Application of Convolutional Neural Network (CNN) Models for Automated Monitoring of Road Pavement and Winter Surface Conditions Using Visual-Spectrum and Thermal Video Cameras

Ce ZHANG

Department of Civil Engineering and Applied Mechanics

McGill University

August 2020

A thesis submitted to McGill University in partial fulfillment of the requirements of a degree of Master of Engineering in the Department of Civil Engineering and Applied Mechanics.

Abstract

Road condition plays a critical role in traffic safety, management and operations. Defective pavement conditions such as cracks or potholes deteriorate road user's comfort, damage vehicle, generate crashes and increase emissions. In addition to pavement deterioration, northern cities suffer from adverse road surface conditions caused by a long period of snow and ice during winter months. To guarantee traffic operations and safety, transportation agencies spend significant amounts of their available resources to monitor and maintain pavement and winter road surface conditions. To reduce costs, the use of automated monitoring solutions is crucial for the systematic detection of pavement distress as well as the detection of hazardous winter road surface conditions, such as snowy or icy conditions. Pavement distress and winter surface evaluation can be done through manual surveys, i.e., visual inspections of pavement images obtained through video cameras attached to an inspection vehicle. To reduce the costs of manual inspections, research by industry have moved quickly towards the development and implementation of automated road surface assessment systems using automated image processing.

Taking into account the latest developments, the objectives of this research work are: 1) to propose and evaluate an original convolutional neural-network methodology for automated detection and classification of pavement distress types using a low-cost data collection strategy and alternative data generation models; 2) to extend the methodology for the automated detection of winter-road surface conditions combining both RGB and thermal video images.

For the first objective, the pavement distress deterioration classifications used includes linear or longitudinal cracking, network cracking, fatigue cracking or potholes, pavement marking, etc. The models are trained and tested based on an image dataset collected from Montreal's pavement conditions. A sensitivity analysis was done to evaluate different regularization scenarios and data generation strategies, especially from the input image scaling and partitioning. The detection rate and classification accuracy of the proposed approach with trained Convolutional Neural Network (CNN) models goes up to 83.8% for the test set, which is promising when compared to the latest top research in the literature. More Specifically, the F1- Score are 0.808 for "Pothole", 0.802 for "Patch", 0.860 for "Marking",

0.796 for “Crack-Linear” and 0.813 for “Crack-Network”. However, by merging linear and network crack classes together, the F1-Score over the merged class increases to 0.916.

For the second objective, RGB and thermal images are collected and manually classified into four general classes: snowy, icy, wet and slushy. From the original dataset, three data generation methods were evaluated using artificial, split and multiple class data generation strategies. Alternative Convolutional Neural Network model structures with single data stream and double data streams were tested. The results show that the “Snowy”, “Wet” and “Slushy” conditions have better detection in RGB images while “Icy” conditions are better observed in thermal images. The multiple stream input network has the best result based on the average precision on the original dataset. Moreover, Moreover, it was found that the multiple stream input network with low weight improved the performance and it is surmised that artificial images could also result in the same effect. However, one shortcoming of artificial images is the problem with overfitting. The recall, precision and F1-**S**core over the test dataset of double data stream model is 0.948,0.948 and 0.927, respectively and the F1-Score for each class is 0.866 for “Snowy”, 0.935 for “Icy”, 0.985 for “Wet” and 0.888 for “Slushy”.

Résumé (Fr)

L'état des routes joue un rôle critique dans la sécurité routière, la gestion et l'exploitation du réseau routier. Les chaussées en mauvaise condition ayant des fissures ou des cavités (des nids-de-poule) détériorent le confort des usagers de la route, endommagent les véhicules, causent des collisions et augmentent les émissions. En plus de la détérioration de la chaussée, les villes nordiques souffrent de conditions de chaussée défavorables causées par la neige et la glace durant les mois d'hiver. Pour garantir l'opération et la sécurité de leurs réseaux, les agences de transport dépensent des budgets importants pour surveiller et maintenir l'état de la chaussée et de la surface des routes pendant l'hiver. Pour réduire les coûts, l'utilisation de solutions de surveillance automatisées est essentielle pour la détection systématique des dégradations de la chaussée ainsi que pour la détection des conditions dangereuses de la surface des routes en hiver, telles que la neige ou le verglas. L'évaluation de la dégradation de la chaussée et de la surface hivernale peut être effectuée manuellement, c'est-à-dire des inspections visuelles des images de la chaussée obtenues par des caméras vidéo fixées à un véhicule d'inspection. Afin de réduire les coûts des inspections manuelles, la recherche et l'industrie ont évolué vers le développement et la mise en œuvre de systèmes automatisés d'évaluation de la surface des routes à l'aide du traitement automatisé des images.

En tant compte des développements les plus récents, les objectifs de cette recherche sont les suivants 1) proposer et évaluer une méthodologie originale de réseau neuronal convolutif pour la détection et la classification automatisées des types de dégradation de la chaussée en utilisant une stratégie de collecte de données peu coûteuse et des modèles alternatifs pour la génération de données ; 2) développer un extension de la méthodologie pour la détection automatisée des conditions de surface des routes en hiver en combinant à la fois des images vidéo RVB et thermiques.

Pour le premier objectif, la classification de la détérioration de la chaussée comprend la fissuration linéaire ou longitudinale, la fissuration en réseau, la fissuration par fatigue ou les nids-de-poule, le marquage de la chaussée, etc. Les modèles sont calibrés et testés sur la base d'un ensemble de données d'images recueillies sur les conditions de la chaussée de Montréal. Une analyse de sensibilité est effectuée pour évaluer différents scénarios de régularisation et stratégies de génération de données, en particulier la mise à l'échelle et le partitionnement des images d'entrée. Le taux de détection et la précision de la classification de l'approche

proposée avec les modèles de réseaux neuronaux convolutifs (RNC) formés atteignent 83,8% par rapport à l'ensemble des tests, ce qui est prometteur en comparaison avec les dernières recherches de pointe dans la littérature. Plus précisément, les scores F1 sont de 0,808 pour "nid-de-poule", 0,802 pour "remplissage", 0,860 pour "marquage", 0,796 pour "fissure-linéaire" et 0,813 pour "fissure en réseau ". Cependant, en fusionnant les classes de fissure linéaire et de fissure réseau, le score F1 sur la classe fusionnée augmente à 0,916.

Pour le second objectif, des images RVB et thermiques sont collectées et classifiées manuellement en quatre classes générales : enneigé, glacé, mouillé et neige fondue « en sloche ». À partir de l'ensemble de données original, trois méthodes de génération de données ont été évaluées en utilisant des stratégies de génération de données artificielles, fractionnées et à classes multiples. Des structures alternatives de modèles de réseaux neuronaux convolutifs avec un flux de données unique et un flux de données double ont été testées. Les résultats montrent que les conditions enneigées, mouillées et en sloche sont mieux observées sur les images RGB alors que les conditions glacées sont mieux observées sur les images thermiques. Le réseau d'entrée à flux multiples a produit le meilleur résultat en fonction de la précision moyenne de l'ensemble de données d'origine. En outre, il a été montré que le réseau d'entrée à flux multiples avec un poids faible pouvait améliorer les performances et que l'ajout d'images artificielles aurait le même effet. Cependant, le seul défaut des images artificielles est le problème du surapprentissage. Le rappel, la précision et la cote F1 sur l'ensemble de données de test du modèle à double flux de données sont respectivement de 0,948, 0,948 et 0,927, et la cote F1 pour chaque classe est de 0,866 pour la condition enneigée, 0,935 pour la condition glacée, 0,985 pour la condition mouillée et 0,888 pour la condition en sloche.

Acknowledgements

I would like to thank my supervisor, Dr. Luis Miranda-Moreno and Dr. Lijun Sun for their support and guidance throughout this project. I would like to thank Prof. Miranda-Moreno for the conception of this study and his offer to loan the device. I would like to thank Prof. Miranda- Moreno and Prof. Lijun Sun for their support of this project and my studies.

I extend my gratitude toward Ehsan Nateghinia for his assistance with installing the camera, collecting the dataset and reviewing the project. I would like to thank Ting Fu for taking time out of his research to teach me how to use the equipment. I would like to thank Bismarck Ledezema-Navarro and his friends' assistance with labeling the image data. Thank you to Mohamed Abdel Salam for the translation of my abstract.

I would like to thank all the members of the IMATS lab for their friendship, encouragement, and their words of advice. They made the working environment of the lab lighthearted and lively. I would like to thank my classmates and professors from Shandong University, without their professional and personal guidance, I would not be where I am today. Finally, I would like to thank my parents and friends for their endless support of my endeavors.

Contribution of Authors

This thesis is a combination of papers written for publishing or conference presentations. My contributions to this research include setting up the system, collecting the data, labeling the data, preprocessing the data, building the model and writing the manuscript. My supervisor, Prof. Luis Miranda-Moreno, provided guidance, comments, and editorial revisions throughout the entire process. The co-authors of the publications help in data collection, providing comments, and editing the papers.

Chapter 3

Pavement Distress Detection Using Convolutional Neural Network (CNN): A Case Study in Montreal, Canada. This paper was submitted for publication to the International Journal of Transportation Science and Technology.

Chapter 4

Winter Road Surface Conditions Classification using Convolutional Neural Network (CNN): Visible - Light and Thermal Images Fusion. This paper was accepted to the TRB Annual Meeting 2021 to take place in January 2021. It has been also submitted to the Canadian Journal of Civil Engineering.

I am the sole author of all additional chapters.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Background

Road pavement surface conditions play a critical role in motor-vehicle traffic operations and their safety. Defective surface conditions in the form of pavement cracks or potholes deteriorate vehicles, increase operating costs and emissions and more importantly can generate crashes and road injuries [1]. Pavement road conditions have been linked to operating costs, greenhouse gas emissions and noises [2].

In addition to the pavement deterioration, northern cities suffer from cold winters in which road surface (friction) conditions can be deteriorated by the presence of snow and/or ice during winter months. Severe weather conditions and fluctuations between seasons in northern countries cause greater damage due to the freeze-thaw cycles (look for citation). Long periods of snowy weather and low temperature are responsible for slippery roads [3]. The reduced friction of slippery frozen roads have been studied for their negative impact on traffic safety for many years [4]. From 2007 to 2016, more than 1,235,000 crashes were weather-related and the crashes caused more than 400,000 injuries and 5,000 fatalities in the U.S. [5]. Besides, photokeratitis (also called snow blindness) happens more often to drivers who are crossing the boundless snow for a long time. To guarantee traffic operations and safety, transportation agencies spend significant amounts of their available resources to monitor and maintain pavement and winter road surface conditions. To reduce the costs of manual inspections, research by industry have moved quickly towards the development and implementation of automated road surface assessment systems using automated image processing. The use of automated monitoring solutions is crucial for the systematic detection of pavement distress as well as the detection of hazardous winter road surface conditions, such as snowy or icy conditions. Pavement distress and winter surface evaluation can be done through manual surveys, i.e., visual inspections of pavement images obtained through video cameras attached to an inspection vehicle. However, this approach requires a lot of human resources and substantial amounts of financial support. For instance, to measure the distress on the pavement, the American Society for Testing and Materials (ASTM) standardized the

Pavement Condition Index (PCI), which was originally developed by the United States Army Corps of Engineers (USACE) [6]. The PCI is based on a visual survey where researchers identify the distress types and area first as it will mean the use of different coefficients during calculation. They then collect the total area before finally calculating the PCI using the distress coefficient, area, and total area. The result is a PCI value that ranges from 0 to 100, or worst to best condition, respectively. Although PCI is able to provide a clear indication of the road conditions, it relies on well-trained personnel to complete the survey procedure. As for the winter road conditions, the government has to make the snow removal decision from the stationary Road Weather Information System (RWIS). RWIS is an integrated system that includes environmental sensors for measuring temperature, humidity, air pressure, visibility, water level conditions and pavement conditions, the communication system for data transition, data processing hardware, etc. With RWIS, the transportation management department can collect real-time precise road weather data from the sensors and then decide on the snow removal work accordingly. However, it is impossible to cover all the road networks with such huge, expensive, and complicated RWIS. Recently, road distress detection has benefited from the development of different types of cameras. Many camera-based systems have been built to collect pavement distress [7-10]. Some other systems include Lidar and laser [11], which also attract researchers' interests. Despite the alternative technologies, image classification algorithms make great progress given the last developments in computer vision and machine learning. Instead of relying on the traditional edge detection or image thresholding technologies [8], Support Vector Machine (SVM) [12, 13], decision tree [14] and Convolutional Neural Network (CNN) [7, 15] have become more and more popular.

