136–143. https://doi.org/10.1016/j.conbuildmat.2015.03.060.
[17] K. Gopalakrishnan, Deep learning in data-driven pavement image analysis and automated distress detection: A review, Data. 3 (2018) 28. https://doi.org/10.3390/data3030028.
[18] S. Chatterjee, A.B. Brendel, S. Lichtenberg, Smart Infrastructure Monitoring: Development of a Decision Support System for Vision-Based Road Crack Detection, in: Proc. 2018 International Conference on Information Systems, San Francisco, USA, 2018: pp. 1–16. https://aisel.aisnet.org/icis2018/green/Presentations/9.
[19] H. Nhat-Duc, Q.L. Nguyen, V.D. Tran, Automatic recognition of asphalt pavement cracks using metaheuristic optimized edge detection algorithms and convolution neural network, Automation in Construction. 94 (2018) 203–213. https://doi.org/10.1016/j.autcon.2018.07.008.
[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. 157 (2017) 322–330. https://doi.org/10.1016/j.conbuildmat.2017.09.110.
[21] G. Loprencipe, A. Pantuso, A Specified Procedure for Distress Identification and Assessment for Urban Road Surfaces Based on PCI, Coatings. 7 (2017) 65. https://doi.org/10.3390/coatings7050065.
[22] S. Chatterjee, P. Saeedfar, S. Tofangchi, L. Kolbe, Intelligent Road Maintenance: a Machine Learning Approach for Surface Defect Detection, in: Proc. 26th European Conference on Information Systems,

Portsmouth, United Kingdom, 2018: pp. 1–16. https://aisel.aisnet.org/ecis2018_rp/194.
[23] G.H. Beckman, D. Polyzois, Y.J. Cha, Deep learning-based automatic volumetric damage quantification using depth camera, Automation in Construction. 99 (2019) 114–124. https://doi.org/10.1016/j.autcon.2018.12.006.
[24] A. Zhang, K.C.P. Wang, Y. Fei, Y. Liu, C. Chen, G. Yang, J.Q. Li, E. Yang, S. Qiu, Automated Pixel-Level Pavement Crack Detection on 3D Asphalt Surfaces with a Recurrent Neural Network, Computer-Aided Civil and Infrastructure Engineering. 34 (2019) 213–229. https://doi.org/10.1111/mice.12409.
[25] Z. Tong, J. Gao, Z. Han, Z. Wang, Recognition of asphalt pavement crack length using deep convolutional neural networks, Road Materials and Pavement Design. 19 (2018) 1334–1349. https://doi.org/10.1080/14680629.2017.1308265.
[26] Y.J. Cha, W. Choi, O. Büyüköztürk, Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks, Computer-Aided Civil and Infrastructure Engineering. 32 (2017) 361–378. https://doi.org/10.1111/mice.12263.
[27] N.A.M. Yusof, M.K. Osman, Z. Hussain, M.H.M. Noor, A. Ibrahim, N.M. Tahir, N.Z. Abidin, Automated Asphalt Pavement Crack Detection and Classification using Deep Convolution Neural Network, in: Proc. 9th Institute of Electrical and Electronics Engineers International Conference on Control System, Computing and Engineering, 2019, Institute of Electrical and Electronics Engineers, Penang, Malaysia, 2019: pp. 215–220. https://doi.org/10.1109/ICCSCE47578.2019.9068551.
[28] H. Oliveira, P.L. Correia, CrackIT - An image processing toolbox for crack detection and characterization, in: Proc. 2014 Institute of Electrical and Electronics Engineers International Conference on Image Processing, Institute of Electrical and Electronics Engineers, Paris, France, 2014: pp. 798–802. https://doi.org/10.1109/ICIP.2014.7025160.
[29] S. Drouyer, An "All Terrain" crack detector obtained by deep learning on available databases, Image Processing On Line. 10 (2020) 105–123. https://doi.org/10.5201/ipol.2020.282.
[30] Q. Zou, Z. Zhang, Q. Li, X. Qi, Q. Wang, S. Wang, DeepCrack: Learning hierarchical convolutional features for crack detection, Institute of Electrical and Electronics Engineers Transactions on Image Processing. 28 (2019) 1498–1512. https://doi.org/10.1109/TIP.2018.2878966.
