A SCALABLE DEEP LEARNING FRAMEWORK FOR AUTONOMOUS ROAD ASSET CLASSIFICATION

by

Sadegh Nouri Gooshki

A thesis submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Master of Science in
Mechanical Engineering

Charlotte

2019

Approved by:

_____
Dr. Stuart Smith

_____
Dr. Hamed Tabkhi

_____
Dr. Omidreza Shoghli

_____
Dr. Amirhossein Ghasemi

ABSTRACT

SADEGH NOURI GOOSHKI. A Scalable deep learning framework for autonomous road asset classification. (Under the direction of DR. STUART SMITH and DR. HAMED TABKHI)

The focus of this thesis is on automation in road asset inspection using deep neural networks. Even though some progress has been made in automation of data collection and condition assessment, the amount of manual operation and the cost of road inspection is still considerable. Although, adding automation to the inspection process has been investigated by many studies, most of the works either are focused on traditional computer vision and classical machine learning methods with a low scalability or they have explored novel deep learning methods, but they cover a few number of road assets. Road assets are valuable components of the road infrastructure such as guardrail and pavement. In this thesis a deep learning based framework for classification of a wide range of road assets from guardrail to pavement and slope is introduced as the primary step to design a scalable system for automated road asset inspection. A Convolutional Neural Network (CNN) is used as the computational core of the model for visual analytics. Since the available dataset is challenging due to limited amount of data with high variety of visual scenes under each class, transfer learning is used to obtain the knowledge of a large scale dataset with images including similar basic features and improve the discriminative capability of the model. Accuracy is measured as the rate of correct class prediction. Results are reported for training a model with 2 to 14 classes showing the scalability of the model and 80% test accuracy for the final model with 12 classes. A comprehensive confusion and misclassification analysis is accomplished on the model outputs. Moreover, the proposed model is utilized in a hierarchical structure for designing a multi-level classification model which is able to generate two levels of class predictions to explore the possibility of road asset classification and assessment both in one integrated model.

DEDICATION

To my devoted parents

ACKNOWLEDGEMENTS

Words can not express my gratitude to Dr. Hamed Tabkhi for giving me the chance of pursuing my area of interest and providing me with valuable advice during my research. Without his supervision and support this work would not be possible.

I am also grateful to Dr. Stuart Smith for guiding me on my thesis and helping me to make this work possible.

Special thanks to Dr. Omidreza Shoghli and Dr. Adrian Burde for giving ideas on the transportation aspect of the project and their continuous support and suggestions and Dr. Amirhossein Ghasemi for his time and guidance.

I would like to acknowledge the Virginia Department of Transportation (VDOT) for sharing the image dataset. Without the road asset images, training the model would not be possible.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

Artificial neural networks are useful tools to approach many control engineering challenges such as dealing with nonlinearities, adaptation, and multivariable systems. One of the main reasons that a neural network is more capable of solving complex problems comparing to the controllers with a fixed mathematical structure is the ability of learning complexity and nonlinearities from the input data [1, 2]. With recent advances in the field of machine learning and the available computational capacity of deep learning algorithms, and large amount of data that is being generated in manufacturing, machine learning and deep learning can be utilized in manufacturing to learn from the available data to increase the level of automation [3, 4]. Also, in the field of robotics, the application of deep learning in vision tasks and autonomous platforms for robotic systems has been explored in many studies [5, 6]. In a broad context, progress in the field of machine learning facilitates expanding the scale of automation [7]. Multiple studies have demonstrated the functionality of deep learning models for automated visual inspection in different applications such as visual inspection in manufacturing facilities, tunnel condition assessment, power line inspection, and concrete crack detection [8, 9, 10, 11]. As a specific real world challenge, in this thesis adding automation to road asset inspection is studied.

1.1 Problem statement

The US interstate system construction has been accomplished and maintenance is now a primary concern for transportation programs [12]. Repair and maintenance of the U.S. infrastructure is a high cost process [13]. It is critical to maintain the highway assets and have a clear estimation of their condition [14] and road asset

inspection is an essential task for achieving an accurate condition assessment. The current inspection process involves a high volume of manual operation [15]. A smart solution to reduce the cost and manual operation is to use autonomous systems. This is also important from a safety perspective for inspection technicians. Machine learning and visual analytics have shown promising results for adding automation to the inspection systems [16, 17, 18].

Multiple studies in the literature have proved the potential efficiency of computer vision and machine learning methods for road inspection applications [19, 20, 21]. However, the current methods either have a limited scope or they rely on traditional computer vision or classical machine learning methods which are not scalable enough. There exist some recent works that have used deep learning methods for road inspection applications. But they are focusing only on one or two assets such as pavement, or traffic signs. A model that is focused on a few road asset items is not extensible enough to build a comprehensive model for road inspection. An attention to a broader range of road assets is essential for moving toward a fully automated platform.

There are many works that used one or multiple traditional methods such as morphological descriptors, segmentation, and classical machine learning methods such as Support Vector Machine[1] (SVM) for approaching problems like road surface distress and crack detection, and pavement defect detection and classification. [23, 24, 25]. A broader scale method for road asset recognition based on segmentation was presented in [26] which used semantic and geometric segmentation.

Even though the classical machine learning and traditional computer vision methods are able to provide a proof of concept showing that automation through visual analytics is feasible, achieving robust models that results a high accuracy even while inference on new unseen images is not possible without benefiting from state of the art deep learning models. Deep learning models have the ability of learning complexities

[1]Support vector machine is a machine learning model that uses hyperplanes in space for performing a task such as classification [22].

through the high computational capability of deep neural networks [27]. A category of deep learning models is Convolutional Neural Network (CNN) that is capable of learning visual information [28]. The structure of CNN is explained in section 3.1. Recurrent Neural Network (RNN) is another type of deep learning models that is able to learn the relations in data with sequential nature [29, 30]. A recent trend can be observed on using deep learning methods for road inspection tasks. Most of the presented methods have utilized convolutional neural networks for defect detection and classification. A convolutional neural network was used in [31] for road crack detection and achieved better results comparing to SVM and Boosting methods[2]. In another study, using CNNs for crack detection in pavements was explored [33]. Similar studies have been carried out for crack detection in concrete [11]. In spite of the novelty in methodology of current deep learning based frameworks, the current research does not cover a wide range of road assets.

1.2 Proposed research

The proposed research introduces Deep TRAC[3] (Deep transfer learning for road asset classification), a deep learning based classification framework which leverages transfer learning to transfer the information learned from abstractions of a large standard image dataset to elevate the discriminative power of the classifier on a challenging dataset including images of road assets with various defect types. Figure 1.1 shows how Deep TRAC framework works from data collection to inference and classification results. The proposed research uses a convolutional neural network that is pre-trained on ImageNet dataset [34]. The CNN network is retrained on an image dataset collected from road asset items in the state of Virginia which in the remaining of this work is called the road asset dataset. The Retraining is done in an optimized way to benefit from the abstractions learned from ImageNet and road asset dataset

[2]Boosting is a method based on learning weak assumptions [32].

[3]The code is available at `https://github.com/TeCSAR-UNCC/AutomatedHighwayConditionAssessment`.

at the same time. The proposed method is extensible to build an integrated model for classification and assessment of road assets at the same time. To explore this possibility, the proposed asset classification model is utilized in a hierarchical multi CNN architecture for designing a multi-level classifier which is able to generate two levels of class labels.

The experiments start with a dataset with 21890 images categorized in 14 classes and continues with modifying the arrangement through calculating a confusion matrix. The final model is a classifier trained on 12 classes followed by a binary classifier for challenging classes. The list of all classes under the road asset dataset and the process of coming up with the final 12 classes is explained in Chapter 5. After training, The model is tested on new unseen images to make sure about the generalization capability of the model. The proposed model achieves 97% training accuracy and 80% test accuracy on 12 classes.

The remaining of this thesis is as follows: Chapter 2 is on a review of related research on automation of road inspection. Chapter 3 explains the background of deep learning, CNNs, and transfer learning and it's benefits. Chapter 4 introduces the proposed Deep TRAC approach and provides a detailed explanation of it. Chapter 5 describes the experimental results and configuration for different models as well as misclassification analysis on classification results. Finally, Chapter 6 is the conclusion of this thesis.

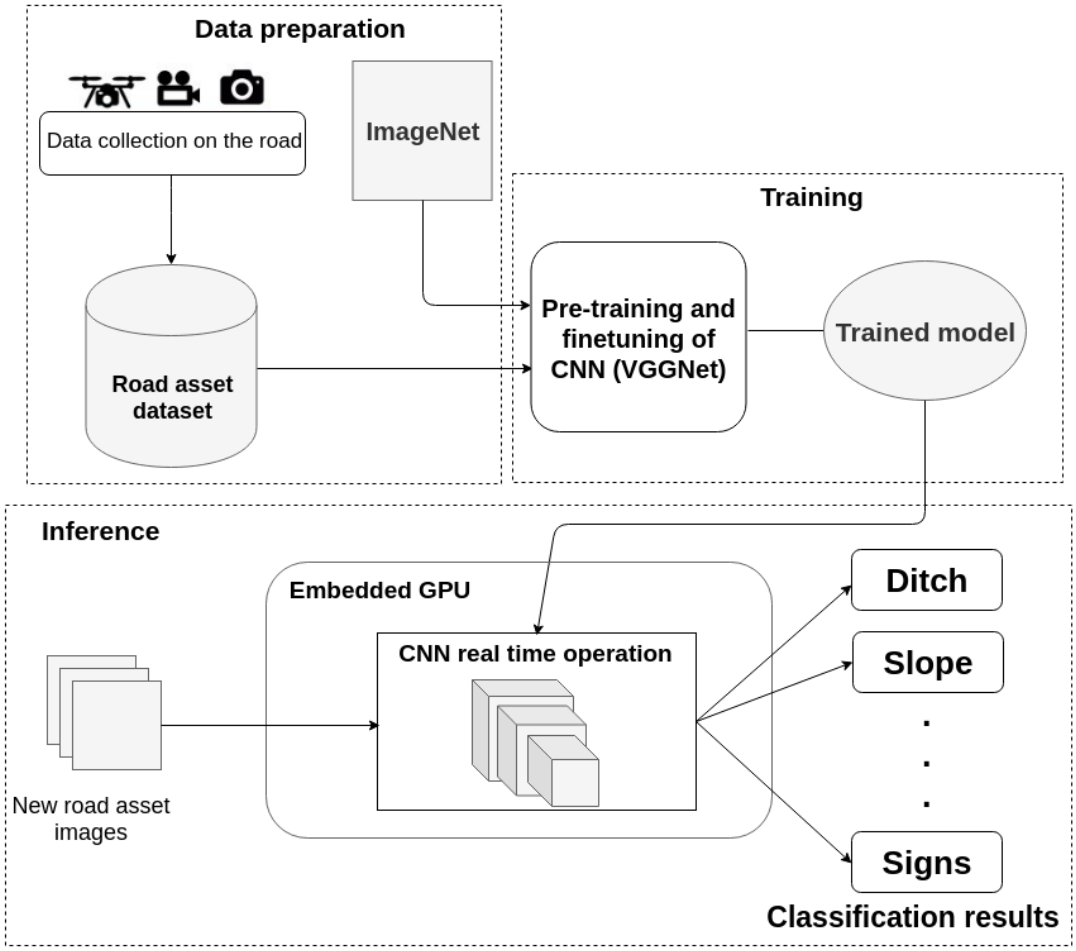


Figure 1.1: Deep TRAC framework. The proposed framework enables using transfer learning for road asset classification.

CHAPTER 2: RELATED WORK

In this section the current works that relate to the topic of this thesis are introduced with a main focus on road asset classification and defect detection. However, the possibility of using multi-level classification in road inspection is also explored. Therefore, literature is reviewed in two main sections: inspection automation in infrastructure, and multi-level classification. The related works are explained in categories and for each category a table including example papers is provided. The purpose of providing these tables is not to give an accurate evaluation of the available works. Instead, the goal is to present a general picture of the current studies. Hence, for each paper in the tables few of its presented techniques and numerical results are selected to be mentioned.

2.1 Inspection automation in infrastructure

A considerable amount of effort has been put into addressing smaller parts of the challenge of inspection automation. Since asset classification and defect detection both can be considered a classification task from a technical perspective, The literature is reviewed for both deeply. There are four general areas of focus as follows: 1- Pavement and road surface crack detection, 2- Sign detection and classification, 3- 3D Light Detection And Ranging (LiDAR) modeling and aerial photography, 4- Road asset recognition. In the remaining of section 2.1 the details of related works for the four categories mentioned above are explained. At the end, recent works on using transfer learning with CNNs for inspection applications are reviewed.

2.1.1 Pavement and road surface crack detection

In [35] a method formed by segmentation and entropy filtering and thresholding implemented for road crack detection and classification. Authors in [23] proposed a method based on segmentation and classical machine learning for road surface crack detection and classification by focusing on detection using segmentation and utilizing support vector machines (SVM) and multiple instance learning (MIL) for classification. Semantic Texton Forest (STF) was used for Pavement defect detection and classification in [36]. A platform for crack detection was designed composed of conditional texture anisotropy and neural network in [25]. Authors of [24] investigated the capability of morphological descriptors together with AdaBoost as a machine learning method for road surface crack detection. An active learning method was offered in [37] to enable training of a deep neural network with limited labeled data for surface defect detection and classification. Authors of [38] have developed a dataset for road surface damage detection and have applied a CNN-based approach for detection and classification of road surface defects. An approach for detecting cracks on road with training a deep neural network on patches of images was introduced in [39]. In [40] using convolutional neural networks with transfer learning for the task of detecting cracks in buildings was studied and capability of multiple state of the art convolutional neural networks in accomplishing this task was compared. It was observed that VGGNet and GoogleNet can achieve best scores in most cases. Many works have used deep learning methods for crack detection for civil infrastructures. A CNN based classifier for concrete crack detection was used in [11]. A convolutional neural network called DDLNet was introduced for detecting concrete defects with a specific attention to the location of defects in [41]. Table 2.1 shows samples of methods and results for [25], [24], [36], and [41] respectively.