In terms of the winter road condition detection system, earlier research relied on wheel-based vehicles [16]. However, in the past few years, many camera-based systems were proposed by researchers [9, 17-20]. Based on the physical structure and surface temperature feature differences between water, ice and snow, the audio-visual [21] system and near-infrared technology [22] have taken much more attention. In classification algorithms, it is no doubt that SVM [19, 23], Random Forest (RF) [20], Artificial Neural Network (ANN), K-Nearest Neighbor (KNN)[19, 21] and CNN [20] are the most popular classifiers.

1.2 Literature Gaps

This research strives to fill several gaps which currently exist in the literature. Although a large portion of the literature focuses on highway distress detection, most of their data collection systems are complicated, expensive and heavy. Thus, it is necessary to install their system on special vehicles resulting in a long time to cover most of the cities' pavements. Moreover, most available pavement distress datasets were collected from highways, which are regularly maintained. As a result, their datasets have clearer images and less distress related ones. Hence, there is a particular gap surrounding urban pavement distress conditions.

In terms of winter road condition research, many studies have looked at the application of RGB cameras, but few of them have examined the application of thermal cameras and even fewer have combined both of them. At the time of submission, no other studies are known to use a double camera-based system to collect the winter road condition data. In order to classify the winter road conditions, most studies have used Support Vector Machine (SVM), Random Forest (RF), and Convolutional Neural Networks (CNN). Although these models can be a powerful tool for analysis, their performance cannot achieve researchers' satisfaction because single input data has limited information. To our knowledge, no studies have looked at the performance of double stream input model, especially two different image type input. Furthermore, a comparison between the impacts of different image preprocessing methods has been left untouched.

1.3 Objectives

The general objective of this thesis is to develop a computer-vision based methodology to collect and automatically classify road condition images as a complete dataset and introduce CNN models as a classifier to distinguish multi-class road conditions.

The specific objectives are as follows:

- To propose and evaluate an original CNN methodology for automated detection and classification of pavement distress types using low-cost data collection systems and alternative training-data generation models. In this work, pavement distress deterioration classification includes linear or longitudinal cracking, network

cracking, fatigue cracking or potholes, pavement marking, etc. As part of this objective, a sensitivity analysis is done to evaluate alternative regularization scenarios and data generation strategies.

- To expand the proposed CNN methodology to winter-road surface applications by combining RGB and thermal video images. For this purpose, RGB and thermal images are collected and manually classified into four general classes: snowy, icy, wet and slushy. Alternative data generation methods are evaluated including artificial, split and multiple-class strategies. Alternative CNN model structures were tested with single and double data streams.

This research uses the City of Montreal, which is one of the biggest cities in Canada and suffers from pavement deterioration and adverse winter road conditions, as a case study. Firstly, its high latitude location causes a long period of extensive solar radiation in summer, hence, increasing its likelihood to fissure. Furthermore, Montreal has more than 120 rainy days and in the winter, Montreal uses more than 130,000 tons of salt for de-icing annually [16]. Both of the rain and the salt corrode the roads severely. Lastly, high traffic volumes and heavy trucks contribute to the damaged asphalt. In terms of winter road conditions, Montreal suffers from a number of snowy days, excessively low temperatures and unexpected freezing rain that lead to frequent road freezing from November to April. Every year, the City of Montreal allocates an important budget in road maintenance and snow removal. In 2019, Montreal announced 378 million dollars in road repair [17], and in the winter of 2018, Montreal spent 8 million dollars in snow-removal costs [18].

1.4 Contributions

This research contributes to the literature on the automated camera-based road pavement monitoring systems and the development of suitable CNN models for automated image analysis using visual spectrum and/or thermal video. Based on the gaps identified in the literature, the specific contributions of this work are as follows:

- To provide a procedure for collecting and generating pavement distress datasets with a large number of high-resolution images for training deep neural networks.

- To demonstrate the efficiency and applicability of CNN models for multiple distress-type detection and classification.
- To evaluate the impacts of parameters such as image sizes or regularization structures on classifying distress type.
- To propose a low-cost image data collection system with both thermal and RGB cameras.
- To provide way to collect and generate winter road condition datasets for thermal images for CNN model training.
- To evaluate the performances of alternative CNN networks and evaluate the performance of each model.

1.5 Organization

This research is organized in the following way: Chapter 2 provides a literature review of the existing work regarding topics on road condition classification with a focus on data collection systems and classification algorithms. Chapter 3 comprises the Montreal pavement distress dataset collected and generated by a camera-based system and Convolutional Neural Network models to classify the multiple distress type. Chapter 4 discusses the winter road condition classification that includes a complete dataset collected by RGB and thermal cameras, and alternative CNN models which are trained on the dataset. The methodology presented in each chapter consists of: data collection system setting, image preparation, labeling, and generation as well as CNN model calibration and testing. Chapter 5 concludes this research by summarizing all relevant findings. Limitations of this research and avenues for future work are also included.

2 Literature Review

The literature review related to this work concentrates on image-based data collection systems and detection-classification methods for road surface conditions. Accordingly, this literature review provides an overview of the past work on image-based detection system for pavement distress and winter conditions followed by automatic classification methods.

2.1 Data Collection Systems

2.1.1 2D & 3D Road Distress Detection Systems

One of the first two-dimensional (2D) image-based detection systems was proposed in 1991 by Mahler et al. [8] who designed a video-based imaging system named Automatic Crack Monitor (ACM). This system which was composed of a time code generator to record the milepost data and a video camera to take gray level photos acquired pavement data and detected cracks. To offset noise from the vehicle motion, they also installed a strobed, high-intensity xenon flash lamp. In their algorithms, they adopted crack detection, thinning, and crack-tracing algorithms where the gradient histogram was used to select the threshold of images and then to convert it into binary images. The crack detection algorithms were used for segmentation and labeling for the binary image. The thinning algorithm and crack-tracing algorithm were used to extract crack parameters, including branches, length and pixels. To our knowledge, this research was the first work that showed road conditions could be detected automatically. The main demonstration in this paper was the close attention to quantitative crack parameters. As a result, they found their system could detect more features of cracks like direction, length and width.

Later on in 2016, Yashon et al. [10] published a study that presented a traditional empirical approach for identification of incipient linear distress in asphalt pavements, including longitudinal, transverse, diagonal, block (random), and alligator (fatigue). In their study, they used multichannel RGB cameras instead of grayscale cameras. This enabled the separation and characterization of the different particle types on a heterogeneous surface. They used two different camera types in two different places: Canon EOS Mark II camera in Stuttgart and Nikon D7000 in Kenya. Afterward, they presented a triple-transform approach

for distress detection, isolation and classification, This approach comprised 2D Discrete Wavelet Transform (DWT), Successive Morphologic Transformation filtering (SMF), and Circular Radon Transform (CRT). The detection accuracy reached 83.2% for their test dataset, showing that the triple algorithms were reliable for automated linear distress detection, extraction, and classification.

More recently, in 2017, Markus et al. [7] presented the German Asphalt Pavement Distress (GAPs) dataset, which was a free 2D image pavement distress dataset being large enough to train deep neural networks. Following the specific approach for data collection called Road Monitoring and Assessment (RMA), they used a certified measuring vehicle. The main components of the vehicle were an inertial navigation system, laser sensors, a 2D laser range finder and two cameras with a frame rate of 32 fps and a resolution of 1920×1080 pixels. In total, their dataset had 1,969 gray images for six distress types, according to the German Road and Transportation Research Association (FGSV): crack, pothole, inlaid patch, applied patch, open joint, and bleeding. With the generated data, they trained several CNNs and compared them with classical networks with different techniques. As a result, they found that deep learning approaches were able to achieve satisfactory detection results. Also, they found that drop out was more efficient to get generalization results. Despite that the GAPs dataset is complete and standard, due to the lack of the precise damage location labels in terms of pixel coordinates, this labeling is not appropriate to train a classifier.

With more and more advanced equipment developed, many studies have also been conducted using 3D systems. Maria et al. [9] used a road condition monitoring system to collect road condition information in 2009. It included a monochrome-stereo camera system that could take a pair of photos with 640×480 pixels and then convert them into a 3D image. Another efficient 3D system was built based on LIDAR technology. Recently, many studies applied LIDAR technology in the laser crack measurement system (LCMS). In their system, the LIDAR part includes a digital area scan camera and a structured light projected a laser line. The camera took images of the structure light. Then the deformations of the laser line on the object area were analyzed to evaluate the depth for each point with a known horizontal position on the object [11]. Yi-Chang et al. [11] built a laser crack measurement system (LCMS) on a vehicle by installing two high-performance laser profiling units and each of them consisted of a laser line projector, a custom filter and a camera. The LCMS produced

4,160 points per profile and covered a 4m pavement width. With 100 km/h speed for the vehicle, the system collected the transverse profiles at 4.6mm intervals and the accuracy in elevation could achieve 0.5mm and in transverse 1mm. John et al. [19] built an LCMS using 3D LIDAR technology. Their system included 3D laser profilers, custom filters and a camera. Their system could automatedly detect crack types, crack depth and crack severity. As a result, the accuracy of the system reached 95% in general cracks classification. Although the LIDAR technology had quite a promising accuracy, the complexity and expense stopped the LIDAR system from becoming more popular in daily road maintenance.

2.1.2 Image-Based Winter Road Condition System

Later after the crack detection system, the image-based winter road condition system became popular. In 2010, Raqib et al. [20] proposed a road surface monitoring system based on GPS tagged images from low-cost cameras mounted on non-dedicated vehicles such as public transport or police vehicles. Their system focused on low-cost road surface data collection and processing applications, which is practical for large-scale implementation. Three images resources supplied that system: drive recorders, web cameras and analog videos, and divided into bare, snow-covered and wheel tracks. In their experiment, they found that the color difference was a significant matter between snow-covered and bare. Their proposed system reported an accuracy of 86% while applying image edge detection operation and SVM classifier. However, they indicated the accuracy would be affected by image resolution, camera angles and illumination condition.

More recently, Michael et al. [21] built a smartphone-based winter road surface condition (RSC) monitoring system installed on any smartphone device. In their system, the smartphone was mounted on the inside of the vehicle to capture the roadway ahead. The system recorded Global Position System (GPS) tagged, time-stamped images and then the images sent to an online server for processing and classification. By comparing with the manual winter patrol records from Ontario Ministry of Transportation (MTO), the results showed their proposed system had an average classification accuracy of 73% with over 16,000 collected images of three classes including bare, partly snow-covered, and fully snow-covered. This result meant that their smartphone-based system is a crowdsourcing solution for obtaining RSC information and analyzing snow cover density from the traveling public.

The previous works were mostly focused on weather forecast data, sensor information

and stationary cameras' image features, which only provide point-based information.

In 2012, Zhang et al. [22] proposed a video monitoring scheme capable of reflecting more spacious conditions. In their work, the snow area was detected by using a background edge model firstly. Then, a set of image features were extracted to tune and classified by three classifiers, including the Neural Network, Support Vector Machine, and K-Nearest Neighbor (KNN). As a result, their KNN outperformed the other two classifiers and reached an accuracy of 93% for three types of defined classes, including heavy snow cover, mild snow cover, and dry. However, their system did not perform well on distinguishing wet and icy road conditions.

Maria et al. [9] built a stereo camera-based on reflected light polarization changes from the road surface. Their system included a monochrome stereo camera placed inside, remote road surface state sensor, remote surface temperature sensor, DSC111 spectrometric camera system, and Volvo's Road Eye. By applying feature extraction algorithms based on light polarization changes and graininess analysis, their accuracy reached 90% when distinguishing icy, wet, snowy, and dry road conditions.

In 2018, Guangyuan et al. [23] applied image-based road condition monitoring system with Convolutional Neural Networks (CNN). They collected images from the patrol vehicles with a mobile collection unit and manually classified them to the two-class, three-class and five-class. The two-class includes bare and snow-covered; the three-class includes bare, partly snow-covered, and fully snow-covered; and the five class includes bare, less half snow-covered, half snow-covered, more than half snow-covered, and fully snow-covered. Their pre-trained CNN model reached 78.5% accuracy.

Before camera-based systems developed, a traditional way existed to discriminate road conditions using weather data, which includes temperature, humidity, wind speed, and wind direction from stationary Road Weather Information System (RWIS). However, it is unreasonable to only rely on this information. For example, it is improper to use the road temperature as a single variable that differentiates between wet and ice because icing conditions may occur at road temperatures between -20 °C to 0 °C instead of precisely at 0 °C. This is because the de-icing chemicals will cause freezing points of the surface fluid decrease [24].