[31] M. Eisenbach, How to Get Pavement Distress Detection Ready for Deep Learning? A Systematic Approach Motivation Why do we need Deep Learning for road condition assessment?, in: Proc. 2017 International Joint Conference on Neural Networks, Anchorage, Alaska, USA, 2017: pp. 2039–2047. https://doi.org/10.1109/IJCNN.2017.7966101.
[32] R. Stricker, M. Eisenbach, M. Sesselmann, K. Debes, H.M. Gross, Improving Visual Road Condition Assessment by Extensive Experiments on the Extended GAPs Dataset, in: Proc. 2019 International Joint Conference on Neural Networks, Institute of Electrical and Electronics Engineers, Budapest, Hungary, 2019: pp. 1–8. https://doi.org/10.1109/IJCNN.2019.8852257.
[33] L. Some, Automatic image-based road crack detection methods, Masters Thesis. (2016). http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-189245 (Accessed date: 12 January 2022).
[34] H. Majidifard, P. Jin, Y. Adu-Gyamfi, W.G. Buttlar, Pavement Image Datasets: A New Benchmark Dataset to Classify and Densify Pavement Distresses, Transportation Research Record: Journal of the Transportation Research Board. 2674 (2020) 328–339. https://doi.org/10.1177/0361198120907283.
[35] H. Majidifard, Y. Adu-Gyamfi, W.G. Buttlar, Deep machine learning approach to develop a new asphalt pavement condition index, Construction and Building Materials. 247 (2020). https://doi.org/10.1016/j.conbuildmat.2020.118513.
[36] G. Ciaparrone, A. Serra, C. Vito, P. Finelli, C.A. Scarpato, R. Tagliaferri, A Deep Learning Approach for Road Damage Classification, Advanced Multimedia and Ubiquitous Engineering. 518 (2019). https://doi.org/10.1007/978-981-13-1328-8_84.
[37] A. Geiger, P. Lenz, C. Stiller, R. Urtasun, Vision meets robotics: The KITTI dataset, International Journal of Robotics Research. 32 (2013) 1231–1237. https://doi.org/10.1177/0278364913491297.
[38] L. Zhang, F. Yang, Y.D. Zhang, Y.J. Zhu, Road crack detection using deep convolutional neural network, in: Proc. 2016 Institute of Electrical and Electronics Engineers International Conference on Image Processing, Phoenix, Arizona, USA, 2016: pp. 3708–3712. https://doi.org/10.1109/ICIP.2016.7533052.
[39] H. Maeda, Y. Sekimoto, T. Seto, T. Kashiyama, H. Omata, Road Damage Detection and Classification Using Deep Neural Networks with Smartphone Images, Computer-Aided Civil and Infrastructure Engineering. 33 (2018) 1127–1141. https://doi.org/10.1111/mice.12387.
[40] D. Arya, H. Maeda, S. Kumar Ghosh, D. Toshniwal, H. Omata, T. Kashiyama, Y. Sekimoto, Global Road Damage Detection: State-of-the-art Solutions, in: Proc. 2020 Institute of Electrical and Electronics Engineers

International Conference on Big Data (Big Data), Atlanta, Georgia, USA, 2020: pp. 5533–5539. https://doi.org/10.1109/BigData50022.2020.9377790.
[41] Y.J. Wang, M. Ding, S. Kan, S. Zhang, C. Lu, Deep Proposal and Detection Networks for Road Damage Detection and Classification, in: Proc. 2018 Institute of Electrical and Electronics Engineers International Conference on Big Data (Big Data), Institute of Electrical and Electronics Engineers, Seattle, Washington, USA, 2018: pp. 5224–5227. https://doi.org/10.1109/BigData.2018.8622599.
[42] A. Alfarrarjeh, D. Trivedi, S.H. Kim, C. Shahabi, A Deep Learning Approach for Road Damage Detection from Smartphone Images, in: Proc. 2018 Institute of Electrical and Electronics Engineers International Conference on Big Data (Big Data), 2018: pp. 5201–5204. https://doi.org/10.1109/BigData.2018.8621899.