Table 2.1: A summary of example researches on pavement and road surface crack detection

Year and author	Methodology examples	Results examples
2009, Nguyen et al.	Conditional texture anisotropy, neural network (MLPNN)	Above 90% detection success
2012, Cord et al.	Morphological descriptors, AdaBoost	A minimal error of 10.75% is gained at threshold of 0.585
2016, Radopoulou et al.	Semantic texton forest	Above 82% overall accuracy
2018, Li et al.	Deep learning, Convolutional neural network	86% localization recall

2.1.2 Sign detection and classification

In [42] methods such as histogram of oriented gradients (HOG) and SVM were implemented for detection and classification of traffic sign images. Novel convolutional neural networks were employed and trained on street view images for the task of sign detection and classification in [43]. A CNN named Multi-scale Deconvolution networks (MDN) was proposed, trained and evaluated in [44] for detecting and classifying traffic signs. Another study utilized and improved a deep learning object detection model for recognition and detection of traffic signs [45]. Table 2.2 reports the example methodologies and results for [42] and [43] respectively.

Table 2.2: A summary of example researches on sign detection and classification

Year and author	Methodology examples	Results examples
2015, Balali et al.	Histogram of oriented gradients and SVM	76.20%, 89.31%, and 94.83% accuracy for three different methods
2016, Zhu et al.	Convolutional neural network	88% accuracy and 91% recall for detection and classification

2.1.3 3D LiDAR modeling and aerial photography

Authors in [46] took the benefit of light detection and ranging (LiDAR) for 3D modeling of the ground for inspection and evaluation applications in highway management. A study Showed the possibility of using aerial photography for crack detection and

bridge joint inspection [47]. In [48] adding automation to the process of inspection based on LiDAR collected data is investigated. Table 2.3 reports a summary of [46] and [47] respectively.

Table 2.3: A summary of example researches on 3D LiDAR modeling and aerial photography

Year and author	Methodology examples	Results examples
2006, Duffel et al.	LiDAR based 3D modeling	Qualitative results were discussed.
2011, Chen et al.	Aerial photography and feature matching	Qualitative and quantitative evaluations were discussed

2.1.4 Road asset recognition

In a more comprehensive scale, a tool for road asset recognition was provided with applying semantic and geometric segmentation on road assets in video frames in [26]. Authors in [13] have done the road asset recognition task through 3D point cloud reconstruction and multiple SVMs for classification. D4AR technique, a reconstruction method based on structure from motion,along with semantic texton forest were applied in [49] to present a system for highway asset recognition. With benefiting from structure from motion to generate 3D point cloud and randomized decision forests in the process of pixel categorization, a method was introduced in [50] for highway assets recognition. Table 2.4 shows a few techniques and numerical results for [50] and [26] respectively.

Table 2.4: A summary of example researches on road asset recognition

Year and author	Methodology examples	Results examples
2012, Golparvar-Fard et al.	Structure from motion, randomized decision forests	76.50% accuracy on segmentation and 86.75% accuracy for recognition
2015, Golparvar-Fard et al.	Semantic and geometric segmentation	88.24 % classification rate and 82.02% segmentation accuracy on one dataset

2.1.5 Transfer learning with CNN for inspection

Recently some researches showed the advantage of using pre-trained networks and transfer learning on CNN-based deep learning models for inspection tasks. Pavement distress detection challenge has been approached in [51] benefiting from pre-trained VGG-16 network. Authors in [52] Utilized a pre-trained deep residual network combined with Fully Convolutional Network (FCN) for road crack detection and proposed transfer learning to overcome the data limitation. In [53] transfer learning was used on a CNN model for damage detection and classification in beam and wall. Also transfer learning was utilized in [54] for parameter initialization on an FCN-based concrete damage detection model. Table 2.5 provides example results and methods for [52] and [54] respectively.

Table 2.5: A summary of example researches on transfer learning with CNN for inspection

Year and author	Methodology examples	Results examples
2018, Bang et al.	Deep residual network with transfer learning	84.90% recall and 93.57% precision
2019, Li et al.	FCN with transfer learning	98.61% pixel accuracy and 84.53% mean intersection over union

2.2 Multi-level classification

The problem of multi-level classification can be approached in different ways. In this work when multi-level classification is mentioned, it is mainly about providing label prediction in a hierarchical fashion for categories and subcategories. However, a broader range of works in literature are reviewed to have a comprehensive understanding of the research in this area. Since classification is a general task, the methodology can vary based on application. First, some works that relate to multi-level classification in a more general scale are reviewed. Then works that are specifically related to image classification are explored.

Many of the primary works on classification of objects in a hierarchical way has been done with considering the general problem of hierarchical classification with more focus on non-visual data. To provide an easier way of searching words under different topics, authors in [55] investigated hierarchical classification of text data based on relation of words to topics. A hierarchical architecture including multiple binary classifiers is presented in [56] to explore grouping of the classes in a dataset. Considering a class taxonomy for a hierarchical multi-label structure in a dataset is studied in [57]. Multiple studies have been done on the problem of hierarchical multi-label classification and have proposed a wide range of methods to approach the problem by using techniques and tools such as kernel-based methods, decision trees, Bayesian decision theory, multi-layer perceptron (MLP), RNN, and CNN [58, 59, 60, 61, 62, 63]. In the field of fault diagnosis there are some works that presented deep learning based frameworks that include multiple deep neural networks in a hierarchical arrangement for mechanical fault diagnosis using vibration signal data [64, 65]. In [65] deep neural networks are trained separately and then all tested together in the hierarchical structure. A similar method is used in this thesis in which CNNs are trained separately but are tested altogether in a hierarchical arrangement.

2.2.1 Multi-level classification for visual tasks

In case of visual data, hierarchical structure of the datasets have been approached in few different ways. Extracting visual semantic relations and generating and learning hierarchies by using SVMs to build a hierarchical classification structure is studied in [66] and [67]. In [68], multiple SVMs are used to learn a hierarchy in data for visual tasks. Another work in this area is [69] that its focus is on utilizing the semantic information of hierarchical structure in dataset for image categorization. A hierarchical Bayesian model to learn hierarchical structure of parameter vector and visual appearance of object classes is proposed in [70] to improve the detection of objects with less examples. A study on categorizing classes in a hierarchy based on mutual

features for multi-class object detection is presented in [71]. In [72] authors have focused on learning similarity metrics in a given class taxonomy with an introduced approach based on nearest neighbors. In [73] a study on sharing the knowledge between classes by utilizing tree structure of the data is studied. The learning tasks are done with deep neural networks such as CNN and Deep Boltzmann Machine (DBM). Learning relations between labels using a hierarchical graph is studied in [74] and it is implemented as a layer on top of a CNN. Authors in [75] introduced a method to improve a trained neural network by using the knowledge that is already acquired by the network to group sets of labels based on visual similarity and learn more information with adding a new arrangement of fully connected layers to the network. Based on nearest class mean classifiers, a hierarchical approach is used in [76] to utilize available labeled data of coarse classes for improving classification accuracy of subcategories. Authors in [77] have introduced hierarchical deep CNN (HD-CNN) which is benefiting from separating the classification task for easy and challenging classes through a hierarchical CNN model to increase the focus on the classification of challenging classes in an image classification task. Instead of considering a fixed architecture, a method for learning a hierarchical model structure is presented in [78]. A model that is able to produce both superclass and subclass labels using a combination of CNN and RNNs is proposed in [79]. Applying multiple CNNs to superclasses and subclasses in a hierarchical way for designing incremental learning is studied in [80] and [81]. To provide a general view of studies on multi-level classification, table 2.6 reports the sample results for [56], [68], [77], and [79] respectively.

Even though there are many works in the literature on road inspection, and transfer learning on CNNs is explored in some recent works, to the best of my knowledge there is not still an available comprehensive work covering a wide range of road assets proposing scalable and extensible platform which can be used for automated classification of road assets. In the following chapters it is explained how the proposed

Table 2.6: A summary of example researches on Multi-level classification

Year and author	Methodology examples	Results examples
2002, Kumar et al.	Multiple binary classifiers	94.7% and 96.8% accuracy for two configurations
2011, Gao et al.	Multiple SVMs	Multiple experiments show that presented method can reduce the complexity without losing performance.
2015, Yan et al.	CNN based network (HD-CNN)	The model improves the VGG-16 accuracy on ImageNet
2018, Guo et al.	CNN and RNN	82% and 90.69% accuracy on ImageNet without and with providing coarse label

method, Deep TRAC, benefits from computational power of CNNs and optimum knowledge transfer with transfer learning to enable classifying a variety of road assets and design an expandable model for future development. Furthermore, the feasibility of using CNNs in a hierarchical structure in road asset inspection in order to generate two levels of class prediction is investigated by training Deep TRAC in a multi-level structure and comparing the results with the single level Deep TRAC classification results. Generating two levels of class prediction can enable road asset classification and assessment both in one model.

CHAPTER 3: BACKGROUND AND MOTIVATION

This chapter starts with demonstrating the advantages of deep learning comparing to traditional methods. Then the basics of CNNs and specifically VGG-16 and MobileNetV2 as the CNN models that are used is explained. Then, the basics of transfer learning will be described. Lastly, it is argued that multi-level classification is a suitable platform for road asset inspection.

Machine learning techniques have been used for a long time [82]. However, deep neural networks have not emerged until recently that computational resources for training large scale models have become available [83]. Deep learning models include a high number of parameters and are able to learn complex patterns if provided with enough data. Due to the high learning capacity of deep learning models, they are able to generalize well if provided with the right distribution of data for training and testing [84]. Two common varieties of deep neural networks are CNNs and RNNs [85]. CNNs have shown a high capability in visual tasks while RNNs are more useful for the applications with a sequential nature [86, 29, 30]. CNNs have been successful in many visual tasks such as classification, detection, and semantic segmentation [87].

3.1 Convolutional Neural Network (CNN)

Convolutional neural networks are a powerful tool for learning from visual data [88]. CNNs typically are composed of multiple convolution layers, pooling layers, and can include fully connected layers based on the application [89]. The specific layerwise structure of CNNs is designed based on animals' visual cortex computational structure [90]. Convolution layer extracts the important information from the input in a hierarchical fashion by applying specific kernel types for any target feature. Pooling

layers simply compress the information based on a mathematical operation such as averaging or maximization to provide abstractions in a more brief representation [91]. It is a challenging work to design a CNN with an optimum order of layers and optimize hyperparameters of the network.

3.2 VGGNet

VGG-16 is utilized as a powerful deep convolutional neural network for processing the visual complexity of the road asset image dataset used in this research. VGGNet is a powerful CNN architecture and the 2nd place winner of ILSVRC-2014 challenge. VGG-16 is a variation of VGGNet including 16 weight layers. In VGG-16, five convolution blocks made of convolution and pooling layers are followed by fully connected and softmax layers at the end [92]. The main design criterion of VGGNet is to use smaller kernels with ReLU activation and not using intermediate pooling layers [93]. VGGNet is flexible to use for different datasets and classification tasks by changing the fully connected and softmax configuration based on the nature of a new application. Figure 3.1 shows the blockwise architecture of VGG-16 till the last pooling layer. It can be seen that the last layers are deeper comparing to the first layers.

3.3 MobileNetV2

MobileNetV2 is a neural network which is specifically designed for mobile applications with a requirement of low power consumption. Generally, the criterion for designing neural networks for mobile applications is to reduce the number of parameters of the network without significant drop in accuracy. This way, the network can achieve the same range of accuracy (Or a satisfactory level of accuracy) while it is able to do the task with a lower power consumption which is the desirable case for mobile applications. The architecture design of MobileNetV2 has been done through optimizations and module designs such as reducing non-linearities, using inverted residuals, etc. to achieve a high accuracy while reducing the computation load. Mo-