In 1998, Andreas et al. [25] built a system that combined video data with weather

information, including temperature and wind speed. They collected images from a handheld Hi-8 camcorder standing at or near the roadside and filming various road states. Monochrome images were generated from the color components. Then monochrome images were categorized into five classes, including dry, wet, snowy, icy, and track. The track class only had 8 images, while others had 15-20 images. After extracting the main features including mean gray level, the standard deviation of the gray level, mean of the absolute value of the gray response to [-1, 1], mean of the absolute value of the gray response to [-1, 1]T, ratio of the standard deviation of the red image to the standard deviation of the blue image, and ratio of the standard deviation of the red image to the standard, they used neural network from MATLAB Toolbox as a main method to classify. Even they only had limited images, their result indicated that it was possible to distinguish with a 40 to 50 % rate of classification between all road states except for ice and wet, and ice and track.

In 2002, Kevin McFall et al. [26] introduced an audio-visual system without combining the image information with weather information. Firstly, the image data were collected by an analog high-resolution grayscale Sony SPT-M124 video camera and a Matrox Meteor II frame grabber card. Then, the acoustic data was captured using a rugged Larson/Davis microphone, designed specifically for outdoor use, connected to a Digital Audio Labs Deluxe sound card. Moreover, in order to ensure the synchronization of image and audio data collection, a necessary ring buffer was used. After that, they extracted grayscale pixel-level values, edge detection features, the size, and distribution of the spots containing the 10% brightest pixels from the image; from the audio data, the extracted feature was the signal spectrogram. After the feature extraction, they used the KNN as a classifier and found Hybrid (image and acoustic) result, except for dry condition, was reliable for icy, snowy, and wet road conditions. They concluded that the results for the dry conditions might need to be improved with more representative training data and further integration with other RWIS sensors.

In 2011, Patrik Jonsson [27, 28] evaluated the sufficiency of an extensive dataset retrieved from an RWIS site for a more accurate road condition classification than the results of a single image analysis tool. In their first experiment, the data was collected by a standard Swedish RWIS field station equipped with a standard near infra-red sensitive camera with infra-red searchlight. The RWIS could provide weather data including air temperature,

humidity and dew point, precipitation particle count, relative precipitation size, surface temperature, wind speed and direction, image data and day time or night time signal detection. Especially in their image data, the RWIS station extracted the image features like grayscale, edge, and emphasis. The Principal Component Analysis (PCA) was applied to dozens of features to find the relations between the input variables and expel the abundant variables. By applying the PCA, Patrik found 7 principal components were the best for the model performance, if more than 7 components applied, it could not fit the ground truth better and decreased the quality. In his results, the neural network accuracy reached 91% for dry, 100% for Icy, 100% for snowy, 74% for tracks, 100% for wet. Compared with simple image-based systems, the results show that the RWIS data can be a valuable input for road conditions classification to improve the performance of camera-based detection systems.

2.2 Classification Algorithms

2.2.1 Traditional Image Processing

The improvement of pavement monitoring systems has been not only in the optical sensors (data collection technologies) but also in algorithms for road condition detection. Automatic road condition detection has already attracted much interest in the literature and many image-processing algorithms for this purpose emerged in the past few years. These are divided mainly into two groups: traditional image processing and machine learning approaches, which are discussed as follows.

The first group, traditional image processing, mainly uses road-image thresholds or edge detection. For crack image thresholding, the most successful approach is the one that uses CrackIT Toolbox on MATLAB by Henrique et al. [29]. The toolbox includes four main modules: image preprocessing, crack detection, crack characterization into types, and evaluation routines. In the toolbox, images are mainly classified based on brightness and gray level differences.

Another popular method makes use of image edge detection. One example is the work of Ayenu et al. [30] that designed a Sobel detector to detect cracks after image smoothing and spackle-noise removal using a bi-dimensional empirical mode decomposition approach. Later on, Qin et al. [31] proposed CrackTree algorithm to address incapability problems of edge

detection. At first, they designed a new algorithm to remove the pavement shadow. Then they constructed a crack probability map using tensor voting [32], which was efficient to offset noise and fragment. After that, they constructed a graph model by sampling crack seeds from the former map, then constructed the minimum spanning tree (MST) of the graph and conducted recursive edge pruning in the MST to identify the final crack curves.

Moreover, image edge detection has a wide range of applications in extracting feature map for winter road condition detection. Raqib et al. [20] involved an image edge detection technique when extracting feature maps. They constructed RGB histogram from RGB images while they applied Gaussian Smoothing and gradient mask on gray images. After that, they inputted the processed images to the SVM classifier. In the paper by Zhang et al. [22], their feature maps were extracted from the co-occurrence matrix that represented the distribution of co-occurring adjacent vertically, horizontally, or diagonally pixel values by comparing their angular second moment, entropy, contrast, etc. Moreover, Maira et al. [9] applied two methods of measurements: light polarization changes and graininess analysis. The light polarization changes method is similar to that of Zhang et al. [22], which was based on light-reflecting difference between different surfaces like ice, water and dry road. The features were extracted by co-occurrence matrix from comparing the intensity differences of two images. Graininess measure was worked as a gradient mask to perform low-pass filtering to the image, which makes it blurrier.

2.2.2 Machine Learning

As an alternative approach, nowadays, machine learning became very popular in transportation applications in general and in road surface monitoring in particular. The decision tree [14], a very popular tool in machine learning, includes both paths and nodes. There are three types of nodes: decision nodes, chance nodes and end nodes. Every node works like a ‘test’ on if the feature from the image suit for the specific condition from the top node (roof) and then pass to the next node until the end node (leaf). Random Forest [33] is constructed by a multitude of decision trees. This method is not likely to suffer from overfitting problem like in decision trees. Guangyuan et al. [23] applied the Random Forest classifier in their research and achieved an accuracy of 74.0 %.

As an alternative technique, SVM is one of the supervised learning models that is often used in classification. With lots of points in space that cannot be classified using linear

regression, SVM maps these points into high dimensions in order to divide points by a clear hyperplane. In 2010, Jin et al. [12] proposed a recognition algorithm based on SVM with Gaussian kernel, and their results were deemed competitive in pothole detection. In addition to the application of distress classification, most research in winter road condition classification [20, 22] has first looked at the SVM.

In addition to SVM, k-Nearest Neighbors (k-NN) [34] is a non-parametric method used for classification and regression. In k-NN classification, the output is a class membership. The object labels are decided by a plurality vote of its neighbors, with that object assigned to the class most frequency among its k nearest neighbors. If k = 1, then the object is simply assigned to the class of the nearest neighbor. In the research of winter road condition detection [22, 26], K-NN is one of the most popular algorithms similar to SVM.

As another type, Artificial Neural Network (ANN) [35] is one of the symbols of machine learning, and it is inspired by the biological neural networks that constitute the animal network. An ANN is comprised of connected neuron units or nodes called artificial neurons and by the connection, each neuron unit can transmit a signal to other units. After the signal enters the neuron unit, the output is computed by some non-linear functions. Typically, neuron units are aggregated into layers. Different layers may perform different transformations on their inputs. Signals travel from the first layer (the input layer), to the last layer (the output layer), possibly after traversing the layers multiple times. If the nonlinear function between the neuron unit input and output is a convolutional function, the ANN then becomes the Convolutional Neural Networks (CNNs).

There is no doubt that CNN is the most popular method for image classification now – despite its work as a black box. CNN is composed of three types of layers [36]: convolutional, pooling, and fully connected layers. The convolutional layer applies the convolutional kernel to scan the image and extract the feature or texture from the image and then sends it to the next layer. The convolutional kernel is used repeatedly to every kernel sized block of pixels (also called receptive fields) instead of applying parameters for every pixel. In other words, the convolutional layers share the same parameters from convolutional kernels. The pooling layer generally goes after several convolutional layers; it reduces the dimensions of the data by combining the outputs of neuron clusters at one layer into a single neuron in the next layer. The full connected layer comes up later than convolutional and pooling layers, but its idea of

connecting every neuron in one layer to every neuron in another layer is common in traditional multi-layer perception neural network (MLP). CNN only leverages the full connected layer in the last part and most of the other layers are sparse connected, which reduces the redundant parameters reasonably. In 1998, Yann et al. [37] designed LeNet, including convolutional, polling, and fully connected layers. In 2012, Geoffrey et al. [36] proposed AlexNet, which added a non-linear activation function, Rectified Linear Unit (ReLU) and introduced Dropout to anti-overfitting. In 2014, Andrew et al. [38] from Oxford produced a VGG net. It has 16 or 19 layers and was demonstrated that the representation depth is beneficial for the classification accuracy. After that, Christian et al. [39] from Google designed GoogleNet. For GoogleNet, they thought more about the time and computation consumption so that they constructed Inception model, which can process different operations simultaneously. Furthermore, in 2015, Kaiming et al. [40] from Microsoft proposed ResNet. It has 152 layers, but because it uses residual from the former layer, it still keeps a low level of complication. The ResNet is the best CNN model in 2015 with 3.6 percent error rate in ImageNet Large Scale Visual Recognition Challenge (ILVRC), which is even better than human beings.

There are several studies that focus on road condition detection using CNN, finding that CNN methods perform better than other machine learning methods. For instance, Lei et al. [41] have compared CNN and SVM in road distress classification. Their result showed the F1 Score of CNN on their binary dataset is 0.8965, which is much higher than SVM. In road condition detection studies, Raqib et al. [20] indicated the performance of CNN on their dataset is much better than ANN, Random Forest, and CNN. In their result of a multi-class dataset, the accuracy of CNNS is 76.7%, which is much better than 75.6% of ANN, 77.9% of Random Tree, and 74.0% of Random Forest.

3 Pavement Distress Detection Using Convolutional Neural Network (CNN): A Case Study in Montreal, Canada

3.1 Introduction

Because of exposure to abrasion from vehicle traffic, material aging, and extreme weather conditions, pavement surface conditions deteriorate. This is manifested in the form of cracks or other distress types (distortion, disintegration, etc.) [42]. Cracks in highways make people uncomfortable while driving, damage vehicles, and even cause serious incidents or crashes. As part of the pavement infrastructure maintenance and rehabilitation, transportation agencies spend significant amounts of their available resources to monitor pavement conditions. To reduce costs, the use of automated monitoring solutions is crucial. Despite that pavement distress detection is through manual surveys (using visual inspections of pavement images from inspection vehicles), automated video-based road surface monitoring systems have emerged in research and practice in the last years. Collecting pavement surface data is a tedious process and the governments need to invest in human resources, special trucks, multiple sensors or cameras, and the mechanical mounting arms and tools to provide robustness against shaking and acceleration for the data collecting unit [9]. For this purpose, cities use multifunctional vehicles equipped with sensors. In their surveys, cities compute a score called pavement condition index (PCI) for each block of the road, based on distress type and severity by manual measurement [43]. The PCI ranges from 1 to 100, where 100 represents a section without any imperfection. Manual or automated classification of distress types is required for this purpose.

In the literature, there are many studies focused on crack detection in highways' pavement, for instance, Mahler et al. [8] applied image processing techniques to extract crack features from highway images and analyzed those features to see if a repair is needed. However, there are few studies about cracks on urban road pavements in North American cold cities with often higher exposure to traffic and, very low temperature and frost. According to the Montreal PCI Report in 2018, the average PCI for 14114 urban pavements was 61.6, and more than 30% of pavements were in bad or worse conditions. Compared to highway pavement cracks, urban pavement cracks can have more critical impacts on levels of comfort for road users in particularly those often classified as vulnerable, including pedestrians and bicycles in general, the elderly, and persons with disabilities. For these

groups, surface conditions are more critical. Pavement cracks near intersections or crosswalks can increase the risk of road accidents or falls [44]. Studies in the literature have used machine learning and deep learning methods. However, most of the studies have focused on only one or two distress types. For example, Ouma's research focuses on linear cracks [10] and Thitirat's research focuses on potholes [45]. Besides, the data collection module used in most studies is complicated and heavy, which has to be installed on special vehicles. Their professional cameras cannot provide stable images without their fully equipped vehicle thus making it near impossible to cover the vast amount of urban pavements. The limited budget only allowed their vehicle cruise restricted national highways.

Taking the latest developments into account, this research proposes and evaluates a methodology for automated detection and classification of pavement distress types using convolutional neural-networks and a low-cost video data collection strategy. The camera used in this research can be installed in almost every vehicle, including cars or bicycles. Pavement distress deterioration classification includes linear or longitudinal cracking, network cracking, fatigue cracking or potholes, and pavement markings. A pavement dataset with an adequate number of high-quality images is collected and manually labeled based on video images collected in the City of Montreal. As part of the work, alternative data generation models are evaluated to identify the best alternatives. Alternative CNN models are implemented with various structures, regularization, and input size tuning.