[43] Laboratoire Central des Ponts et Chaussées (LCPC), Catalogue des dégradations de surface des chaussées: version 1998, Catalogue. (1998). http://www.ifsttar.fr/fileadmin/user_upload/editions/lcpc/MethodeDEssai/MethodeDEssai-LCPC-ME52.pdf (Accessed date: 12 January 12 2022).
[44] Tzutalin, LabelImg, (2015). https://github.com/tzutalin/labelImg. (Accessed date: 12 November).
[45] M. Everingham, S.M.A. Eslami, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman, The Pascal Visual Object Classes Challenge: A Retrospective, International Journal of Computer Vision. 111 (2014) 98–136. https://doi.org/10.1007/s11263-014-0733-5.
[46] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D.G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, X. Zheng, G. Brain, I. Osdi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D.G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, X. Zheng, TensorFlow: A System for Large-Scale Machine Learning, in: Proc. 12th USENIX Symposium on Operating Systems Design and Implementation, USENIX Association, Savannah, Georgia, USA, 2016: pp. 265–283. https://doi.org/10.48550/arXiv.1605.08695.
[47] TensorFlow, TensorFlow 1 Detection Model Zoo, (2020). https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf1_detection_zoo.md (Accessed date: 21 July 2021).
[48] S.J. Pan, Q. Yang, A survey on transfer learning, Institute of Electrical and Electronics Engineers Transactions on Knowledge and Data Engineering. 22 (2010) 1345–1359. https://doi.org/10.1109/TKDE.2009.191.
[49] A. Canziani, E. Culurciello, A. Paszke, An Analysis of Deep Neural Network Models for Practical Applications., ArXiv. (2017). https://doi.org/10.48550/arXiv.1605.07678.
[50] R. Roberts, G. Giancontieri, L. Inzerillo, G. Di Mino, Towards Low-Cost Pavement Condition Health Monitoring and Analysis Using Deep Learning, Applied Sciences. 10 (2020) 319. https://doi.org/10.3390/app10010319.
[51] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You Only Look Once: Unified, Real-Time Object Detection, in: Proc. 2016 Institute of Electrical and Electronics Engineers Conference on Computer Vision and Pattern Recognition, Institute of Electrical and Electronics Engineers, Las Vegas, Nevada, USA, 2016: pp. 779–788. https://doi.org/10.1109/CVPR.2016.91.
[52] W. Li, X.S. Feng, K. Zha, S. Li, H.S. Zhu, Summary of Target Detection Algorithms, in: Proc. International Conference on Computer Big Data and Artificial Intelligence, Changsha, China, 2021. https://doi.org/10.1088/1742-6596/1757/1/012003.
[53] Y. Li, S. Li, H. Du, L. Chen, D. Zhang, Y. Li, YOLO-ACN: Focusing on small target and occluded object detection, Institute of Electrical and Electronics Engineers Access. 8 (2020). https://doi.org/10.1109/ACCESS.2020.3046515.
[54] P. Jin, V. Rathod, X. Zhu, Pooling Pyramid Network for Object Detection, ArXiv. (2018). https://doi.org/10.48550/arXiv.1807.03284.
[55] C. Shorten, T.M. Khoshgoftaar, A survey on Image Data Augmentation for Deep Learning, Journal of Big Data. 6 (2019). https://doi.org/10.1186/s40537-019-0197-0.
[56] D. Mazzini, P. Napoletano, F. Piccoli, R. Schettini, A Novel Approach to Data Augmentation for Pavement Distress Segmentation, Computers in Industry. 121 (2020) 103225. https://doi.org/10.1016/j.compind.2020.103225.
[57] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu, A.C. Berg, SSD: Single Shot Multibox Detector, in: Proc. European Conference on Computer Vision 2016, Amsterdam, The Netherlands, 2016: pp. 21–37. https://doi.org/10.1007/978-3-319-46448-0_2.

[58] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, Institute of Electrical and Electronics Engineers Transactions on Pattern Analysis and Machine Intelligence. 39 (2017) 1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031.