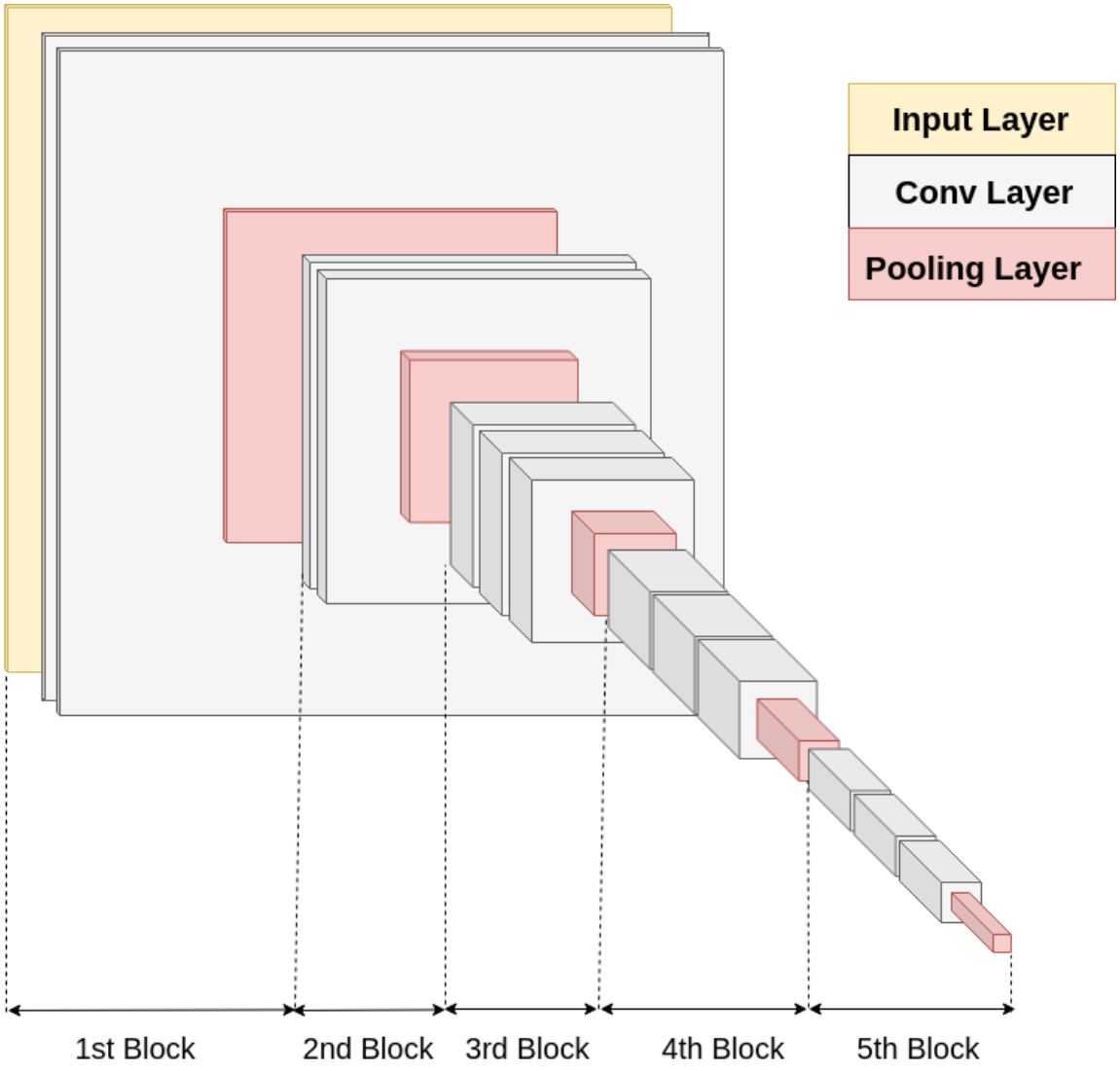


Figure 3.1: Schematic of VGG-16 convolutional blocks. Each covolutional block includes convolution and pooling layers.

bileNetV2 network is generated by designing an optimized building block and using the optimized building block multiple times with different configurations. [94, 95]

3.4 Transfer learning

In the same way that learning from previous experiences helps human to generalize new similar situations [96], deep learning models can obtain the information that has been learned in previous experiences for doing new tasks. This idea of transferring the obtained information to a new model or task is called transfer learning. Transfer

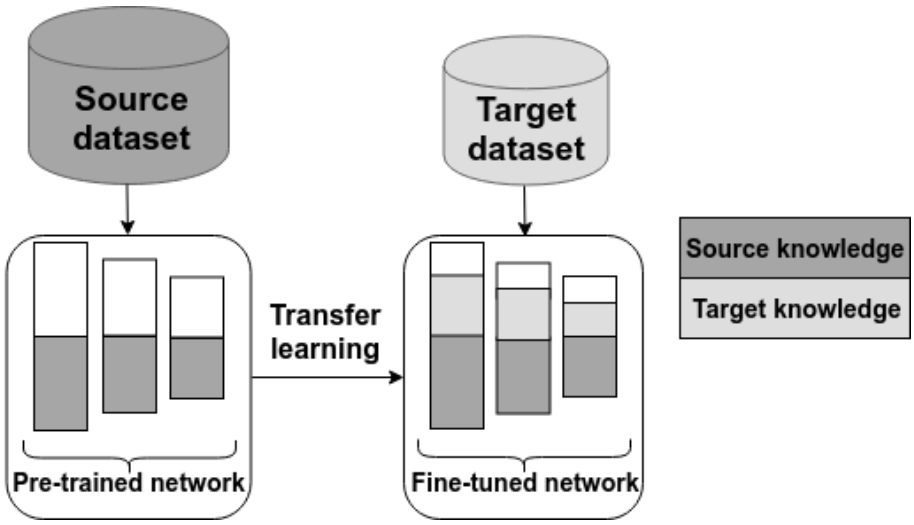


Figure 3.2: Transfer learning on CNNs. The network learns the basic knowledge of visual features from a source dataset by pre-training and will be fine-tuned on more detailed information on a target dataset.

learning has been practiced in many different ways such as weight transfer for supervised learning and policy transfer for reinforcement learning [97, 98]. One way that transfer learning can be used for CNNs is to first train a CNN on a large dataset with similar basic visual features to the target dataset for that includes data for the new task. Then retrain the network on the new dataset for the targeted application. Utilizing pre-trained networks not only reduces the training time significantly, but also enables the CNN to learn from small datasets with high sparsity [99, 100, 101]. Figure 3.2 shows how transfer learning works on CNNs.

3.5 Multi-level classification

Road asset assessment can be considered a classification task if the goal is to distinguish between different types of defects or to differentiate defected road assets from non-defected road assets. Therefore, by designing a multi-level network which is able to do two levels of classification at the same time, it is possible to provide an integrated model for classification and assessment of road assets. Figure 3.3 shows how data can be classified in multiple levels. In the first level of network all the data

is classified into a number of main classes. Then in the level two every main class is classified into subclasses.

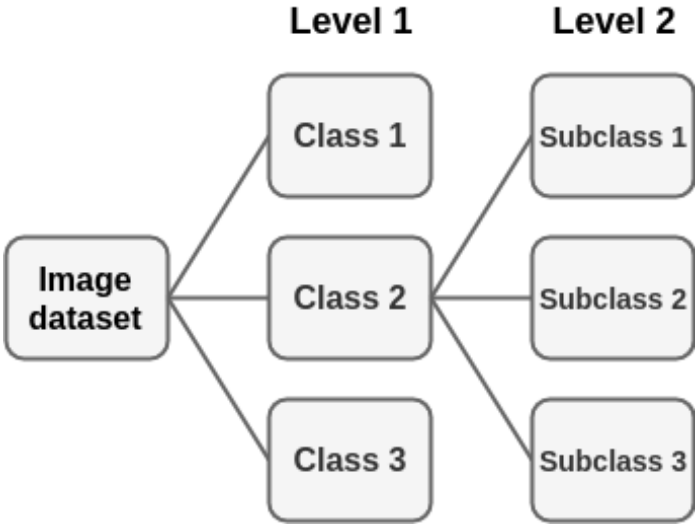


Figure 3.3: Multi-level classification of data in two levels

In chapter 2 many works are introduced that pay attention to the hierarchy among data. There are some works that specifically use CNNs in multiple levels in respect to the hierarchy of task or data structure. Arranging CNNs in multiple levels based on task hierarchy was done in [64] with vibration signal data. Also, using hierarchical structure of CNNs in a framework for incremental learning on image data is presented in [80] and [81]. A similar structure is employed to put Deep TRAC in a hierarchical CNN structure and generate two levels of labels in a way that a two level hierarchical structure of CNNs is shaped and CNNs will be trained separately, but they will be tested all at the same time as a whole system. This way, at the time of testing, data will be fed to the second level according to the prediction of the first level. Therefore, if system be trained with a proper dataset including labelled images of defected and non-defected road assets, it can be used for the inference as an integrated road asset classification and assessment system considering that it can generate two levels of labels.

CHAPTER 4: APPROACH

In this section the proposed approach, Deep TRAC, a comprehensive deep learning based model for road asset classification is introduced. The details of Deep TRAC structure and implementation, and the work flow of the whole system for training and testing is described. Furthermore, the details of using multi-level classification on Deep TRAC is discussed.

4.1 System structure

The proposed solution is established based on deep neural networks which include convolutional layers. A VGG-16 network (A variation of VGGNet) that is pre-trained on ImageNet as a large scale image dataset is used. The pre-trained VGG-16 is re-trained on the target dataset including images of road assets using transfer learning. Ideally, it is desirable to have enough number of standard images with enough similarity in the pattern of objects under each class. But there are some challenges regarding the available road asset dataset that is used for training the proposed model including but not limited to: high sparsity of the data under every class due to complexity and variety of features and defect scenarios along each class, similarity between images of different classes, partial or full occlusion of many asset items, and lack of data especially for non-defective asset items. Since the images in road asset dataset are originally taken for manual inspection purposes, all of the images are from defected road assets and it makes the classification task more challenging comparing to when all images are from non-defective road assets.

The challenges mentioned above makes the model design a more complex task. Considering these challenges, a clear representation of similar and dissimilar features

among images under each class and between different classes is needed. This way the model can mainly overcome the sparsity and data limitation challenges. An approach to achieve this aim is transfer learning. ImageNet as a base knowledge source provides valuable information regarding general features of the objects that exist in the scenes of the target dataset. This helps the model to recognize intra-class similarity and inter-class differences. The experiments with VGG-16 are performed in two different configurations. First VGG-16 is used until the last pooling layer with frozen weights for all layers as a feature extractor for the images of road assets and a fully connected classifier is trained on top of VGG-16. This architecture is called "Model 1". Multiple experiments and analysis are done with model 1 and for the final experiment with VGG-16, the VGG-16 is retrained in an optimum way on the road asset dataset. Since primary layers of VGG-16 are responsible for extracting low level features, it is preferred to keep the ImageNet weights for the primary layers of VGG-16 and retrain the last layers. Hence, during the retraining process last two convolutional blocks of VGG-16 are retrained and first three blocks are frozen since last layers are responsible for learning more high level features. The latter approach is addressed as "Model 2" in the rest of paper. Figure 4.1 shows a diagram for the training and test process of model 2 in three stages. First stage (Top) shows the pre-training process of all VGG-16 convolutional blocks on ImageNet as the source dataset.The pre-trained VGG-16 model in Keras library is used to save time and computational resources. Having the pre-trained model available, second stage (Middle) demonstrates the retraining process of last two convolutional blocks of VGG-16 on road asset images (Training set) along with training the fully connected classifier on top of VGG-16 while the weights of first three convolutional blocks are kept as ImageNet weights. After the second stage the fine-tuned model is ready to use for inference using new unseen images of road asset (Test set) which is shown in the third stage (Bottom).

A fully connected classifier is designed on top of VGG-16 convolutional blocks and

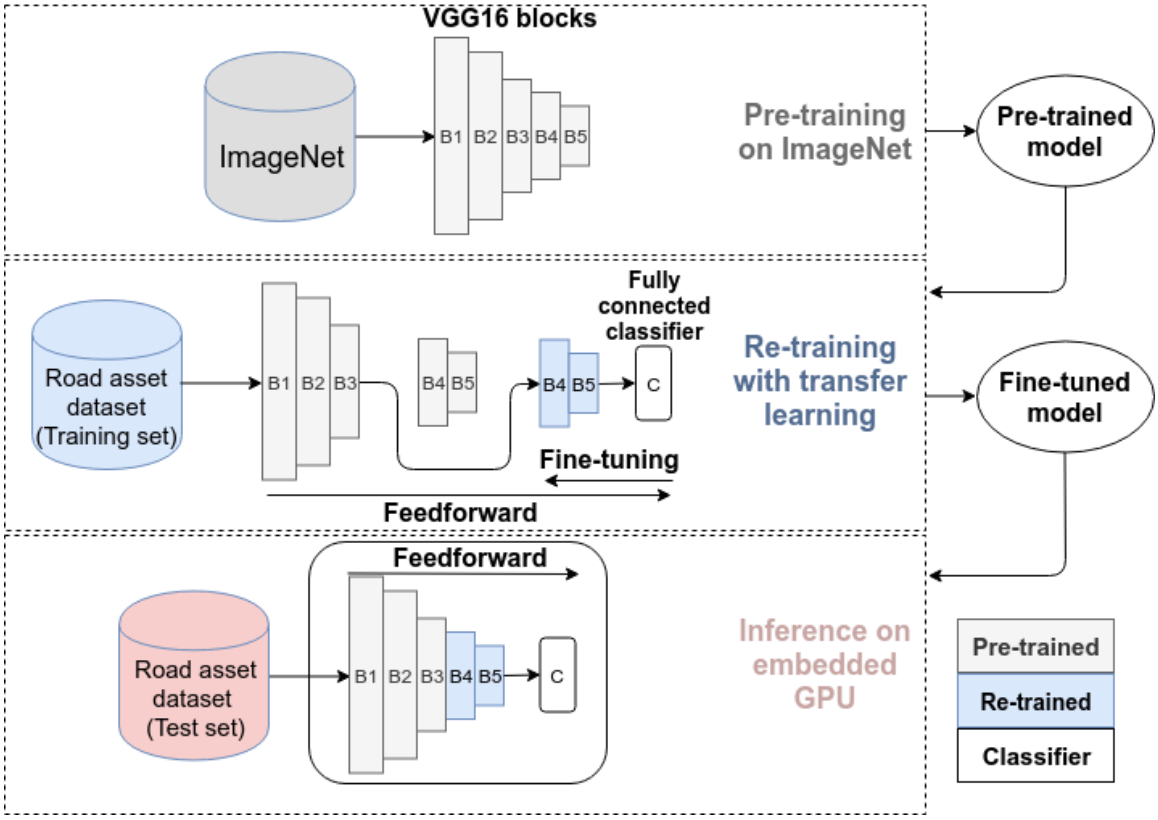


Figure 4.1: Retraining VGG-16 using transfer learning in model 2. The pre-trained VGG-16 network is retrained on road asset dataset. Last two convolutional blocks of VGG-16 are retrained while the weights of first three blocks are kept as ImageNet weights. In model 1 the weights of all convolutional blocks of VGG-16 are kept as ImageNet weights and only fully connected classifier is trained.

the output of the last pooling layer is fed to the fully connected classifier. The fully connected classifier includes a fully connected layer followed by a softmax output layer. A customized design is considered based on the current classification task. For improving the accuracy of the model on classes that have a high level of similarity, the proposed research takes the benefit of binary networks as an extra level of classification.