3.2 Methodology

The methodology consists of four steps: 1) data collection, 2) dataset preparation, 3) image labeling and 4) building deep neural networks to learn the patterns of different pavement distresses in pavement images. These distress types we investigate in this paper include linear cracking, network cracking, fatigue cracking or pothole, patches, marking, and clean pavements. This section describes the details of each of these processes.

3.2.1 Dataset Preparation and Image Annotation

The city of Montreal road network is used as an application environment in this study. Collecting images from Montreal streets provides a great opportunity to study unusual pavement distresses. In this city, as a result of winter weather conditions, urban pavement surfaces are likely to be distressed. The city's high latitude and good air quality expose pavements to high-level of sun radiation; besides, the large temperature difference between its

daytime and nighttime in summer and winter accelerates the pavement aging. Moreover, Montreal has a humid continental climate with several rainy days in spring and fall, and snowy days in winter. The water weakens the pavement surface and the winter maintenance operations cause heavy erosion.

For data collection, a camera-based approach is adopted with a camera being mounted in the front bumper of a vehicle. After selecting some streets, the vehicle circulates in the city and takes consecutive images from the pavement.

To collect high-quality images without shaking and vibration, a popular sports camera, GoPro Hero 7, is used. The super-smooth function of the camera, which reduces the effects of anti-sport shakes, was used when taking road surface images. The camera resolution was set as 720 dpi, focal-ratio (aperture) as f/2.8, film speed as ISO 293, and the exposure time as 1/256 seconds. The dimensions of the output images have been set to 3000×4000 pixels, and the camera automatically records one image per second. After 6 hours of data collection from streets with cracking issues, 12,000 images containing a variety of pavement surface types were collected in June 2019. From these 12,000 images, the 2,000 best images, which include most of the classical distress types, were selected. During the algorithm's development step, it was found out that the number of pothole images was excessively small. Therefore, the data collection was repeated for the second day and 105 pothole images were added to the dataset.

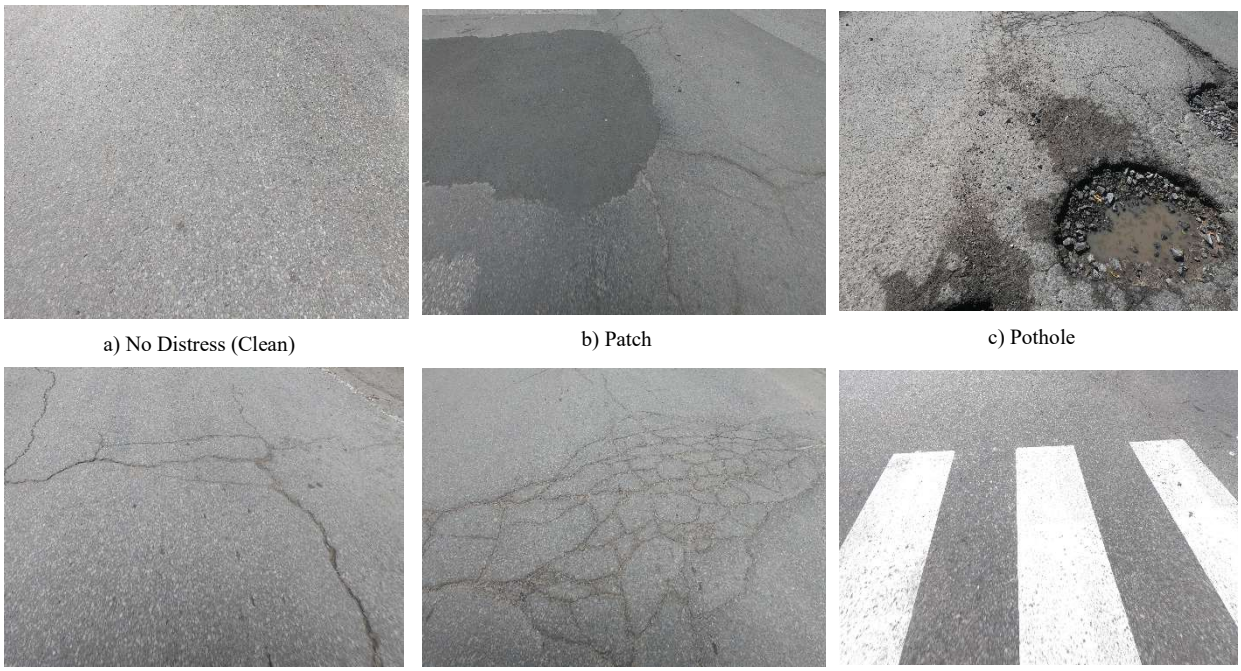
To stable and standardize the dataset, images were converted into grayscale firstly because when detecting the distress, the texture features are much more important than color features, which most distress types are monochromatic and gray images will lead to fewer parameters to improve the efficiency. Then, image histogram equalization was applied to them to homogenize the intensity distribution of different images. Finally, a median filter was applied to filter out Salt & Pepper noise.

Based on the standard from American Society for Testing and Materials (ASTM) International, there are 18 distress types generally [46]: blow up/buckling, corner break, divided slab, durability crack, faulting, joint seal, lane/shoulder, linear cracking, patching (large), patching (small), polished aggregate, pop-outs, pumping, railroad crossing, scaling, shrinkage, spalling corner, and spalling joint. To build the dataset, four types of general pavement distress, in addition to the pavement markings, were manually labeled by using small polygons with different colors. All the distress types observed in the collected dataset

were labeled as either *patch*, *pothole*, *linear crack* (longitudinal, transverse), or *network crack* (alligator) like durability crack, which is the most common distress type in Montreal. Therefore, the collected dataset has six classes, including the four distress types along with marking and clean images classes.

In this research work, "*Clean*" type means there is no distress in the pavement's image (Figure 1 (a)). The "*Patching*" class, illustrated in Figure 1 (b), means there is an area of pavement that has been replaced by new materials to repair old distresses on the pavement. This looks darker and fresher than other parts. The "*Pothole*" class, Figure 1 (c), corresponds to the pavement's images that have small - bowl-shaped depressions on the pavement surface that generally has sharp edges and vertical sides near the top of the hole.

In this dataset, the "*Linear crack*" class, Figure 1 (d), has been defined as a set of longitudinal, transverse, joint reflection, or/and a block of cracks. Although these crack types may have a difference in directions and connections, they are all mainly comprised of long-crack lines and should be treated in the same way. Moreover, the "*Crack network,*" called alligator, illustrated in Figure 1 (e), is very different from the linear crack. It is comprised of a series of interconnected cracks by fatigue and has many-sided sharp-angled pieces that make the pavement looks like an alligator back. Lastly, the "*Pavement marking*" class, Figure 1 (f), includes images of road surface marking (crosswalks, traffic lane marking, etc.), which are commonly in white or yellow colors with significant edges.

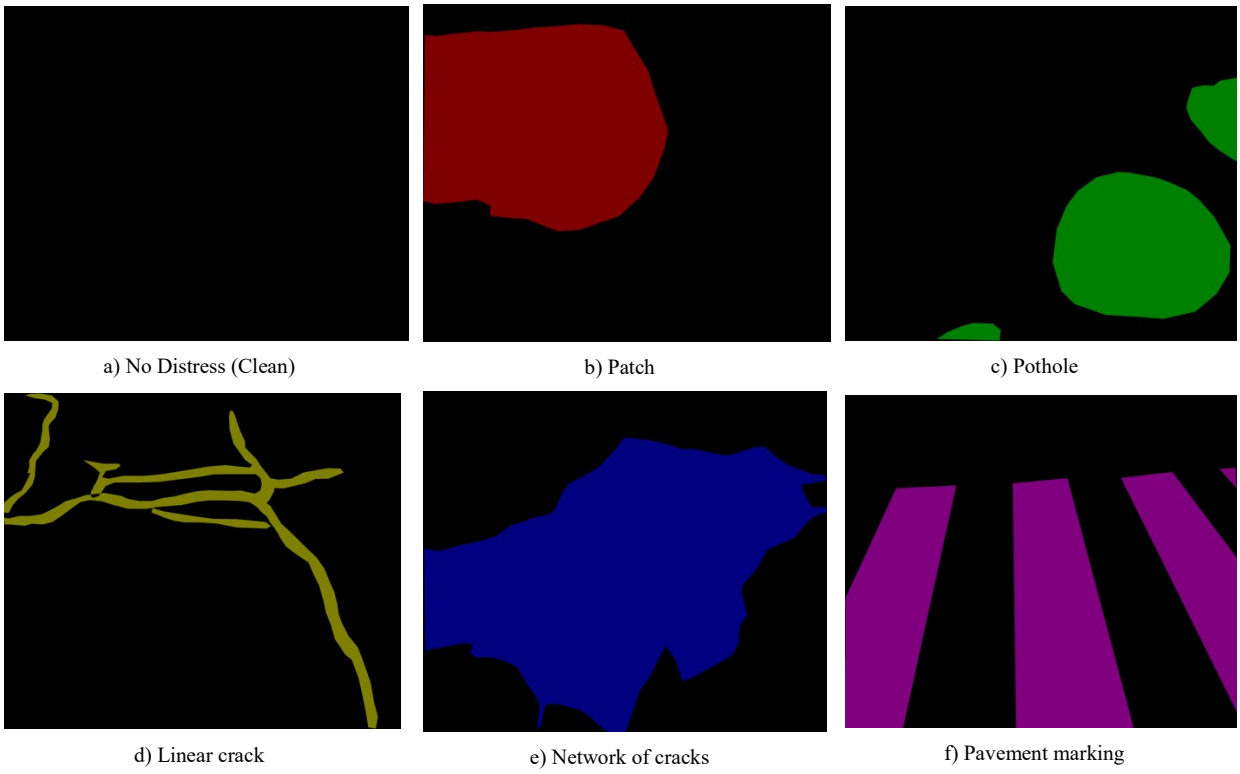


a) No Distress (Clean) b) Patch c) Pothole

d) Linear Crack e) Network of Cracks f) Pavement Marking

Figure 1: Different pavement distress types

The images in Figure 2 show the annotated images for the examples provided in Figure 1. For annotating different distress types, different colors (intensities) are used, and the value of the intensity is considered as the numerical value of the corresponding labels. The clean image (Figure 2 (a), does not have any annotation since there is no distress in the image. For the patching and pothole examples, polygons with different colors have been used, Figure 2 (b & c). The image with linear crack is also annotated with polygons, Figure 2 (d), but instead of the whole area, only cracks are filled with the narrow polygons. On the other hand, the network of cracks is annotated the same as patch and pothole, Figure 2 (e), since the whole area of distress needs to be repaired. Lastly, the marking annotation, Figure 2 (f), is the same as the marking shape.



a) No Distress (Clean) b) Patch c) Pothole

d) Linear crack e) Network of cracks f) Pavement marking

Figure 2: Annotated images of the sample images in Figure 1.

3.2.2 Images Partitioning and Sampling

Each of the original images is in the size of 3000×4000 pixels and takes about 4 MB of the storage. For applying the deep learning model, each image needs to be split into a smaller image. Generally, the small partitioned images, which are used for training the deep neural network, are in the sizes between 100×100 to 200×200. The resolution of the original image is high and splitting it to an image with these sizes produces additional images without vital information. To avoid redundant information, the original image needs to be resized to a smaller image: for example, resizing to half and then can be split to small sub-images (for example, 100×100). However, choosing the resizing and splitting factors is considered as hyper-parameters - meaning that a variety of settings needs to be implemented and tested for tuning these two factors.

In this research, four different cases are implemented and tested. The details of these four sub-datasets are shown in Table 1. The two resizing factors are 0.5 and 0.25, which convert the original image from 3000×4000 pixels to 1500×2000 pixels and 750×1000 pixels, respectively. This results in splitting sizes of 100, 150, and 200. However, to avoid having half-size sub-images, the resizing factors are a bit different from these two values. For example, for the second dataset, each original image is resized to 1500×2100 pixels instead of

1500×2000 and then split to 10×14 sub-images with the size of 150×150 pixels. This process splits an original image to 140 sub-images and, in general, splits all 2,105 images into 294,700 partitioned images. Finally, 294,700 sub-images were re-arrayed and grouped into 21 structures (*NumPy* files), each one having approximately 14,000 sub-images. Similarly, the 2105 labeled mask images were also resized, partitioned, and re-arrayed into the 21 *NumPy* structure, each structure having 14,000 sub-masks.