4.2 Binary classifier

To reduce the negative effect of classes with high inter-class confusion, using a separate classifier as the second stage of network is proposed. The process of choosing

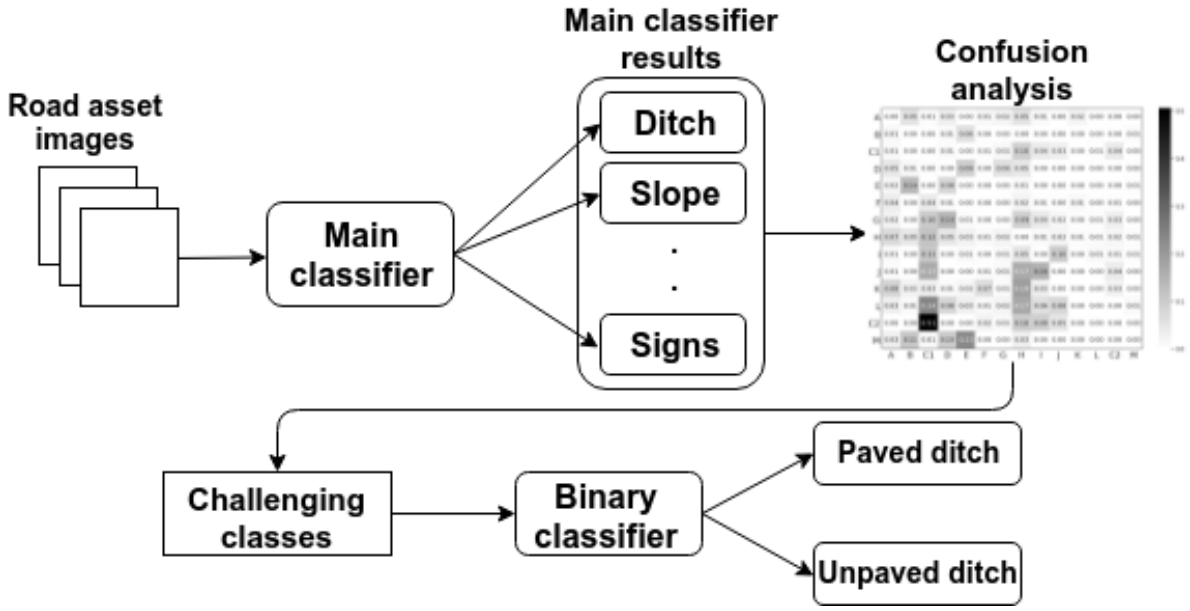


Figure 4.2: Using binary classifier for challenging classes based on confusion analysis

challenging classes for the second stage of network is shown in figure 4.2. First the main classifier is used to classify all road assets. The results of main classifier is analyzed by generating a confusion matrix. Based on confusion analysis, challenging classes are picked to be classified by a binary classifier. The dataset for main classifier is modified and the challenging classes are combined under the input dataset of main classifier. In the case of this study, since unpaved ditch has a low accuracy and high confusion rate with paved ditch, these two classes are combined as ditch in the main classifier and a binary classifier is used to classify them. It can improve the accuracy for unpaved ditch and reduce the negative effect of unpaved ditch on paved ditch and as a result an increase in paved ditch class accuracy will be gained. This happens due to the fact that binary classifier will be specialized in distinguishing between paved ditch and unpaved ditch by optimizing the network's weights only on ditch images.

4.2.1 Over sampling

Imbalanced datasets are hard to learn since the amount of available knowledge from all classes is not equal. One way to tackle the problem of imbalanced datasets is

oversampling [102]. Oversampling is used for training the binary classifier due to the imbalanced data distribution between paved ditch and unpaved ditch. Oversampling is used for the class with less data which is unpaved ditch in this case.

4.3 Real-time road asset classification

In order to assure that the proposed approach is applicable for real-time asset classification, a MobileNetV2 network that is pre-trained on ImageNet is used and retrained on the road asset dataset using transfer learning. Since MobileNetV2 is a lightweight network, it is useful for embedded platforms. As in real world applications, the final model may needs to run on an embedded system with low power supply, being able to use such a small network is essential. MobileNetV2 has much less parameters comparing to VGGNet, hence is more robust to overfitting[103]. Therefore, the whole MobileNetV2 network is retrained and no layer is frozen. Similar to the approach considered for training VGGNet, only MobileNetV2 convolutional blocks are used and the output of last pooling layer is utilized as the input of a fully connected classifier which consists of a fully connected layer followed by a softmax output layer. This model is called "Real-time asset classifier".

4.4 Multi-level road asset classification

In previous sections all the focus has been on designing the Deep TRAC framework for road asset classification and optimizing the model. In this section utilizing Deep TRAC to design a multi-level classifier framework is investigated. Figure 4.3 shows the details of the multi-level classifier. Two levels of CNNs is considered for this purpose. The first level CNN architecture is the same as road asset classifier and is responsible for receiving all of the images in a dataset and predicting the label for some defined main class labels . The second level includes multiple CNNs and each CNN is responsible for receiving the images of a specific main class based on the prediction of the CNN in first layer and extract the label for a group of subclasses.

For all of CNNs the same CNN architecture and configuration as model 2 is used except for the fully connected classifier which is modified based on number of classes for each CNN. For each CNN a VGGNet which is adapted for road asset classification in Deep TRAC with transfer learning is used in which first three convolutional blocks are frozen and last two blocks are being retrained along with the fully connected classifier. Each CNN is trained separately. The first level CNN is trained on the whole dataset and each of the second level CNNs are trained on the images of the specific main class which is supposed to be categorized into corresponding subclasses. Then all of the CNNs are put together to form the multi-level classifier for inference. During the inference phase, First an image is fed to the CNN in the first level and the main class prediction is done. Then based on the prediction of main class, the image is fed to a CNN to the second level to predict the subclass.

It is observable that if a suitable dataset of multiple road assets including labelled images of defected and non-defected road asset for each class is available, the multi-level classifier can be trained in such a way that the first level does the road asset classification and by feeding the images of each asset to a specific CNN in second level based on the prediction of first level, the asset assessment and generating label for defected versus non-defected asset or different types of defects can be done in second level. Even though such a dataset is not accessible for this research, the progress on design and evaluation is not stopped. The same dataset that is used for training and testing the proposed single-level road asset classifier is utilized in a hierarchical arrangement by grouping the classes to do a primary evaluation of the multi-level classifier. The details of the experiment is explained in Chapter 5.

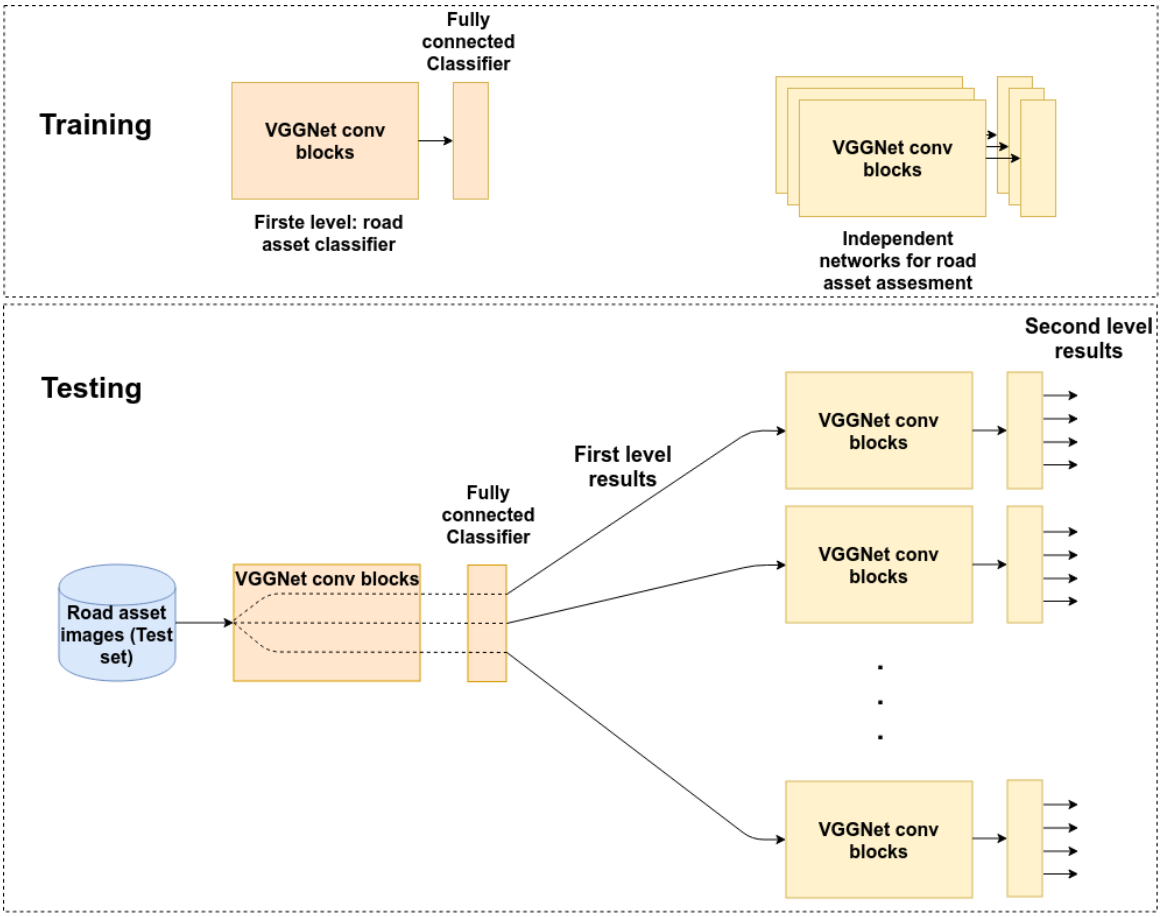


Figure 4.3: Multi-level classifier architecture includes one CNN in the first level of the network and multiple CNNs in the second level. Training is done separately and all the CNNs are tested together.

CHAPTER 5: EXPERIMENTAL RESULTS AND DISCUSSION

The proposed model is trained and tested for different number of classes to gradually build up a more inclusive network and show the scalability of the model. After coming up with the number of classes, two models with different configurations are trained on 12 classes. Then a lightweight network (MobileNetV2) is trained to evaluate the accuracy of Deep TRAC for real-time applications. To evaluate the real-time performance of Deep TRAC, the model is tested on embedded GPUs as well. Finally, the multi-level classifier is trained and tested. This section starts with explaining the experimental setup. Then the dataset and results for model 1 are explained, followed by results for confusion and misclassification analysis, model 2, real-time asset classifier, and performance analysis. At the end, the results for training and testing multi-level classifier are described.

5.1 Experimental setup

The model is implemented in Keras library with Tensorflow backend. The classifier learning rate for experiments with model 1 is 2e-4. For the experiments with Model 2, first three blocks of VGGNet are frozen (No backpropagation occures on the first three blocks) and the learning rate for last two blocks of VGG-16 and classifier is 2e-7. This number is acquired by starting with 2e-4 and decreasing the learning rate gradually. Training VGGNet layers with a learning rate of 2e-4 caused non convergence and that is because of the fact that ImageNet weights are good enough that changing them quickly causes ruining those optimal values. But a low learning rate such as 2e-7 changes the pre-trained weights gently to fine-tune the weights on the road asset dataset and adapt them based on the nature of road asset visual data

to help the model to accumulate the knowledge of new domain. For 12 classes, using the configuration of model 2 results a 5% increase in accuracy comparing to the experiment for 12 classes with model 1 (Freezing all VGGNet layers). The only point on this manner of training is a higher training time. The same learning rate as model 2 is used for real-time asset classifier and multi-level classification experiments.

All the training experiments are done on a server using an NVIDIA TITAN V GPU. Dataset is prepared by resizing the input images to the size of 224*224 and putting them under separate folders for each class since the model is implemented in a way that loads the data in a categorical way to generate the class labels. Data is fed in a batch size of 20. A categorical cross entropy loss and RMSprop optimizer are considered to train all of the models. The activation for convolutional and fully connected layer is ReLU and for the output layer is softmax. A dropout of 0.5 is applied on the fully connected layer to help it on having a generalized learning and prevent over-fitting.

The results are reported regarding the flow of work. The process of this work starts with training model 1 on 2 to 14 classes to make sure about the scalability of the platform for further developing. Then, the inter-class confusion for 14 classes is demonstrated using a confusion matrix. Based on the results of confusion matrix some modifications are done to increase the accuracy and design a more robust platform. Finally this research ends up with 12 classes and train model 2 on 12 classes as the final model. A misclassification analysis for classes with lower accuracy per class values is performed for both model 1 and model 2. Then the accuracy of Deep TRAC for real-time applications is evaluated by training MobileNetV2 as a lightweight network. Next, A performance analysis is done both on an NVIDIA TITAN V GPU and embedded GPUs. At the end, results for training and testing the multi-level classifier is reported.

5.2 Dataset

Road asset dataset includes images of road assets collected in the state of Virginia. Since the images are taken primarily for the inspection purposes, images are from defected assets. The dataset includes total number of 21890 images in 14 classes of road asset items. The proposed approach starts with experimenting on 14 classes and reducing the number of classes to 12 to optimize the model. Final model is trained on 20425 images in 12 classes that are splitted with an 80/20 ratio for training and testing. The classes under the dataset are in a wide range from slope to rigid pavement and ditch. Table 5.1 represents the details of road assets under the dataset. It is noteworthy that some road assets such as guardrail and Object markers and delineators are combined into one class due to high visual similarity of the image data for these classes.