Table 1: Images Partitioning Scenarios

Dataset Number	Original Image	Number of Original Images	Reduced Image Size	Sub-Image Size	Number of Sub-Images	Number of Image of Clean Class	Number of Images of Other Classes	Number of *Numpy* Structure	Number of Images in *Numpy* Structures
1	3000×4000	2105	1500×2000	100×100	631,500	456,028	175,472	21	30,000
2	3000×4000	2105	1500×2100	150×150 (A)	294,700	206,199	88,501	21	14,000
3	3000×4000	2105	750×1050	150×150 (B)	73,675	49,877	23,798	21	3,500
4	3000×4000	2105	1600×2000	200×200	168,400	115,321	53,079	21	8,000

In addition to the tuning of the image partitioning hyper-parameters, the deep neural network also has its hyper-parameters. To fulfill these two tasks, the second dataset summarized in Table 1 is chosen and used to tuned convolutional neural network models. Then, all the above datasets will be tested with the tuned CNN and the best one will be selected, in terms of minimizing the error measures,

For any of the above datasets, the number of image samples of different classes is not evenly distributed. Table 2 shows the detailed description and splitting criteria of the second sub-dataset with a size of 150×150 (A) sub-image. Of 294,700 total images in the second dataset, only 91,280 images, 31% of total images, are used for the learning process. As we can see, the "*Clean*" class is the largest class by having 70% of the total images. To avoid unbiased dataset and test all the distress types, images are un-evenly sampled from each class and un-evenly split to training and test samples.

For example, of 206,199 *Clean* images, only 26,806 images (13% of the total class size) are randomly sampled for the training set and 2,062 images (1% of the total class size) are sampled for the test set. This split forms 32.2% of the sampled training set and 25.6% of the sampled test set. On the other hand, all the images of the "*Pothole"* class, which is the smallest class. These images are included in the learning process, where 85% of them are

randomly selected for the training set, and the remaining 15% are selected for the test set. For the "*Crack-Linear"* and "*Crack-Network*" classes, 60% of total images in each of these two classes are randomly selected for the training set, and 5% of them are selected for the test set.

Table 2: Description of the Second Sub-Dataset

Classes	Number of Images	% of the Total Dataset	Training Set			Test Set		
			Number of Samples	Percentage of Class	Percentage of Training Set	Number of Samples	Percentage of Class	Percentage of Test Set
Clean	206,199	70%	26,806	13%	32.2%	2,062	1%	25.6%
Patch	16,025	5.4%	11,218	70%	13.5%	1,282	8%	15.9%
Pothole	2,525	0.9%	2,146	85%	2.6%	379	15%	4.7%
Crack-Linear	29,459	10%	17,675	60%	21.2%	1,473	5%	18.3%
Crack-Network	35,007	11.9%	21,004	60%	25.2%	1,750	5%	21.8%
Marking	5,485	1.9%	4,388	80%	5.3%	1,097	20%	13.6%
Total	294,700	100%	83,237	-	100%	8,043	-	100%

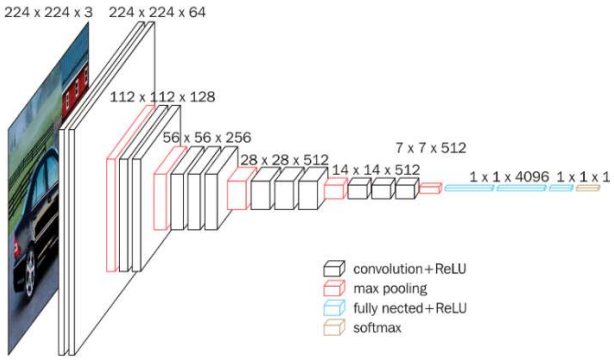
3.2.3 Deep Neural Network Structure

This research focused on implementing deep neural networks to build an automated distress detection and classification system. The Convolutional Neural Networks (CNNs) are very popular in image classification tasks and their promising performance is proven in a variety of applications. Accordingly, CNNs with different structures and combinations of layers are developed and tested with the pavement dataset collected for this research. The proposed CNN models are mainly based on VGG networks [38]. Nevertheless, the original VGG was developed for a complicated real-world image classification tasks and had more than 138 million parameters in its 16 layers. Respect to the size of the collected pavement dataset and the number of classes (six classes), the original VGG network seems to be extremely big for this application and needs to be shrunk.

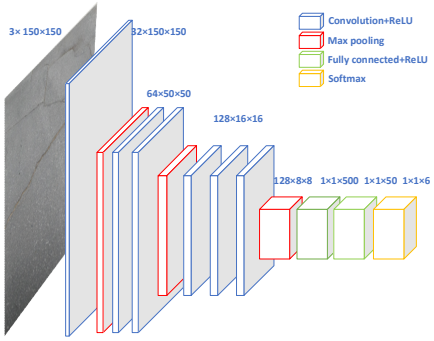
Figure 3 (a) illustrates an example of a VGG16 network with 13 convolutional layers, five pooling layers and three fully connected layers [38], implemented for *ImageNet Large Scale Visual Recognition Competition* (ILSVRC, 2015) while the input image size is 224×224×3 and the number of classes is 1000. The structure of VGG16 [38] has really common and simple features: (i) All convolutional layers have same convolutional kernel size (3×3) and small stride, usually equal to 1×1; (ii) All pooling layers have same kernel size (2×2) and (iii) the whole network is divided into five blocks which each block includes a

max-pooling layer followed by a few convolutional layers. Meanwhile, convolutional layers in every block have the same channels (number of filters). On the other hand, there are some shortcomings because of the large number of parameters: (i) taking long times for training and tuning; (ii) requiring a big dataset for training; (iii) requiring a big memory size.

In this research, different VGG-based CNN models were tested. Concerning the size of the collected dataset and tuning results, it was found that the deeper CNN models could not improve classification accuracy but consumed more computation power and time. As a result, instead of using the original VGG16 network with about 65 million parameters, a nine-layer deep neural network with about 4.3 million parameters is implemented. Figure 3 (b) shows an example of a convolutional neural network for pavement distress classification proposed in this research while the input image size is 150×150×3 and the number of classes is six. This network is built for the second and third datasets in Table 1. For the first and fourth datasets in Table 1, only the size of the input image will change.



(a) An example of VGG16 while the input image size is 224×224×3 [38]



(b) The proposed CNN while the input image size is 150×150×3

Figure 3: The structures of the reference CNN and the proposed CNN

In CNN models, the convolutional layer mainly includes feature maps, kernels, and padding. The first feature map is the original image. The kernel is a filter whose size and stride steps are hyper-parameters and each kernel only processes over its receptive field. The padding is used to preserve the input size in cases that some part of the receptive field of the kernel is on feature map edge and the other part of it is not pointing anywhere.

Following some of the convolutional layers, there is a max-pooling layer. They reduce the size of the feature map by combining the outputs of neurons group at the previous layer into a single neuron in the next layer. It is done by picking the maximum value of a 2×2

window and inserting these windows in the reduced feature map. Each max-pooling layer splits the whole network to different blocks with the same feature map's size within each block. After the last pooling layer, there is a fully connected block with three layers.

In the proposed CNN, the Rectified Linear Unit (ReLU) function (1) is used as the activation function for all the convolutional layers and the first two layers of the fully connected network [36].

$$f(x_i) = \max(0, x_i) \quad (1)$$

Where x_i is the input to the activation unit of i^{th} neuron and $f(x_i)$ is the output of the same neuron. The *SoftMax* function (2) is used as the activation function of the last layer of the fully connected network. The number of neurons in the last layer is equal to the number of classes and each *SoftMax* function calculates the probability of its corresponding class. Then the class with the maximum probability is chosen as the decision or label [47].

$$P(Y = i|x_j) = g(x) = \frac{exp(x_i)}{\sum_{j=0}^{K} exp(x_j)} \quad (2)$$

Table 3 illustrates the proposed deep neural networks with three convolutional blocks: i) the first block includes only one convolutional layer with 32 channels followed by a max-pooling layer, ii) the second block has two convolutional layers with 64 channels followed by a max-pooling layer, and iii) the last block has three convolutional layers with 128 channels followed by a max-pooling layer. Every convolutional layer uses the same padding to handle the edge of the images or feature maps.

The output of the last convolutional block is flattened where it makes a vector of 8192 elements, and then is given to the last block, which is a fully connected network with three dense layers. The first and second dense layers of the fully connected network have 500 and 50, respectively. The output layer has six neurons, and each one corresponds to one of the six pavement distress types that were mentioned earlier in Section 3.2.1. Except for the output layer of the fully connected network that has a *SoftMax* activation function, all the other layers are activated with a *ReLU* function. Each output neurons represent the probability of belonging to the corresponding class and the neuron with the maximum value (the closest value to 1) is considered as the decision for that sample pavement image.

Table 3: The Proposed Deep Neural Networks with 150×150 Size Input

Layer Type	Filter Size	Number of Feature Maps	Stride	Output Size	Number of Parameters
Convolutional	3×3	32	1×1	32×150×150	(32×(3×3)) +32 = 320
Max-Pooling	3×3	-	1×1	32×50×50	-
Convolutional	3×3	64	1×1	64×50×50	((32×(3×3)+1)×64) = 18496
Convolutional	3×3	64	1×1	64×50×50	((64×(3×3)+1)×64) = 36928
Max-Pooling	3×3	-	1×1	64×16×16	
Convolutional	2×2	128	1×1	128×16×16	((64×(2×2)+1)×128) = 32896
Convolutional	2×2	128	1×1	128×16×16	((128×(2×2)+1)×128) = 65664
Convolutional	2×2	128	1×1	128×16×16	((128×(2×2)+1)×128) = 65664
Max-Pooling	2×2	-	1×1	128×8×8	-
Flattening	-	-	-	128×8×8=8192	-
Fully Connected	500	-	-	500	(8192×500) +500 = 4096500
Fully Connected	50	-	-	50	(500×50) +50 = 25050
Output	6	-	-	6	(50×6) +6 = 306
Total Number of Parameters	-	-	-	-	4,341,824

3.2.4 CNN Regularization Scenarios

In addition to the dataset partitioning and the network structure, tuning the regularization parameters of the proposed CNN is a key point for having a generalized model that can perform well on the test set or a new dataset. The two important and widely applied operations for the regularization of CNN are Dropout and Batch Normalization.

Dropout (DO) operation is used to avoid overfitting in neural networks by temporarily and randomly removing some of the learned parameters from the network [48]. This technique improves the performance of neural networks in a wide variety of applications, such as object classification, digit recognition, speech recognition, document classification, and analysis of computational biology data. However, Dropout also has some drawbacks; for instance, it generally prevents the network from fast learning and increases the training time. Therefore, the Dropout rate should be treated as a hyper-parameter and tuned by considering the trade-off between decreasing both overfitting and training time.

Batch Normalization (BN) was proposed by Ioffe & Szegedy in 2015 [49], one year after Dropout and as an alternative regularization method, decreasing the sharp impacts of the Dropout. This method deals with the internal covariate shift, which was defined from the fact that the distribution of each layer's inputs changes during training, as the parameters of the previous layers change by normalizing layer inputs [50]. In the training stage, the distribution of the input variable of a neuron may tend to the top or the bottom of the non-linear Stochastic Gradient Descent (SGD) function. These are the saturation zones where the

covariate shift will vanish the gradient, decrease the learning speed, and eventually increase the convergence time. Normalizing activation functions throughout the network prevents the training from getting stuck in the saturated regimes of nonlinearities.

In addition to Dropout and Batch Normalization, Image Augmentation (IA) is also employed. Although it is not a regularization method, it helps to avoid overfitting [51]. Plausibly, any classifier performs better on a dataset with lots of images than one with a limited number of images. If the dataset covers all the possible scenarios, the chance of overfitting toward a specific class is reduced. In those cases that collecting more images is not feasible, Image Augmentation can be helpful. An Image Augmentation technique modifies an image and inserts it into the dataset as a new image. Flipping on vertical or horizontal edges, rescaling, rotating, whitening, and shifting are some examples of these techniques. Additionally, using Image Augmentation will improve the classifier performance over the test set, because the augmented images add unseen patterns into the training set that might already exist in the test set.

In this research work, six scenarios, represented in Table 4, are defined to evaluate the effects of using different regularization policies. At first, scenarios 1 and 2 are modeled to compare the influence of different Dropout rates applied to fully connected layers. Then the scenarios 3 and 4 are modeled to compare the performance of applying different Dropout rates to the pooling layers. In scenario 5, only Batch Normalization is applied to the convolutional and fully connected layers, and in scenario 6, the only Dropout is applied to the max-pooling layers. The results of different scenarios are demonstrated and discussed in Section 3.3.4.