Table 5.1: Road asset dataset statistics

Indicator	Asset item	Number of images
A	Guardrail and Object markers and delineators	3418
B	Pavement markers	3091
C1	Paved ditch	2837
D	Paved shoulder	2138
E	Flexible pavements	1709
F	Brush and Tree	1700
G	Slope	1538
H	Debris and road kill	1465
I	Under-edge drains	922
J	Small pipes and box culverts	897
K	Signs(static)	636
L	Storm drains and drop inlets	525
C2	Unpaved ditch	523
M	Rigid pavements	491

5.3 Accuracy of Model1 asset classifier

The idea of transfer learning on VGG-16 is started to be explored to find out what is a configuration that satisfies a high accuracy on the road asset dataset and enables designing an extensible model. Since model 1 has a lower training time, model 1 is used for the primary exploration and experiments. Figure 5.1 shows the results of training model 1 on 2 to 14 classes. By increasing the number of classes the accuracy slightly decreases, but keeping with low number of classes would not be favorable since a key point is to be able to include a broad range of classes. The bar graph shows that by increasing the number of classes the accuracy does not decrease significantly, thus the proposed approach is scalable for further expansion. Classes are sorted based on number of images under each class in a descending way to make sure that there is no bias due to having less number of images for the model with less classes. One contributing factor to accuracy decrease by increasing the number of classes under the model can be the fact that classes with less number of images are more challenging to learn and they are added latter.

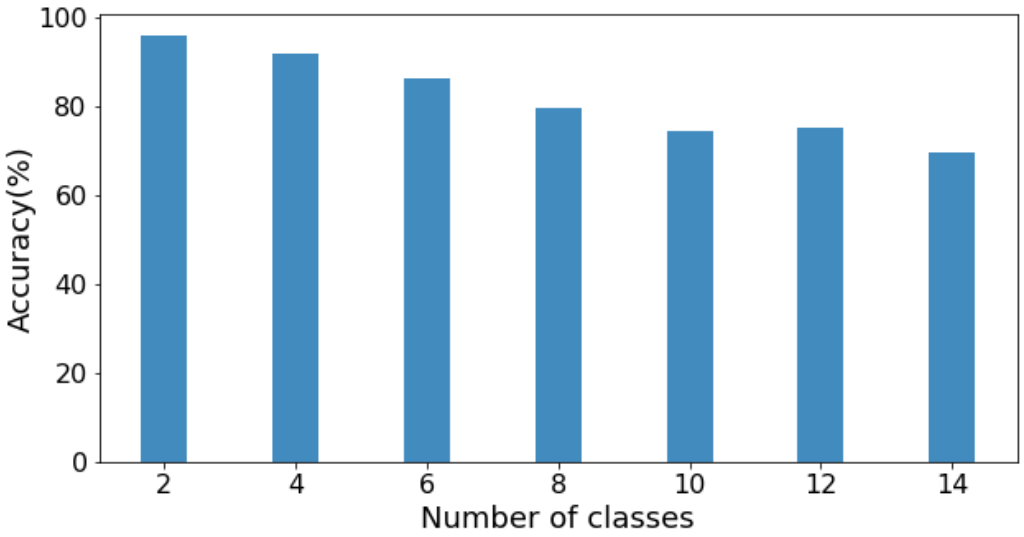


Figure 5.1: Accuracy for different number of classes. Model 1 is trained on different number of classes to show the scalability of the proposed approach.

5.4 Confusion and misclassification

To have a deep understanding of the inter-class negative effects and confusion, a confusion matrix is calculated for 14 classes. Figure 5.2 shows the confusion matrix for all 14 classes of road assets. By looking at each row it can be understood that how much a specific class is affected by other classes. Looking at a column determines how much is the negative effect of a class on the accuracy of other classes. The proposed research relies on analyzing the confusion matrix to modify the structure of model and the dataset.

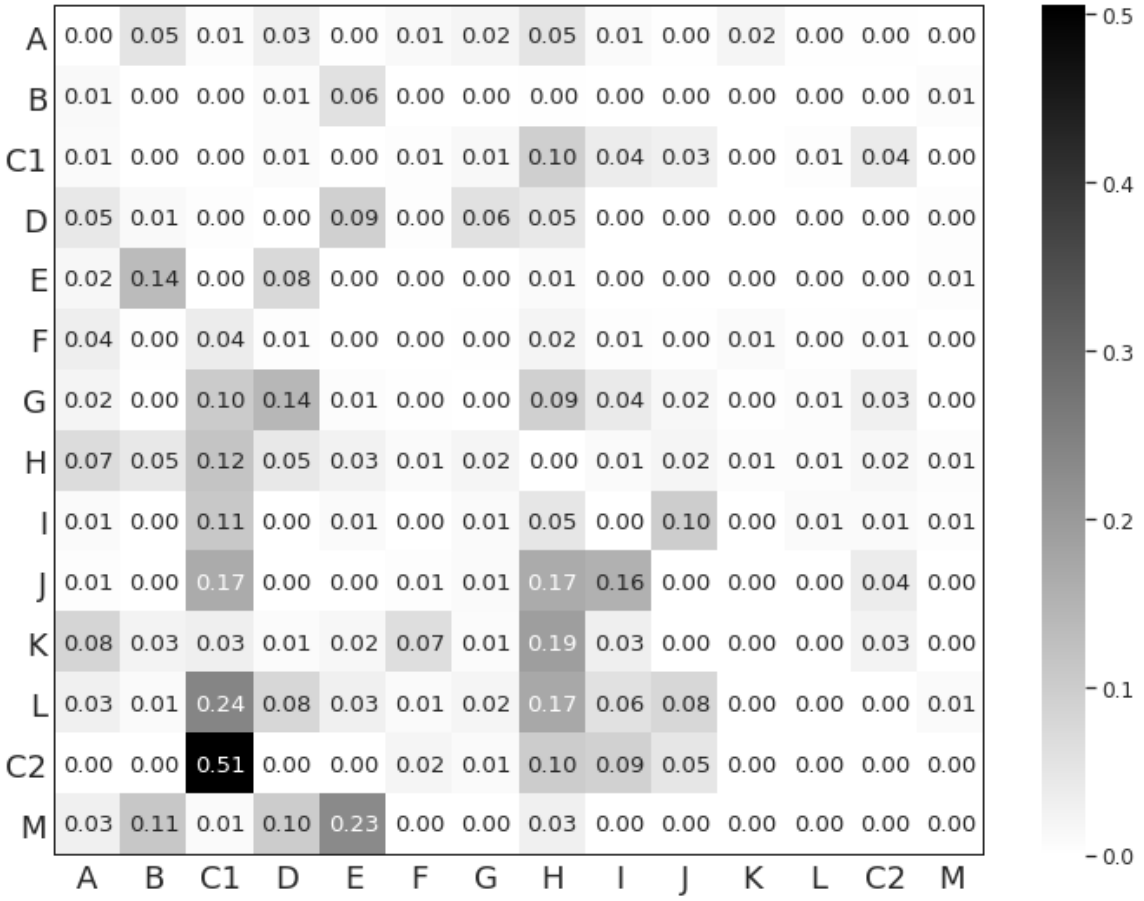


Figure 5.2: Confusion matrix for 14 classes. The confusion rate between different classes is demonstrated.

Two major conclusions can be drawn from the confusion matrix. Firstly, it is

demonstrable that the class of "debris and road kill" has a negative effect on the majority of other classes. The second clear point is the high false prediction of unpaved ditch as paved ditch. These two cases are understandable from a visual perspective. Debris and road kill can be everywhere around the road area, so the visual scenes under this class may include other road assets. In case of paved and unpaved ditch, considering that images are for defected or occluded items, these two asset can have a high visual similarity in many cases. Based on the above two arguments, two modifications are applied: 1- Debris and road kills class is taken out of the dataset to prevent decreasing the accuracy of other classes. 2- Paved ditch and unpaved ditch classes are combined as a larger class as "Ditch" and a binary classifier is used to classify these two classes separately. After applying these two modifications, the final number of classes under the main classifier would be 12. Figure 5.3 shows the training and testing accuracy diagram for model 1 on the modified dataset with 12 classes. According to final results on the modified dataset with 12 classes, training the classifier on 12 classes achieves satisfactory results. Therefore, this research does not go further with reducing the number of classes for achieving a better accuracy since the purpose is to include maximum number of road assets without significant drop in accuracy.

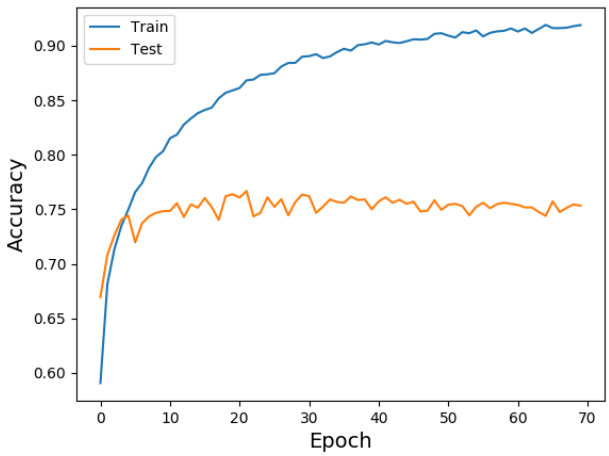
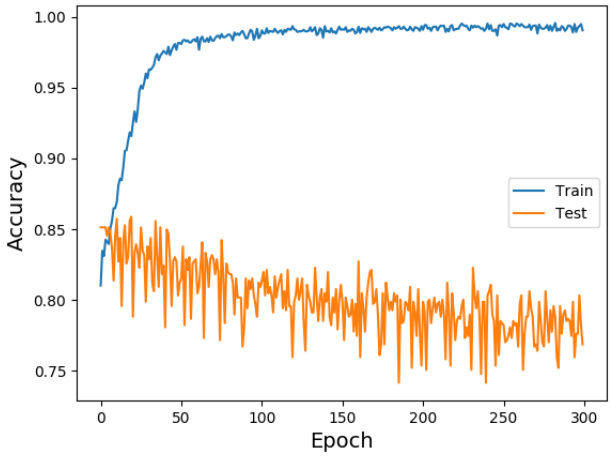
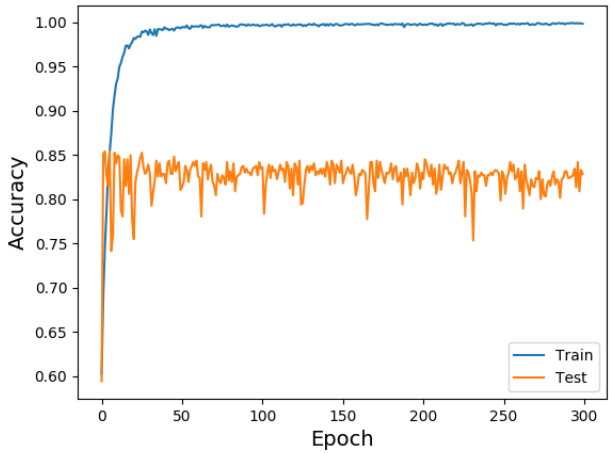


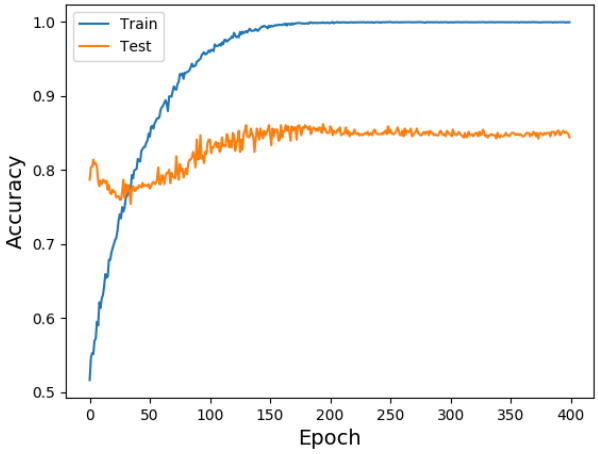
Figure 5.3: Training and testing accuracy for model 1



(a) Model 1, without oversampling



(b) Model 1, with oversampling



(c) Model 2, with oversampling

Figure 5.4: Training and test accuracy diagram for binary classifier. The final model for binary classifier is model 2 with oversampling.

To train a binary classifier for distinguishing between paved ditch and unpaved ditch, this research starts with using the same VGG-16 network as the main classifier. Figure 5.4 shows the training results for the binary classifier. First a pre-trained VGG-16 on ImageNet is used with the configuration of model 1. But due to the imbalance data distribution between the two classes, model does not converge. Oversampling technique is used to overcome the imbalanced dataset challenge and caused convergence as well as an increase in paved ditch class accuracy. A binary classifier based on model 2 is trained with oversampling and it even achieves a higher accuracy for the unpaved ditch class comparing to model 1. The main reasons for low accuracy of unpaved ditch class can be the visual nature of collected data for this class and low amount of training data for unpaved ditch. Table 5.2 shows the statistics for the dataset and training results of the binary classifier.

Table 5.2: Results for binary classifier

Parameter	Class name	
	Paved ditch	Unpaved ditch
Number of images	2837	523
Model 1 without oversampling	0.75	0.23
Model 1 with oversampling	0.91	0.23
Model 2 with oversampling	0.89	0.37

Even though overall test accuracy of a classification model shows the general ability of the model on the assigned task, but it is not enough for an in depth analysis of the model performance on every specific class. To be able to analyze the model performance in details, Figure 5.5 provides the test accuracy per class values for model 1 as well as overall test accuracy which is 75%.