Table 4: Regularization Scenarios

Layer	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5	Scenario 6
Convolutional	BN	BN	BN	BN	BN	-
Max-Pooling	-	-	DO (rate=0.5)	DO (rate=0.35)	-	DO (rate=0.5)
Convolutional	BN	BN	BN	BN	BN	-
Convolutional	BN	BN	BN	BN	BN	-
Max-Pooling	-	-	DO (rate=0.5)	DO (rate=0.35)	-	DO (rate=0.5)
Convolutional	BN	BN	BN	BN	BN	-
Convolutional	BN	BN	BN	BN	BN	-
Convolutional	BN	BN	BN	BN	-	-
Max-Pooling	-	-	DO (rate=0.5)	DO (rate=0.35)	-	DO (rate=0.5)
Flattening	-	-	-	-	-	-
Fully Connected	DO (rate=0.5) BN	DO (rate=0.35) BN	DO (rate=0.5) BN	DO (rate=0.35) BN	- BN	DO (rate=0.5) -
Fully Connected	DO (rate=0.5) BN	DO (rate=0.35) BN	DO (rate=0.5) BN	DO (rate=0.35) BN	- BN	DO (rate=0.5) -
Output	-	-	-	-	-	-

3.3 Experimental Results & Performance Evaluation

The performance of the proposed Convolutional Neural Network (CNN) in pavement distress detection and classification is evaluated by applying CNN to collected and labeled images in Montreal Pavement Dataset (MPD). For the evaluation, the following elements are taken into account.

3.3.1 System Setup and Network Description

In this research, Keras, with TensorFlow backend [52], is used to train and test different network structures and to tune all the hyperparameters. The average time to train a deep neural network model on the available GPU (GeForce GTX1060 3GB) was two days, while it takes one minute to process every batch of data. The parameters of the deep neural network are obtained by optimizing the cost function using the Adadelta method [53]. Adadelta uses an adaptive learning rate method and has a lower computational cost than Stochastic Gradient Descent (SGD). The *class weight* function is used to balance un-evenly sampled data during the training, respect to the size of each class. Moreover, the data augmentation powered with Keras is employed to gain more images by using rotation, vertical flip, and horizontal flip functions. These functions keep most details of the images without losing important information. For comparison, six deep neural models with the same structures mentioned in Table 3, but different regularization scenarios are trained and evaluated in this work.

3.3.2 Distress type detection and classification

Figure 4 shows two examples of distress type detections: Figure 4 (a) shows the pavement with a "*Crack-Network"* and a "*Pothole"*. There are also some "*Crack-Linear"* and "*Patching"*; Figure 4 (b) shows the annotated images. The yellow color polygons are for *Crack-Network*, the light green polygons are for "*Crack-Linear"*, the dark blue polygons are for the *Patching*, and the dark green polygons are for "*Pothole"*; Figure 4 (c) is the partitioned label image. The original label image is partitioned in 10×14 sub-images and then for each sub-image, the labels are chosen as a label of the square segment – note that images are most repeated in the segment of 150×150; Finally, Figure 4 (d) shows the output of the proposed CNN with 10×14 sub-images that are concatenated, making a single image. Figure 4 (e-h) show another example of distress type detection. Comparing figures (c) by (d) and comparing (g) by (h) generates the true or false positive or negative values used for producing the error measures.

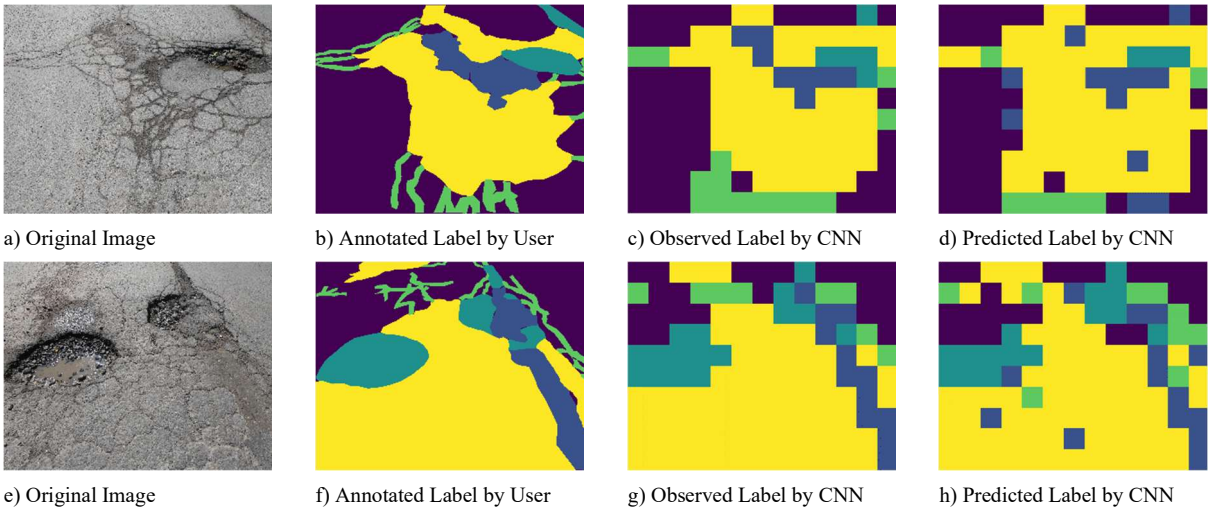


Figure 4: Two samples of distress type detection.

3.3.3 Error Measures

To evaluate the performance of the proposed neural network in the Montreal Pavement Distress Detection application, a variety of error measures are employed or defined. Table 5 mentions the name of each measure, their abbreviation, and their formula.

These definitions use the concept of positivity and negativity of image classes. In this paper, the clean images are considered as a negative class and all the other distress types are considered as positive classes. However, there are five different positive classes, as five

different distress types are distinguished. In this context, true negative (t_n) is the number of the clean images that are predicted correctly, false positive (f_p) is the number of clean images that are predicted as positive or having distress, and false negative (f_n) is the number of positive images with any kind of distress that are predicted as clean. Additionally, true positive (t_p) is split into two sub-factors: true positive-true classified (t_{p-t_c}) is the number of images that have been predicted as positive and the type of distress is detected correctly, and true positive-false classified (t_{p-f_c}) is the number of positive predicted images that are classified as a wrong distress type. This split helps to identify the resources of error more accurately. By considering these five factors, nine different error measures are defined, and all error measures can be calculated for each class individually or of the entire data.

Table 5: Error Measures

Recall or True Positive Rate (TPR)	**False Negative Rate (FNR)**	**False Classification Rate (FCR)**
$\frac{t_{p-t_c}}{\boldsymbol{observed\ positive}}$	$\frac{f_n}{observed\ positive}$	$\frac{t_{p-f_c}}{observed\ positive}$
Precision or Positive Predictive Value (PPV)	False Discovery Rate (FDR)	Positive Discovery-False Classification (PDFC)
$\frac{t_{p-t_c}}{\boldsymbol{predicted\ positive}}$	$\frac{f_p}{predicted\ positive}$	$\frac{t_{p-f_c}}{predicted\ positive}$
True Negative Rate (TNR)	F1-Score	Accuracy (ACC)
$\frac{t_n}{\boldsymbol{observed\ negative}}$	$2\times\frac{PPV\ \times TPR}{PPV+TPR}=\frac{1}{\frac{1}{2}\times\left(\frac{1}{TPR}+\frac{1}{PPV}\right)}$	$\frac{t_{p-t_c}+t_n}{total\ samples}$

The main four performance factors that are discussed throughout this paper are *Recall* or TPR, *Precision* or PPV, *F1-Score,* and *Accuracy*. The F1-Score is the harmonic mean of Recall and Precision. The main two error factors are the False Negative Rate and False Classification Rate. In General, for observed positive samples: $TPR + FNR + FCR = 1$ and for predicted positive samples; $PPV + FDR + PDFC = 1$.

3.3.4 Regularization Evaluation

In fine-tuning the deep-neural network for this application, there are many factors to be considered, including the network structure, the input image size and partitioning, and tuning the regularization hyperparameters. The six regularization scenarios, mentioned in Section 3.2.4, were simulated and results over the second partitioning scenario, splitting images to 150×150 (A), are reported and discussed in Table 6. The fourth regularization scenario, using

Batch Normalization following every convolutional and fully connected layer and Dropout layers (rate= 0.35) following max-pooling and fully connected layers, has the best performance over the training and test sets among all the others.

The F1-Score and Accuracy results show that using only Dropout (sixth scenario) has the worst results. Adding Batch Normalization to the same structure of Dropout layers improves the third scenario outcomes. However, it does not provide a promising result with a high rate of 0.5. The results of first and second scenarios show that using only Batch Normalization in convolutional layers and Dropout in fully connected do not deliver promising results, especially in terms of TPR. Although using only Batch Normalization in all the layer (fifth scenario) has better results than the first and second scenarios, it still seems to suffer from the under-fitting problem since TNR is still low.

In Table 6, the arrows, next to the performance measures, show if greater (↑) or lower (↓) results are better. The highlight results in bold are the measures that the proposed model has achieved over the test set.

Table 6: Fine Tuning of Regularization Hyper-parameters.

		Error Measures								
Set	Scenario	TNR↑	TPR↑	FNR↓	FCR↓	PPV↑	FDR↓	PDFC↓	F1-Score↑	ACC↑
Training	1	0.911	0.759	0.054	0.187	0.768	0.043	0.189	0.764	0.808
	2	0.931	0.716	0.074	0.210	0.747	0.034	0.219	0.731	0.785
	3	0.805	0.607	0.103	0.290	0.614	0.094	0.293	0.610	0.671
	4	0.928	0.832	0.032	0.136	0.831	0.034	0.136	0.831	0.863
	5	0.874	0.713	0.074	0.212	0.724	0.061	0.215	0.718	0.765
	6	0.791	0.660	0.078	0.262	0.646	0.097	0.257	0.653	0.702
Test	1	0.910	0.740	0.062	0.198	0.764	0.032	0.204	0.752	0.784
	2	0.929	0.681	0.085	0.234	0.725	0.026	0.249	0.702	0.744
	3	0.806	0.598	0.108	0.293	0.624	0.070	0.306	0.611	0.651
	4	0.930	0.807	0.041	0.152	0.821	0.025	0.155	0.814	0.838
	5	0.876	0.732	0.068	0.201	0.750	0.044	0.206	0.741	0.769
	6	0.796	0.652	0.083	0.266	0.660	0.071	0.269	0.656	0.689

3.3.5 Input Size Effects

Choosing the hyper-parameters for resizing and splitting original images is a trade-off between the number of produced sub-images and the level of information that each sub-image contains, which both have a direct impact on the performance of the model. In this section, the CNN model, with the same structure as Table 3, is trained with four datasets, obtained by

applying four different partitioning scenarios. In the first, second, and third datasets, the original images are resized to 1500×2000, 1500×2100 or 1600×2000; and then split to 100, 150, and 200 pixels (vertically and horizontally), respectively. Therefore, although the first dataset has many more images, those images have less information comparing to the two other datasets. In the third dataset, the images are resized to 750×1050 and then split to 150×150. This means that although each sub-image has more information comparing to any of the other three datasets, there are few images available for training and test set.

Table 7 shows the four datasets mentioned in Table 1 and the number of samples belonging to each class. Of all the sub-images of the first dataset, 187698 images are randomly selected for the first dataset:171,277 images for the training set and 16,420 images for the test set. This is because every original image has many areas without any distress types. Therefore, many of the sub-images belong to the "*Clean"* class, and adding all of them to the training/test set results in an un-balance dataset. Moreover, "*Crack"* classes have many samples and need random sampling to avoid having an unbalanced dataset. Similarly, 91280, 23722, and 53556 sub-images are randomly sampled from the pool of the second, third, and fourth datasets, respectively.

Table 7: The Dataset Description for All Dataset

Class	100×100		150×150 (A)		150×150 (B)		200×200	
	Training	Test	Training	Test	Training	Test	Training	Test
Clean	59282	4560	26806	2062	6484	499	14991	1153
Patch	23284	2661	11218	1282	2845	325	6597	754
Pothole	4393	775	2146	379	586	104	1283	226
Crack-Linear	33418	2785	17675	1473	4529	377	10774	898
Crack-Network	42515	3543	21004	1750	6035	503	12521	1043
Marking	8385	2096	4388	1097	1148	287	2653	663
Total per class	171277	16420	83237	8043	21627	2095	48819	4737
Total per dataset	187697		91280		23722		53556	

This section discusses the detailed results of applying the proposed CNN's structure (Table 3) with the fourth regularization scenario (Table 4) to the four different datasets of Table 7. Table 8 shows the error measures over the training and test sets. The second dataset 150×150 (A) has the best performance among the other datasets in terms of every error measure over the test set. After that, the fourth dataset 200×200 has the second-best performance. Between the first dataset (the one with many low-resolution sub-images) and the third dataset (the one with few high-resolution sub-images), the results do not suggest any

preference because both of them perform inadequately.

Table 8: Prediction Results of All Four Dataset.

		Error Measures								
Set	Sub-Image Size	TNR↑	TPR↑	FNR↓	FCR↓	PPV↑	FDR↓	PDFC↓	F1-Score↑	ACC↑
Training	100×100	0.837	0.659	0.087	0.254	0.660	0.086	0.254	0.659	0.721
	150×150 (A)	0.928	0.832	0.032	0.136	0.831	0.034	0.136	0.831	0.863
	150×150 (B)	0.848	0.689	0.095	0.216	0.710	0.067	0.223	0.699	0.736
	200×200	0.837	0.719	0.061	0.220	0.711	0.071	0.218	0.715	0.755
Test	100×100	0.847	0.655	0.083	0.261	0.672	0.060	0.268	0.664	0.709
	150×150 (A)	0.930	0.807	0.041	0.152	0.821	0.025	0.155	0.814	0.838
	150×150 (B)	0.832	0.674	0.111	0.216	0.715	0.056	0.229	0.694	0.711
	200×200	0.832	0.724	0.059	0.217	0.728	0.054	0.218	0.726	0.750

3.3.6 Distress Type Detection and Classification Performance

Based on the results of the two previous sections, the fourth regularization scenario with the input image size of 150×150(A) outperforms the other models. Instead of evaluating the models based on distress/clean detection, in this section, the results over the six different classes (Figure 1) are discussed. Table 9 shows the results of the class-wise performances of the proposed and tuned CNN over the test set. The first six columns and rows show the classification confusion matrix where the rows are the observed class and the columns are the predicted class. The first element, from the first column-first row, is the number of images that are correctly classified as "*Clean*" (true negative). The other elements of the first column report the number of images that have been classified as "*Clean"* class (false negative) and the first row reports the number of images that belong to the "*Clean"* class but are classified as others (false positive). The diagonal elements are the correct distress detection (images correctly classified as positive)), and the rest of the elements are true positive but false classified. By considering these factors, TNR, TPR (recall), FNR, FCR, PPV (precision), and F1-Score of each have been calculated.

The Ture Negative Rate (TNR) is about 93%, while the TPR of the classes ranges from 75.7% (*Pothole*) to 84.1% (*Patch*). The results over the test set show that the proposed system has an F1-Score around 86% for the "Marking" class, 80.2% for "Patch", 80.8% for "Pothole", and about 79.6% and 81.3% for the two "Crack-Linear" and "Crack-Network" classes.

Table 9: Detailed Performance Evaluation of CNN over Test Set-Separate Crack Classes

Predicted / Observed	Clean	Patch	Pothole	Crack-Linear	Crack-Network	Marking	TNR/TPR	FNR	FCR	PPV	F1-Score
Clean	1917	74	2	30	37	2	93.0%	-	-	-	-
Patch	98	1078	10	49	46	1	84.1%	7.6%	8.3%	76.7%	80.2%
Pothole	6	33	287	32	19	2	75.7%	1.6%	22.7%	86.7%	80.8%
Crack-Linear	46	49	10	1170	196	2	79.4%	3.1%	17.4%	79.9%	79.6%
Crack-Network	46	60	19	165	1454	6	83.1%	2.6%	14.3%	79.5%	81.3%
Marking	49	112	3	19	77	837	76.3%	4.5%	19.2%	98.5%	86.0%
Total	2162	1406	331	1465	1829	850	80.7%	4.1%	15.2%	82.1%	81.4%

The possible errors regarding the *Recall* value, $1 - Recall$ or $1 - TPR$, are split into two categories: *False Negative Rate* and *False Classification Rate*. For example, the TPR of the "*Crack-Linear"* is 79.4%, and the 21.6% error is split to 3.1% for FNR and 17.4% for FCR. This amount of false classification rate is because of having a "*Crack-Network"* class where many samples of these two class, "*Linear"* and "*Network"*, have been classified in one another. It seems that most of the performance degradation for the two *"Crack"* classes is because of misclassification between these two classes.

Therefore, the "*Crack-Linear"* and "*Crack-Network"* classes are merged and created an additional class, called "*Crack-Total*" class. The result, with regard to this merging, is presented in Table 10 and shows that the detection rate (TPR) of any type of cracks is 92.6%. Although, the proposed system can distinguish all six classes with a total *recall* value of 80.7% and total *F1-Score* of 81.4%, merging two similar classes of cracks increases total *Recall* value to 86.7% and total *F1-Score* to 87.5%.

Table 10: Detailed Performance Evaluation of CNN over Test Set-Merged Crack classes

Predicted / Observed	Clean	Patch	Pothole	Crack-Total	Marking	TPR	FNR	FCR	PPV	F1-Score
Clean	1917	74	2	67	2	93.0%	-	-	-	-
Patch	98	1078	10	95	1	84.1%	7.6%	8.3%	76.7%	80.2%
Pothole	6	33	287	51	2	75.7%	1.6%	22.7%	86.7%	80.8%
Crack-Total	92	109	29	2985	8	92.6%	2.9%	4.5%	90.6%	91.6%
Marking	49	112	3	96	837	76.3%	4.5%	19.2%	98.5%	86.0%
Total	2162	1406	331	3294	850	86.7%	4.1%	9.2%	88.2%	87.5%

3.4 Conclusion

Efficient monitoring systems can help reduce social and economic costs. Automated video-based pavement monitoring solutions are crucial in road maintenance operations and programs.

This research proposes a simple, low-cost, and robust methodology for pavement image distress classification using CNN algorithms. Data was collected using low-cost, high-resolution cameras mounted in a vehicle. The labeled dataset includes 2,105 FHD images selected from 12,000 images and including six common distress types. In the detection process, the calibrated CNN model can not only detect whether there is a crack or not but also classify the type of distress efficiently. From the result, we can see the TPR of the model is 75.7% for “Pothole”, 84.1% for “Patch”, 76.3% for “Marking”, 79.4% for “Crack-Linear” and 83.1% for “Crack-Network”. However, by merging linear and network crack classes, the accuracy over the merged class increases to 92.6%. These results are slightly better for those published recently in the literature.

Different regularizations were implemented and compared, including image augmentation, Dropout, and Batch Normalization. From the result, we find that the model with Batch Normalization and a lower-Dropout rate achieves the best performance. This shows that Batch Normalization can be used as a Dropout to reduce overfitting or can be combined with Batch Normalization using a lower Dropout rate to reduce training time. In comparison with input sizes, it shows that compressed resolution and split size affect the prediction results. Therefore, the selection of the proper size and resolution input image is very important before the training process. Overall, this work shows that deep neural networks integrated into embedded systems can integrated into pavement monitoring vehicles and used for automated pavement distress type detection and classification in road facilities.

For future work, we plan to collect additional video to improve training datasets and models. A variety of road facilities can be included, such as bicycle facilities and highways. Despite the relatively large datasets used in this work, a more balanced dataset of potholes could improve the accuracy of this category. Moreover, image datasets from both regular (visual spectrum) and 3D LiDAR sensors could be combined to capture the depth of cracks or 3D shape of potholes. Additional future work includes the comparison of the VGG model with other modern methods such as Google inception net and ResNet should be tested in next step. The use of GANs could also be interesting to generate new images and compare them with the traditional image generator used in this work. Finally, a topic that was not addressed in this work is the detection of road surface conditions during wintertime. In addition to the pavement distress conditions, the cold winter in Canada deteriorates surface conditions under

the presence of ices and snows, precipitation, and their combinations. Similar to distress detection and classification, algorithms for detecting winter surface conditions can be built using a similar approach – combining regular video and thermal images.

4 Winter Road Surface Conditions Classification using Convolutional Neural Network (CNN): Visible - Light and Thermal Images Fusion

4.1 Introduction

Winter road surface condition plays a key role in traffic efficiency and safety, especially for the countries with long winter and harsh climate. Research literature shows that there is a relationship between road surface conditions and crashes. Driving conditions in winter often deteriorate during snowfall and ice formation, causing a significant reduction in pavement friction and increasing the risk of collision [3]. According to statistics from the America Department of Transportation, almost 26% of traffic accidents in America happen on snowy, icy, wet or slushy roads and 18% crash fatalities happen during the slick winter weather [5].

To maintain appropriate road surface conditions and road safety during wintertime, winter maintenance operations (plowing, salting or sanding, snow removal) are critical. However, winter maintenance operations also incur high monetary costs and negative environmental effects. Every year, the governments spend considerable budget on plowing, salting and sanding of the roads to increase road robustness against freezing. For example, the direct cost of winter maintenance programs in Ontario is estimated to exceed $100 million annually [54]. As a result of a long snowy winter with low-temperature weather, roads in cold-winter regions can freeze for a relatively long time, which reduces the road surface friction significantly [55]. Because of the lower friction, the vehicles' stopping distance and time increases. Besides, the glare of the snow on the road causes snow blindness.

Road surface conditions during winter are particularly an issue in Canadian urban areas likes Montreal, with an extremely long and cold winter and a large network to maintain. According to the government weather report, the winter in Montreal lasts for five months, with an average high temperature of -2.3 °C (daytime) and an average low of -8.9 °C (nighttime). Furthermore, and due to the continental climate, Montreal has more than one hundred snowy days with more than one cm of average snow depth. In mid-winter, the snowpack average goes up to13 cm deep. To ensure road user's safety and traffic operations, cities like Montreal spent a significant number of resources to maintain appropriate surface conditions during winter. For instance, in 2018, the City of Montreal winter maintenance

budget was announced as $192 million [17].

To facilitate surface monitoring during wintertime, automated systems for collecting winter road surface information have emerged in the last years. By gathering and providing real-time road surface information, winter road maintenance operators and on-vehicle and on-road warning systems can reduce risks related to frozen and slippery roads. For instance, using data from automatic monitoring systems, winter operators can better plan and implement anti-freezing (anti-icing) procedures and control the dosage chemical solutions to improve the efficiency of winter operations.

In the last years, some research works have proposed monitoring systems and automated image processing methods based on camera-based systems and computer-vision image processing. One of the main challenges of current systems is the detection of ice and snow on the road surface, for example, Zhang et al. [22] proposed a video monitoring system and combined it with Support Vector Machine and K-nearest neighbors, which performed very well in classified snow and dry. However, their result also indicated that their model could not distinguish wet and icy efficiently. To address this issue, few studies have used near-infrared detectors: Jonsson et al. [28] employed a near-infrared camera and halogen searchlights in their system and reached high accuracy in distinguishing dry, wet, ice and snow conditions. Despite the recent developments, to our knowledge, very little research work has been documented on the performance of thermal video cameras for surface condition monitoring. The comparative performance alone and in combination with regular (visual spectrum) has not been studied. Moreover, the use of CNN method in the context of winter surface monitoring is relatively new, although Guangyuan et al. [23] have applied the CNN with their camera-based system, the limited number of images in their dataset restricted the model performance. Hence, a qualified dataset with a large number of images is vital for CNN models.

This research proposes a solution for automatically monitoring winter road conditions using machine-learning techniques and visible-light and thermal road images. More specifically, this research trains a Convolutional Neural Network (CNN) model based on thermal and regular video cameras to detect and classify images with icy and snow presence automatically. The proposed solution integrates both a regular (GoPro) camera to capture visible light range images and a ThermiCam to capture infrared range or thermal images. For

dataset collection, two cameras are mounted on a vehicle and videos were collected while the vehicle was cruising Montreal streets in winter. As part of this work, different convolutional neural network structures were tested on the training dataset to build a promising winter road condition classifier and to analyze the impacts of each image source, including the performance of visible light and thermal images.

Followed by the introduction, a literature review discussion is offered in the following sections regarding winter road condition monitoring systems, data collection systems, especially the camera-based system, and the development of computer vision and machine learning algorithms.

4.2 Methodology

The methodology proposed for this research includes different steps: 1) data-collection system, 2) data collection and preparation, 3) training and testing of CNN models. The details of the elements involved in this research are detailed as follows.