Based on accuracy per class results for model 1 in figure 5.5, a misclassification analysis is done for the 4 classes with lowest accuracy to find out what are the other classes that have a negative effect on the accuracy of these 4 classes. Figure 5.6 shows the misclassification bar graphs for Under-edge drains, Slope, Storm drains and drop

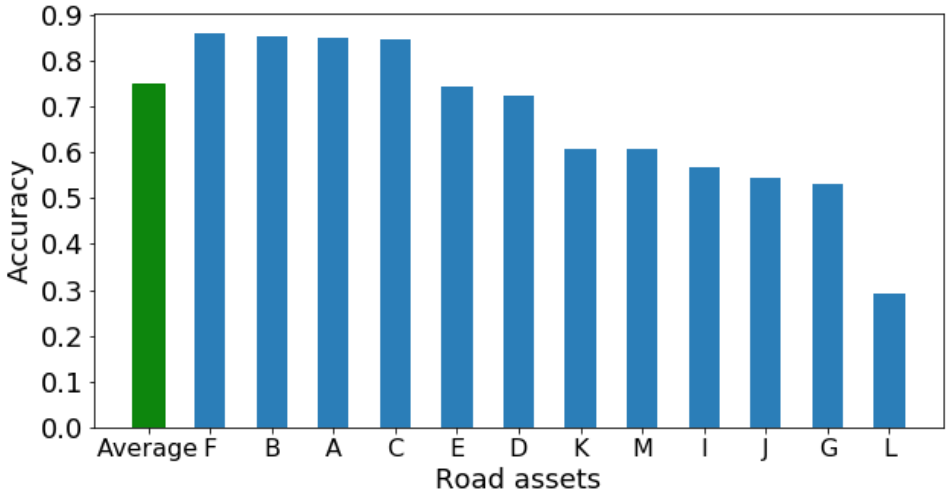


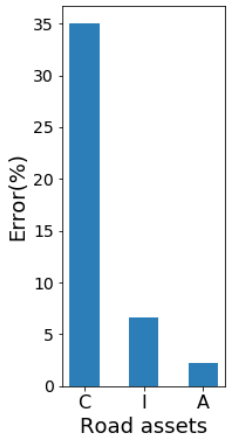
Figure 5.5: Accuracy per class for model 1. It shows how model performs on each class and provides the accuracy for each specific class which is useful for misclassification analysis.

inlets, and Small pipes and box culverts. For example, in the case of Slope as it is demonstrated in Figure 5.6b, Slope and Ditch have a high misclassification rate. High visual similarity of Slope and Ditch can be correlated with high misclassification rate of Slope images as Ditch.

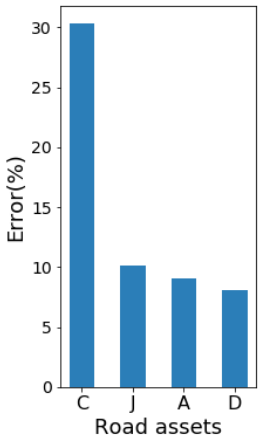
5.5 Accuracy of Model 2 classifier

The experiments show that retraining last two blocks of VGG-16 can improve the accuracy of the model. Figure 5.7 shows training and testing accuracy graph for model 2. Model 2 is trained on 12 classes with a comparatively lower learning rate than model 1 (2e-7) and retraining last two blocks of VGG-16 and resulted 5% increase in accuracy comparing to model 1. Hence, model 2 is able to achieve 80% accuracy on 12 classes.

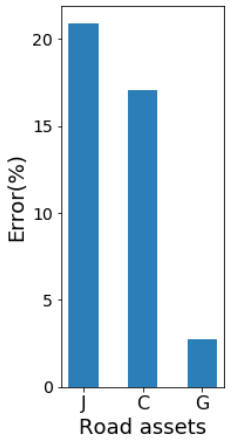
Figure 5.8 compares accuracy per class and overall accuracy for model 1 and model 2. It can be seen that accuracy per class for all road assets is higher for model 2. In the same way as was done for model 1, a misclassification analysis for 4 classes with lowest accuracy is accomplished for model 2. In case of model 2, classes with lowest



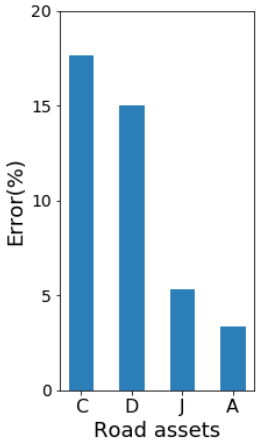
(a) Under-edge drains



(b) Slope



(c) Storm drains and drop inlets



(d) Small pipes and box culverts

Figure 5.6: Misclassification analysis for model 1. The graph shows the misclassification analysis results for classes with lower accuracy with model 1.

accuracy are rigid pavements, slope, storm drains and drop inlets, and small pipes and box culverts. Figure 5.9 shows the bar graphs for misclassification analysis of the above classes. As an instance, the high misclassification rate of Ditch, and Paved shoulder as Slope shows how challenging is the classification of these road assets.

In order to provide a detailed numerical understanding of the negative effect of each class on other classes, confusion matrix for model 2 is calculated. Figure 5.10 shows the confusion matrix for model 2. It helps to demonstrate the confusion and misclassification patterns along different asset items.

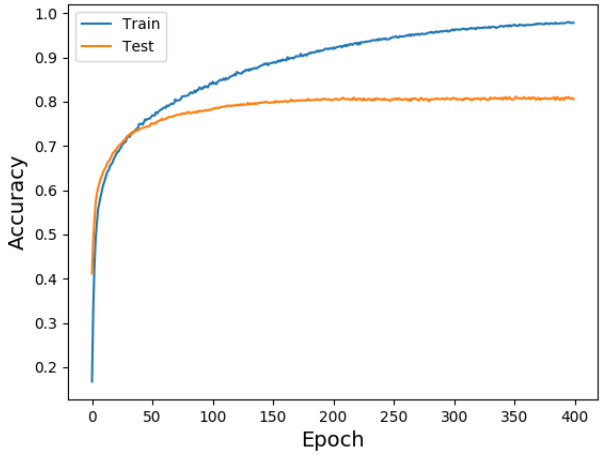


Figure 5.7: Training and testing accuracy for model 2

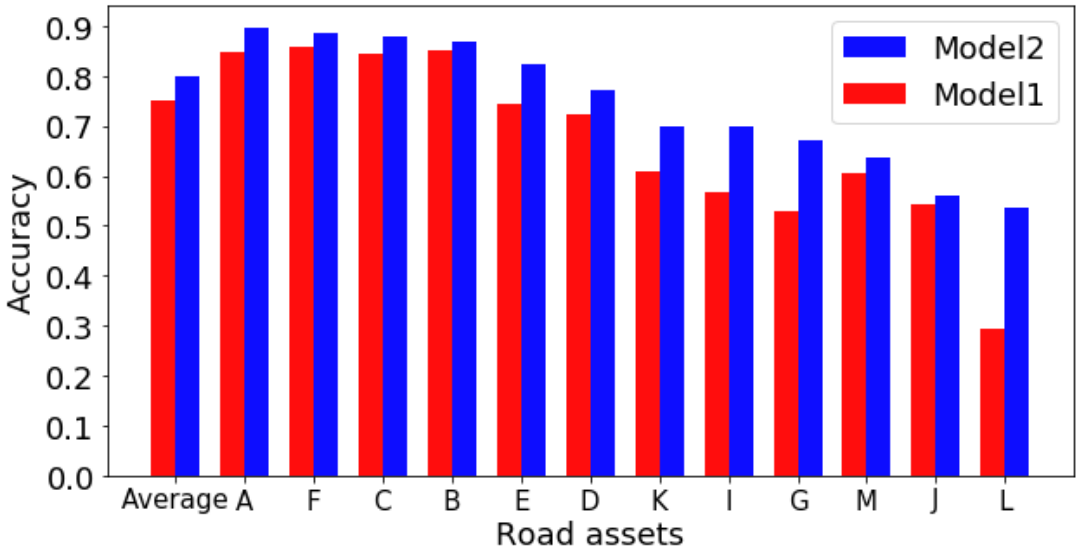
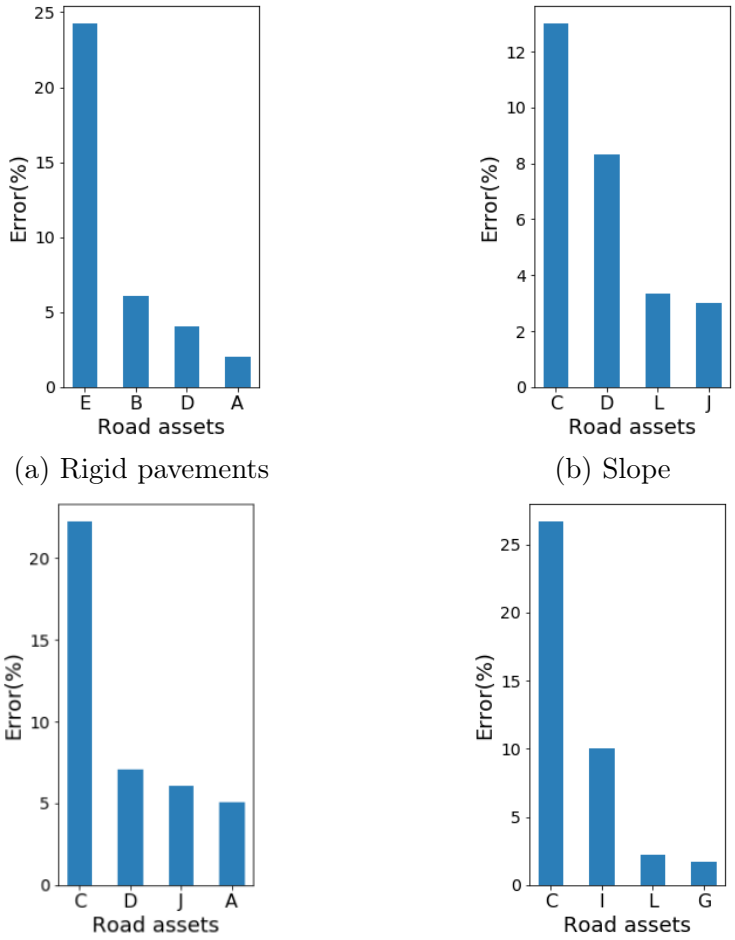


Figure 5.8: Model 1 and model 2 per class accuracy comparison

5.6 Accuracy of real-time asset classifier

MobileNetV2 is retrained on 12 classes. The goal is to show that the proposed approach is practicable for real-time usage. All the layers of MobileNetV2 are retrained on road asset dataset. A learning rate of 2e-7 is used to train the real-time asset classifier. The network width multiplier (alpha) and depth multiplier parameters in Keras are considered as one. The results for retraining MobileNetV2 on road asset



(a) Rigid pavements (b) Slope

(c) Storm drains and drop inlets (d) Small pipes and box culverts

Figure 5.9: Misclassification analysis for model 2. The analysis is done for classes with lower accuracy with model 2.

dataset images show an 81% accuracy which is in the same range of the accuracy achieved by Model 2. This proves that Deep TRAC can be applied in practice with small size networks with low computation. Moreover, Deep TRAC can be implemented with different neural networks. Figure 5.11 shows the training and testing accuracy diagram for real-time asset classifier.

5.7 Performance analysis of road asset classification

A performance analysis is done for measuring the training and testing time and power consumption. These values help to understand computation cost of the model as well as performance of the model in real-time applications. Table 5.3 shows the

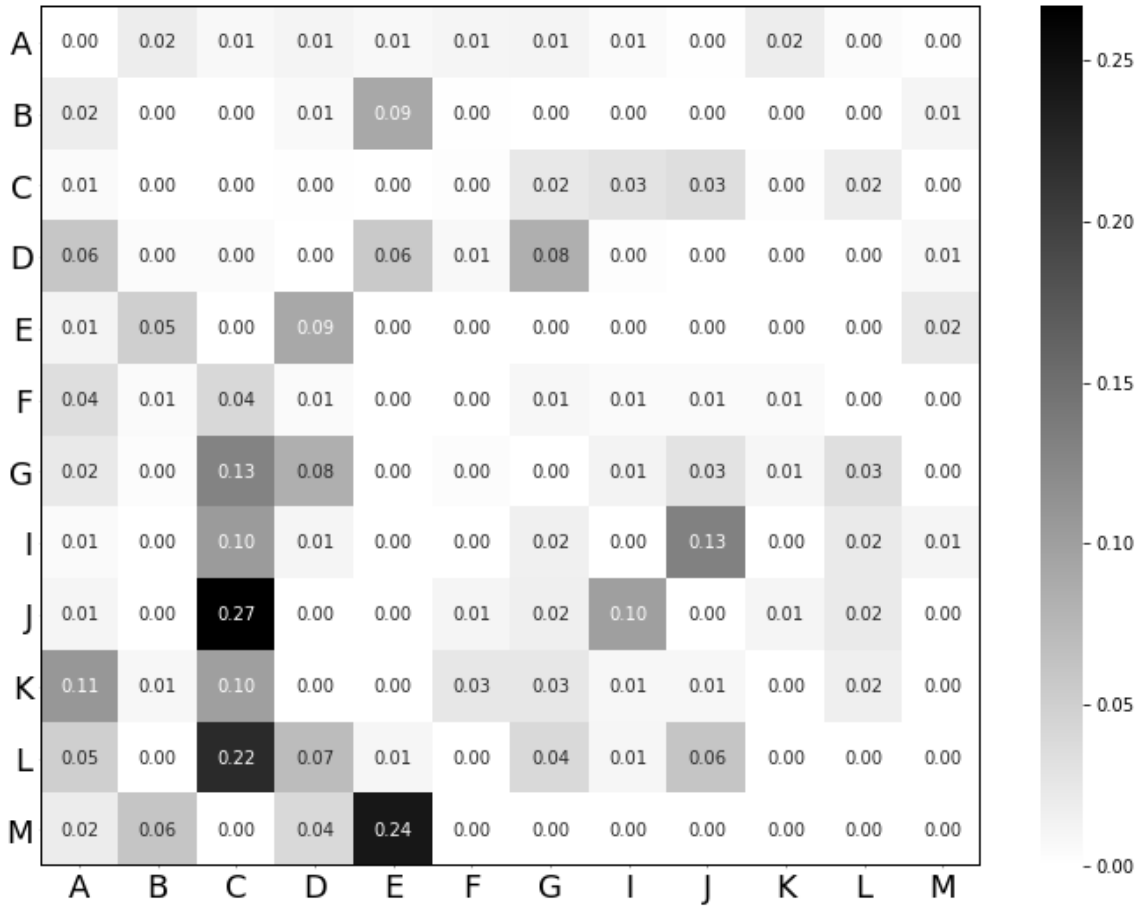


Figure 5.10: Confusion matrix for model 2

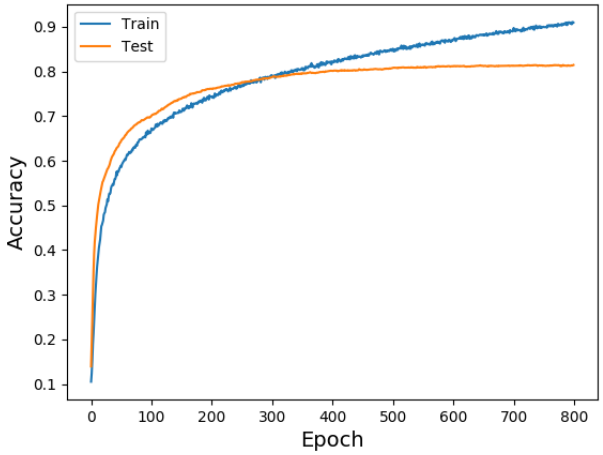


Figure 5.11: Training and testing accuracy for real-time asset classifier

overall training and testing time for model 1 and model 2 on an NVIDIA TITAN V GPU. The training time for model 2 is higher than model 1 since on model 2 last two

blocks of VGG16 were being retrained while in model 1 VGG16 is used as a feature extractor and the backpropagation process was performed only on the fully connected classifier.

Table 5.3: Performance analysis on NVIDIA TITAN V GPU

Measurement	**Model**	
	Model 1	**Model 2**
Overall training and testing time	0:10:26	9:12:29
Inference time(fps)	121	125

5.7.1 Real-time performance on mobile devices

NVIDIA Jetson TX2 and AGX Xavier embedded GPUs were used to measure the real time performance of the model. Model 2 is used as the final model for these experiments. Inference time and power consumption metrics are considered to measure the real-time performance. The results are presented in table 5.4.

Table 5.4: Inference time and power consumption on embedded GPUs

Model	VGG-16		MobileNetV2	
	TX2	Xavier	TX2	Xavier
Inference time (fps)	10.29	30.29	23.77	65.02
Power (W)	9.50	19.12	4.73	6.83

It is noteworthy that to be able to run VGG-16 on Jetson TX2 the GPU memory usage is limited to 50%. The same memory usage is considered for running MobileNetV2 on TX2 to have a fair comparison.

5.8 Accuracy of multi-level classification

The configuration for all CNNs in multi-level classifier is similar to model 2. Since a suitable dataset including labelled images for defected and non-defected road asset items is not accessible for this research, the road asset dataset that is used for training Deep TRAC is adapted by grouping the 12 road assets into three main classes based on visual similarity of road assets. Figure 5.12 shows the structure of the rearranged

road asset dataset which is used for training and testing the multi-level classifier. Each main class includes four subclasses.

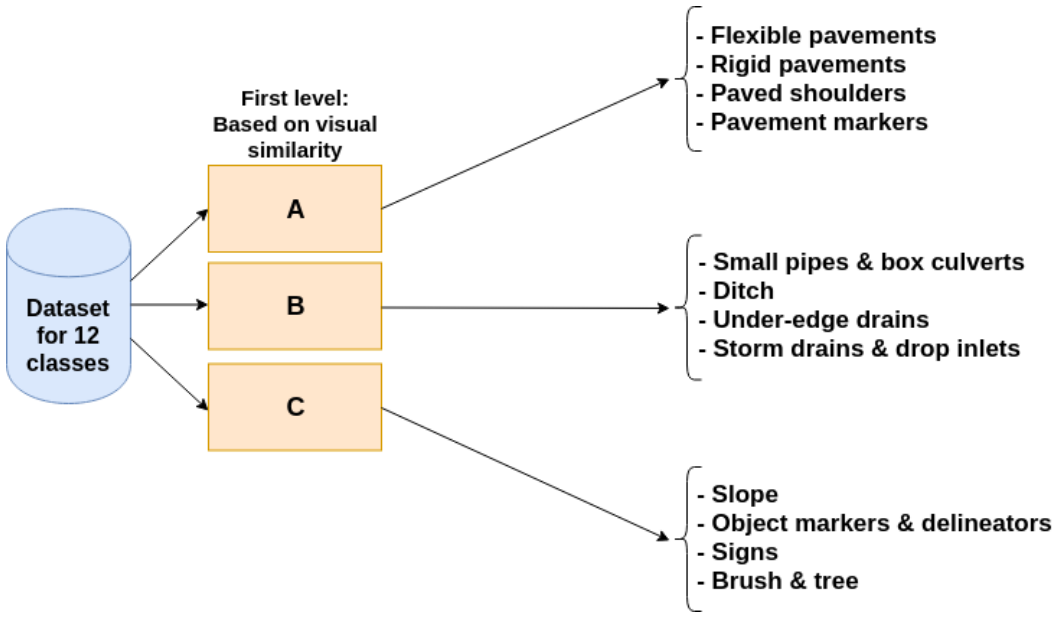


Figure 5.12: dataset arrangement for training and testing the multi-level classifier

The first level CNN is trained on the whole dataset which is categorized into three main classes A, B, and C. Three CNNs are used in second level. Each CNN is trained on the corresponding data for one of the three classes including four subclasses. Once the training is done separately for all four CNNs, all are put together to generate the multi-level structure for inference. The main goal is to evaluate the accuracy of system while generating two levels of labels without providing any intermediate label. During the inference phase, the first level CNN receives an image and predicts a main class label A, B, or C for the image. Once the first level prediction is done, based on the predicted label, the image is fed to the responsible CNN in the second layer for that specific main class and a subclass label is predicted for the image. Table 5.5 shows the results for first and second level test accuracy compared to the single level classifier (model 2) test accuracy.

The criterion that is considered to evaluate multi-level classifier is to achieve the

Table 5.5: Results of multi-level classifier

Model	Accuracy (%)
First level	91
Overall accuracy (Second level)	79
Single level (Model 2)	80

same level of accuracy as the single level classifier while predicting the same class labels. This way a multi-level classifier provides two levels of labels while keeping the same range of accuracy for the final subclass label prediction. Comparing the accuracy of second level with the accuracy of single level Deep TRAC (model 2) for predicting the label of 12 classes under the road asset dataset shows that multi-level classifier is able to achieve the same range of accuracy and predict the class labels for both first level and second level at the same time. In addition, the accuracy of predicting main class labels in the first level is 91%.

CHAPTER 6: CONCLUSIONS

A deep learning based approach for road asset classification is introduced in this thesis. Transfer learning has been applied on pre-trained CNNs to utilize the knowledge of a large dataset including general objects for learning from a challenging dataset. The results show 80% accuracy for training VGG-16 on 12 classes with transfer learning (Model 2). It is shown that the model is scalable by training the model on subsets of road asset dataset with different number of classes from 2 to 14. A detailed misclassification analysis has been done to explore inter-class confusion between different road assets. It demonstrates how important is the effect of visual similarity of different road assets on making the classification task challenging.

To ensure that Deep TRAC is applicable for using on embedded GPUs, MobileNetV2, a state of the art neural network that is designed for mobile applications, trained on road asset dataset using transfer learning and resulted 81% accuracy that is in the same range as model 2 accuracy showing that the proposed approach can be applied for real-time asset classification. Real-time performance analysis is accomplished on Deep TRAC with both CNN networks (VGGNet and MobileNetV2) by running the models for inference on embedded GPUs. According to the reported results for the extensive experiments, Deep TRAC is a major step in using deep learning for adding automation to the road inspection process by providing a scalable model for classification of a broad range of road assets. In addition, the results of implementing Deep TRAC in a multi-level classification structure shows that the proposed framework can be used for designing an integrated road asset classification and assessment system.

REFERENCES

[1] K. J. Hunt, D. Sbarbaro, R. Żbikowski, and P. J. Gawthrop, "Neural networks for control systems – a survey," *Automatica*, vol. 28, no. 6, pp. 1083–1112, 1992.

[2] D. Psaltis, A. Sideris, and A. A. Yamamura, "A multilayered neural network controller," *IEEE control systems magazine*, vol. 8, no. 2, pp. 17–21, 1988.

[3] T. Wuest, D. Weimer, C. Irgens, and K.-D. Thoben, "Machine learning in manufacturing: advantages, challenges, and applications," *Production & Manufacturing Research*, vol. 4, no. 1, pp. 23–45, 2016.

[4] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, "Deep learning for smart manufacturing: Methods and applications," *Journal of Manufacturing Systems*, vol. 48, pp. 144–156, 2018.

[5] J. Ruiz-del Solar, P. Loncomilla, and N. Soto, "A survey on deep learning methods for robot vision," *arXiv preprint arXiv:1803.10862*, 2018.

[6] J. Shabbir and T. Anwer, "A survey of deep learning techniques for mobile robot applications," *arXiv preprint arXiv:1803.07608*, 2018.

[7] M. J. Liberatore, "Automation, AI and OR: in search of the synergy and publication priorities," *European journal of operational research*, vol. 99, no. 2, pp. 248–255, 1997.

[8] A. Luckow, M. Cook, N. Ashcraft, E. Weill, E. Djerekarov, and B. Vorster, "Deep learning in the automotive industry: Applications and tools," in *2016 IEEE International Conference on Big Data (Big Data)*, pp. 3759–3768, IEEE, 2016.

[9] E. Protopapadakis, C. Stentoumis, N. Doulamis, A. Doulamis, K. Loupos, K. Makantasis, G. Kopsiaftis, and A. Amditis, "Autonomous robotic inspection in tunnels.," *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, vol. 3, no. 5, 2016.

[10] V. N. Nguyen, R. Jenssen, and D. Roverso, "Automatic autonomous vision-based power line inspection: A review of current status and the potential role of deep learning," *International Journal of Electrical Power & Energy Systems*, vol. 99, pp. 107–120, 2018.

[11] Y.-J. Cha, W. Choi, and O. Büyüköztürk, "Deep learning-based crack damage detection using convolutional neural networks," *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, no. 5, pp. 361–378, 2017.

[12] M. E. Ozbek, J. M. de la Garza, and K. Triantis, "Data and modeling issues faced during the efficiency measurement of road maintenance using data envelopment analysis," *Journal of Infrastructure Systems*, vol. 16, no. 1, pp. 21–30, 2010.

[13] V. Balali, M. Golparvar-Fard, and J. M. de la Garza, "Video-based highway asset recognition and 3d localization," in *Computing in Civil Engineering (2013)*, pp. 379–386, 2013.

[14] T. R. Board, E. National Academies of Sciences, and Medicine, *Critical Issues in Transportation 2019*. Washington, DC: The National Academies Press, 2018.

[15] T. Siriborvornratanakul, "An automatic road distress visual inspection system using an onboard in-car camera," *Advances in Multimedia*, vol. 2018, 2018.

[16] S. Ravikumar, K. Ramachandran, and V. Sugumaran, "Machine learning approach for automated visual inspection of machine components," *Expert systems with applications*, vol. 38, no. 4, pp. 3260–3266, 2011.

[17] M.-D. Yang and T.-C. Su, "Automated diagnosis of sewer pipe defects based on machine learning approaches," *Expert Systems with Applications*, vol. 35, no. 3, pp. 1327–1337, 2008.

[18] S.-H. Huang and Y.-C. Pan, "Automated visual inspection in the semiconductor industry: A survey," *Computers in industry*, vol. 66, pp. 1–10, 2015.

[19] S. Chambon and J.-M. Moliard, "Automatic road pavement assessment with image processing: review and comparison," *International Journal of Geophysics*, vol. 2011, 2011.

[20] E. Salari and G. Bao, "Automated pavement distress inspection based on 2d and 3d information," in *2011 IEEE INTERNATIONAL CONFERENCE ON ELECTRO/INFORMATION TECHNOLOGY*, pp. 1–4, IEEE, 2011.

[21] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition," *Neural networks*, vol. 32, pp. 323–332, 2012.

[22] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[23] S. Varadharajan, S. Jose, K. Sharma, L. Wander, and C. Mertz, "Vision for road inspection," in *Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on*, pp. 115–122, IEEE, 2014.

[24] A. Cord and S. Chambon, "Automatic road defect detection by textural pattern recognition based on adaboost," *Computer-Aided Civil and Infrastructure Engineering*, vol. 27, no. 4, pp. 244–259, 2012.

[25] T. S. Nguyen, M. Avila, and S. Begot, "Automatic detection and classification of defect on road pavement using anisotropy measure," in *2009 17th European Signal Processing Conference*, pp. 617–621, Aug 2009.

[26] V. Balali and M. Golparvar-Fard, "Segmentation and recognition of roadway assets from car-mounted camera video streams using a scalable non-parametric image parsing method," *Automation in construction*, vol. 49, pp. 27–39, 2015.

[27] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.

[28] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, *et al.*, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[29] I. Sutskever, J. Martens, and G. E. Hinton, "Generating text with recurrent neural networks," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 1017–1024, 2011.

[30] Z. Che, S. Purushotham, K. Cho, D. Sontag, and Y. Liu, "Recurrent neural networks for multivariate time series with missing values," *Scientific reports*, vol. 8, no. 1, p. 6085, 2018.

[31] L. Zhang, F. Yang, Y. Daniel Zhang, and Y. J. Zhu, "Road crack detection using deep convolutional neural network," in *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 3708–3712, Sep. 2016.

[32] R. E. Schapire, Y. Freund, *et al.*, "A short introduction to boosting," *Journal of Japanese Society for Artificial Intelligence*, vol. 14, no. 5, pp. 771–780, 1999.

[33] L. Pauly, D. Hogg, R. Fuentes, and H. Peel, "Deeper networks for pavement crack detection," in *Proceedings of the 34th ISARC*, pp. 479–485, IAARC, 2017.

[34] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 248–255, Ieee, 2009.

[35] H. Oliveira and P. L. Correia, "Automatic road crack segmentation using entropy and image dynamic thresholding," in *Signal Processing Conference, 2009 17th European*, pp. 622–626, IEEE, 2009.

[36] S. C. Radopoulou and I. Brilakis, "Automated detection of multiple pavement defects," *Journal of Computing in Civil Engineering*, vol. 31, no. 2, p. 04016057, 2016.

[37] C. Feng, M.-Y. Liu, C.-C. Kao, and T.-Y. Lee, "Deep active learning for civil infrastructure defect detection and classification," in *Computing in Civil Engineering 2017*, pp. 298–306, 2017.

[38] H. Maeda, Y. Sekimoto, T. Seto, T. Kashiyama, and H. Omata, "Road damage detection and classification using deep neural networks with smartphone images," *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 12, pp. 1127–1141, 2018.

[39] S. Park, S. Bang, H. Kim, and H. Kim, "Patch-based crack detection in black box road images using deep learning," in *35th International Symposium on Automation and Robotics in Construction and International AEC/FM Hackathon: The Future of Building Things, ISARC 2018*, 2018.

[40] Ç. F. Özgenel and A. G. Sorguç, "Performance comparison of pretrained convolutional neural networks on crack detection in buildings," in *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, vol. 35, pp. 1–8, IAARC Publications, 2018.

[41] R. Li, Y. Yuan, W. Zhang, and Y. Yuan, "Unified vision-based methodology for simultaneous concrete defect detection and geolocalization," *Computer-Aided Civil and Infrastructure Engineering*, 2018.

[42] V. Balali and M. Golparvar-Fard, "Evaluation of multiclass traffic sign detection and classification methods for US roadway asset inventory management," *Journal of Computing in Civil Engineering*, vol. 30, no. 2, p. 04015022, 2015.

[43] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu, "Traffic-sign detection and classification in the wild," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[44] S. Pei, F. Tang, Y. Ji, J. Fan, and Z. Ning, "Localized traffic sign detection with multi-scale deconvolution networks," *CoRR*, vol. abs/1804.10428, 2018.

[45] D. Tabernik and D. Skočaj, "Deep learning for large-scale traffic-sign detection and recognition," *arXiv preprint arXiv:1904.00649*, 2019.

[46] C. Duffell, D. Rudrum, and M. Willis, "Detection of slope instability using 3d lidar modelling," in *GeoCongress 2006: Geotechnical Engineering in the Information Technology Age*, pp. 1–5, 2006.

[47] S.-E. Chen, C. Rice, C. Boyle, and E. Hauser, "Small-format aerial photography for highway-bridge monitoring," *Journal of Performance of Constructed Facilities*, vol. 25, no. 2, pp. 105–112, 2011.

[48] H. Zhang, J. Li, M. Cheng, and C. Wang, "Rapid inspection of pavement markings using mobile lidar point clouds.," *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, vol. 41, 2016.

[49] B. Uslu, M. Golparvar-Fard, and J. M. de la Garza, "Image-based 3d reconstruction and recognition for enhanced highway condition assessment," in *Computing in Civil Engineering (2011)*, pp. 67–76, 2011.

[50] M. Golparvar-Fard, V. Balali, and J. M. de la Garza, "Segmentation and recognition of highway assets using image-based 3d point clouds and semantic texton forests," *Journal of Computing in Civil Engineering*, vol. 29, no. 1, p. 04014023, 2012.

[51] K. Gopalakrishnan, S. K. Khaitan, A. Choudhary, and A. Agrawal, "Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection," *Construction and Building Materials*, vol. 157, pp. 322–330, 2017.

[52] S. Bang, S. Park, H. Kim, H. Kim, *et al.*, "A deep residual network with transfer learning for pixel-level road crack detection," in *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, vol. 35, pp. 1–4, IAARC Publications, 2018.

[53] Y. Gao and K. M. Mosalam, "Deep transfer learning for image-based structural damage recognition," *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 9, pp. 748–768, 2018.

[54] S. Li, X. Zhao, and G. Zhou, "Automatic pixel-level multiple damage detection of concrete structure using fully convolutional network," *Computer-Aided Civil and Infrastructure Engineering*, 2019.

[55] S. Chakrabarti, B. Dom, R. Agrawal, and P. Raghavan, "Scalable feature selection, classification and signature generation for organizing large text databases into hierarchical topic taxonomies," *The VLDB journal*, vol. 7, no. 3, pp. 163–178, 1998.

[56] S. Kumar, J. Ghosh, and M. M. Crawford, "Hierarchical fusion of multiple classifiers for hyperspectral data analysis," *Pattern Analysis & Applications*, vol. 5, no. 2, pp. 210–220, 2002.

[57] F. Wu, J. Zhang, and V. Honavar, "Learning classifiers using hierarchically structured class taxonomies," in *International Symposium on Abstraction, Reformulation, and Approximation*, pp. 313–320, Springer, 2005.

[58] J. Rousu, C. Saunders, S. Szedmak, and J. Shawe-Taylor, "Kernel-based learning of hierarchical multilabel classification models," *Journal of Machine Learning Research*, vol. 7, no. Jul, pp. 1601–1626, 2006.

[59] C. Vens, J. Struyf, L. Schietgat, S. Džeroski, and H. Blockeel, "Decision trees for hierarchical multi-label classification," *Machine learning*, vol. 73, no. 2, p. 185, 2008.

[60] W. Bi and J. T. Kwok, "Hierarchical multilabel classification with minimum bayes risk," in *2012 IEEE 12th International Conference on Data Mining*, pp. 101–110, IEEE, 2012.

[61] R. Cerri, R. C. Barros, and A. C. De Carvalho, "Hierarchical multi-label classification using local neural networks," *Journal of Computer and System Sciences*, vol. 80, no. 1, pp. 39–56, 2014.

[62] J. Wehrmann, R. Cerri, and R. Barros, "Hierarchical multi-label classification networks," in *International Conference on Machine Learning*, pp. 5225–5234, 2018.

[63] S. Baker and A. Korhonen, "Initializing neural networks for hierarchical multi-label text classification," in *BioNLP 2017*, pp. 307–315, 2017.

[64] X. Guo, L. Chen, and C. Shen, "Hierarchical adaptive deep convolution neural network and its application to bearing fault diagnosis," *Measurement*, vol. 93, pp. 490–502, 2016.

[65] M. Gan, C. Wang, *et al.*, "Construction of hierarchical diagnosis network based on deep learning and its application in the fault pattern recognition of rolling element bearings," *Mechanical Systems and Signal Processing*, vol. 72, pp. 92–104, 2016.

[66] M. Marszalek and C. Schmid, "Semantic hierarchies for visual object recognition," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–7, IEEE, 2007.

[67] M. Marszałek and C. Schmid, "Constructing category hierarchies for visual recognition," in *European conference on computer vision*, pp. 479–491, Springer, 2008.

[68] T. Gao and D. Koller, "Discriminative learning of relaxed hierarchy for large-scale visual recognition," in *2011 International Conference on Computer Vision*, pp. 2072–2079, IEEE, 2011.

[69] B. Zhao, F. Li, and E. P. Xing, "Large-scale category structure aware image categorization," in *Advances in Neural Information Processing Systems*, pp. 1251–1259, 2011.

[70] R. Salakhutdinov, A. Torralba, and J. Tenenbaum, "Learning to share visual appearance for multiclass object detection," in *CVPR 2011*, pp. 1481–1488, IEEE, 2011.

[71] N. Razavi, J. Gall, and L. Van Gool, "Scalable multi-class object detection," in *CVPR 2011*, pp. 1505–1512, IEEE, 2011.

[72] N. Verma, D. Mahajan, S. Sellamanickam, and V. Nair, "Learning hierarchical similarity metrics," in *2012 IEEE conference on computer vision and pattern recognition*, pp. 2280–2287, IEEE, 2012.

[73] N. Srivastava and R. R. Salakhutdinov, "Discriminative transfer learning with tree-based priors," in *Advances in Neural Information Processing Systems*, pp. 2094–2102, 2013.

[74] J. Deng, N. Ding, Y. Jia, A. Frome, K. Murphy, S. Bengio, Y. Li, H. Neven, and H. Adam, "Large-scale object classification using label relation graphs," in *European conference on computer vision*, pp. 48–64, Springer, 2014.

[75] D. Warde-Farley, A. Rabinovich, and D. Anguelov, "Self-informed neural network structure learning," *arXiv preprint arXiv:1412.6563*, 2014.

[76] M. Ristin, J. Gall, M. Guillaumin, and L. Van Gool, "From categories to subcategories: large-scale image classification with partial class label refinement," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 231–239, 2015.

[77] Z. Yan, H. Zhang, R. Piramuthu, V. Jagadeesh, D. DeCoste, W. Di, and Y. Yu, "Hd-cnn: hierarchical deep convolutional neural networks for large scale visual recognition," in *Proceedings of the IEEE international conference on computer vision*, pp. 2740–2748, 2015.

[78] C. Murdock, Z. Li, H. Zhou, and T. Duerig, "Blockout: Dynamic model selection for hierarchical deep networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2583–2591, 2016.

[79] Y. Guo, Y. Liu, E. M. Bakker, Y. Guo, and M. S. Lew, "Cnn-rnn: a large-scale hierarchical image classification framework," *Multimedia Tools and Applications*, pp. 1–21, 2018.

[80] T. Xiao, J. Zhang, K. Yang, Y. Peng, and Z. Zhang, "Error-driven incremental learning in deep convolutional neural network for large-scale image classification," in *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 177–186, ACM, 2014.

[81] D. Roy, P. Panda, and K. Roy, "Tree-cnn: a hierarchical deep convolutional neural network for incremental learning," *arXiv preprint arXiv:1802.05800*, 2018.

[82] T. G. Dietterich, "Machine-learning research," *AI magazine*, vol. 18, no. 4, p. 97, 1997.

[83] X.-W. Chen and X. Lin, "Big data deep learning: challenges and perspectives," *IEEE access*, vol. 2, pp. 514–525, 2014.

[84] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning requires rethinking generalization," *arXiv preprint arXiv:1611.03530*, 2016.

[85] W. Yin, K. Kann, M. Yu, and H. Schütze, "Comparative study of cnn and rnn for natural language processing," *arXiv preprint arXiv:1702.01923*, 2017.

[86] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4293–4302, 2016.

[87] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, 2016.

[88] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: a brief review," *Computational intelligence and neuroscience*, vol. 2018, 2018.

[89] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive review," *Neural computation*, vol. 29, no. 9, pp. 2352–2449, 2017.

[90] E. Kang, J. Min, and J. C. Ye, "A deep convolutional neural network using directional wavelets for low-dose x-ray ct reconstruction," *Medical physics*, vol. 44, no. 10, pp. e360–e375, 2017.

[91] D. Kang and Y.-J. Cha, "Autonomous uavs for structural health monitoring using deep learning and an ultrasonic beacon system with geo-tagging," *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 10, pp. 885–902, 2018.

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

[93] T. Sercu, C. Puhrsch, B. Kingsbury, and Y. LeCun, "Very deep multilingual convolutional neural networks for lvcsr," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pp. 4955–4959, IEEE, 2016.

[94] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation," *CoRR*, vol. abs/1801.04381, 2018.

[95] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," *CoRR*, vol. abs/1707.01083, 2017.

[96] J. L. Elman, "Learning and development in neural networks: The importance of starting small," *Cognition*, vol. 48, no. 1, pp. 71–99, 1993.

[97] S. J. Pan, Q. Yang, *et al.*, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[98] L. Torrey and J. Shavlik, "Transfer learning," in *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*, pp. 242–264, IGI Global, 2010.

[99] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, "Transfer learning for time series classification," in *2018 IEEE International Conference on Big Data (Big Data)*, pp. 1367–1376, IEEE, 2018.

[100] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1717–1724, 2014.

[101] W. Pan, E. W. Xiang, N. N. Liu, and Q. Yang, "Transfer learning in collaborative filtering for sparsity reduction," in *Twenty-fourth AAAI conference on artificial intelligence*, 2010.

[102] A. Estabrooks, T. Jo, and N. Japkowicz, "A multiple resampling method for learning from imbalanced data sets," *Computational intelligence*, vol. 20, no. 1, pp. 18–36, 2004.

[103] R. Caruana, S. Lawrence, and C. L. Giles, "Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping," in *Advances in neural information processing systems*, pp. 402–408, 2001.