4.2.1 Data collection system

For this research, the ThermiCam Wide, a thermal camera built by FLIR and shown in Figure 5, was used to collect images in infrared spectrum, which are also referred as thermal images. The thermal videos were captured 15 frames per second from RGB channels with an output resolution of 368×296 pixels. The temperature, captured by this thermal camera, ranges from -34℃ to 74℃.

Figure 5: ThermiCam Wide-build by FLIR

Additionally, a GoPro Hero 7 camera is integrated to collect images in the visible spectrum, which are also referred as RGB images. GoPro Hero 7 has a smooth function, which reduced the effects of vibrations caused by the vehicle's movements when collecting images. The RGB video data was collected as 30 frames per second (fps) with an output

resolution of 1920×1080 pixels.

The GoPro is controlled by a smartphone via wireless LAN, while the thermal camera is connected to an on-board computer with a wired LAN. Both cameras, visible spectrum and infrared, were installed in the front of a vehicle (instead of the back) to avoid the heat generated by vehicle exhaust pollution, which may affect images, especially in infrared spectrum. Figure 6 shows the setup of the thermal camera installed on the left side of the vehicle front (from the driver's perspective). This setup covers the road lane centerline. A visible spectrum camera is installed next to the thermal camera without blocking its field of view. Both viewing-camera angles were pointed in a way that both collect images for the same road surface area.

1.

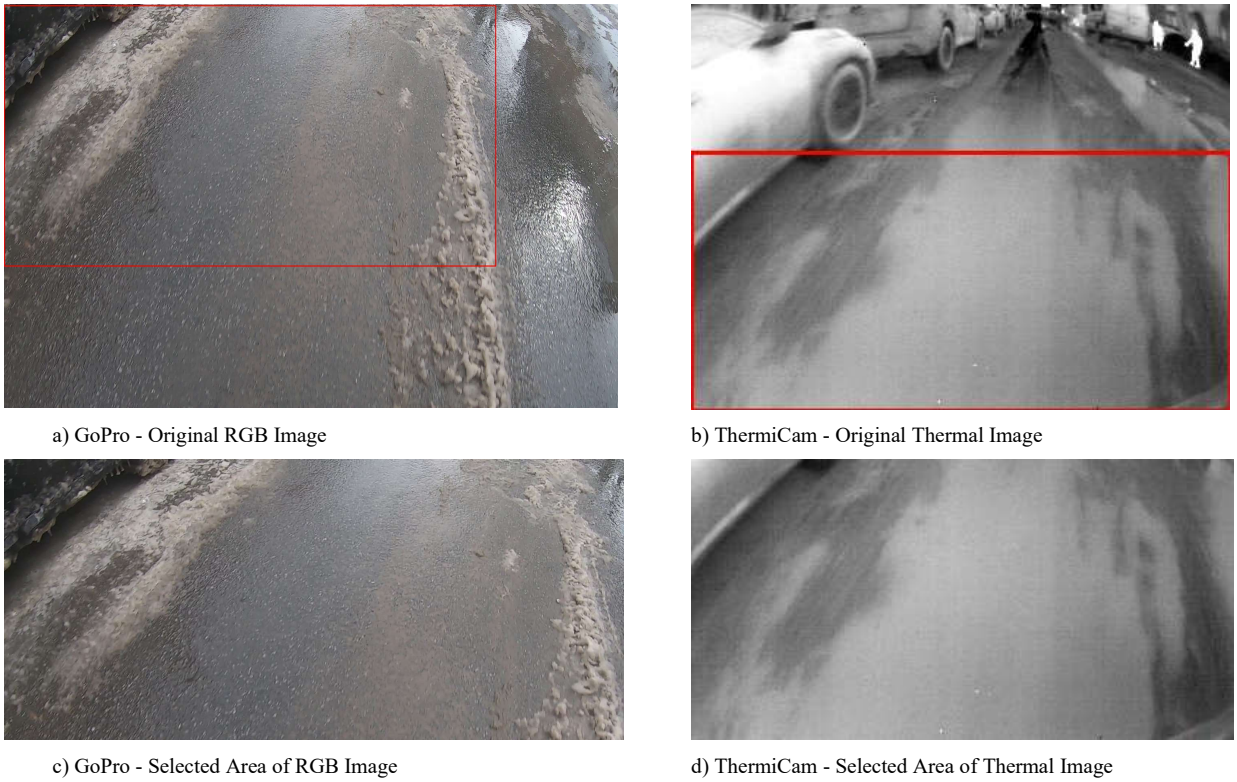
Figure 6: Thermal camera setup (front-view and side-view)

4.2.2 Dataset collection and preparation

The video data was collected on February 28, 2020, while the weather was cloudy, without significant precipitation, and the temperature ranged from -9°C to -6°C. However, the day before data-collection, snow precipitation occurs in Montreal for a long period. The dataset was collected for more than three hours in a diversity of street types in Montreal. During this period, 14 videos were collected with a size of 25.6 GB from the GoPro camera. ThermiCam videos occupied 1.59 GB of the storage. During the data collection, the speed of the inspection vehicle was considerably low. After the video data collected, both GoPro video and ThermiCam video were split into images for one frame per second. Finally, from all the extracted images, 4244 images were manually selected and added to the database.

Although both cameras were set to focusing on the same area, a pixel-by-pixel mapping (matching) between two RGB and thermal images is required to find the exact

overlapping area. Since the cameras' installation was fixed, the relative mapping between thermal and RGB images is the same in all the images. Figure 7 (a) and Figure 7 (b) show the original image of RGB and thermal cameras. The red boxes show the overlapping areas. This mapping was done manually by checking multiple pairs of images and finding the four corners of the box. After finding the pixel's coordinates of the four corners of each box, all the images were cropped and saved into the training dataset. The images after matching and cropping are shown in Figure 7 (c&d).



a) GoPro - Original RGB Image

b) ThermiCam - Original Thermal Image

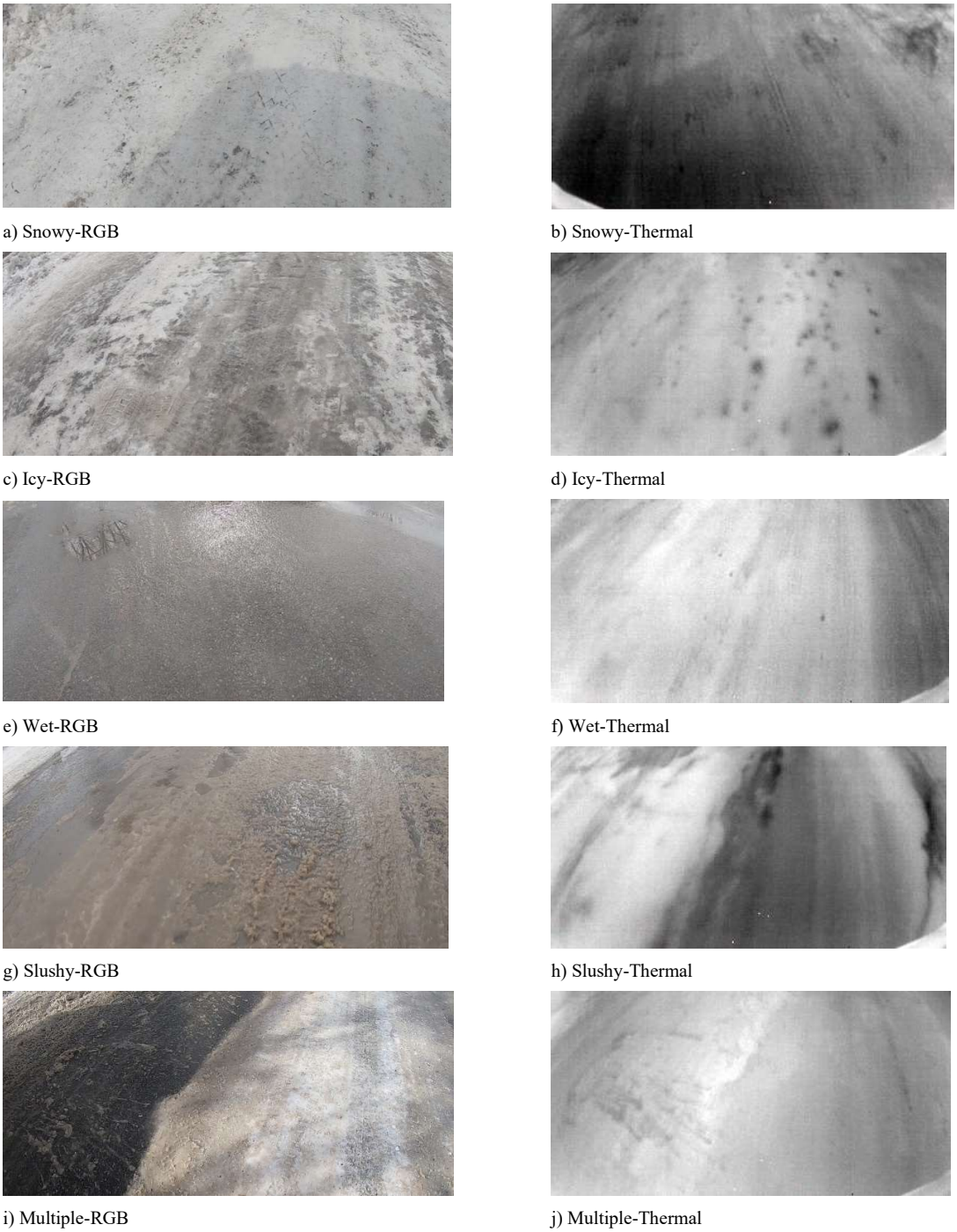
c) GoPro - Selected Area of RGB Image

d) ThermiCam - Selected Area of Thermal Image

Figure 7: The original images with detection box and matching pixels

The size of the cropped thermal image was 188×368 pixels. After cropping the RGB images, they were resized to 188×368 pixels to be the same size as the thermal images. Afterward, the pair of RGB and thermal images were manually labeled into four classes: *Snowy*, *Icy*, *Wet*, and *Slushy*. Both the RGB details and thermal details are considered to label each pair of images. Figure 8 shows samples of the pair of images of each labeled class. The *Snowy* class, shown in Figure 8 (a & b), means that the road was covered by the snow, the color in GoPro image is white and in thermal image is lighter compared to icy. The *Icy* class, shown in Figure 8 (c & d), means that the road was covered by the ice - transparent in GoPro image and darker in thermal image. The *Wet* class shown in Figure 8 (e & f) is clean road

with no clear barrier but only water. The *Slushy* class shown in Figure 8 (g & h) means that the road was covered by black or brown ice or snow, and perhaps melted water because of the vehicle.



a) Snowy-RGB b) Snowy-Thermal

c) Icy-RGB d) Icy-Thermal

e) Wet-RGB f) Wet-Thermal

g) Slushy-RGB h) Slushy-Thermal

i) Multiple-RGB j) Multiple-Thermal

Figure 8: GoPro image(left) and thermal image(right) for each class

Table 11 shows the details of the sample distribution of each class. Due to the road

condition during the data collection, the samples are un-evenly distributed in the four selected classes. The *Icy* and *Wet* classes contribute to 38.5% and 35.4% of the total samples, respectively, while the *Snowy* class has the lowest number of samples supplying 6.7% of the total samples. However, about 80% of the samples of each class were partitioned as training set, and the remaining 20% were partitioned as test set, except for the *Snowy* class, which partitioning factors are 82.6/17.4%.

Additionally, the distribution percentage of each class in the training and testing set has been presented in Table 11, where for example Snowy class forms the 7% of the training set and 5.9% of the test set, while the Icy class forms the 38.3% of the training and 39.3% of the test set.

Table 11: Base Image Dataset

Class	Number of Images	% of the Total Dataset	Training Set			Test Set		
			Number of Samples	% of Class	% of Training Set	Number of Samples	% of Class	% of Test Set
snowy	288	6.78	238	82.6	7.0	50	17.4	5.9
icy	1636	38.54	1302	79.6	38.3	334	20.4	39.3
wet	1505	35.46	1204	80.0	35.5	301	20.0	35.5
slushy	815	19.20	651	79.9	19.2	164	20.1	19.3
Sum	4244	100	3395	79.99	100	849	20.00	100

Meanwhile, alternative datasets have been built and used throughout this research. Table 12 summarize the details of the base dataset and its three alternatives. The three alternative datasets are listed as follow:

The *Multiple* dataset: In addition to the four aforementioned classes, an additional class called *Multiple* is inserted (Figure 8 (i & j)). This class includes images with multiple or mixed surface conditions and has images with both snowy and icy, both snowy and slushy, both wet and slushy, or both wet and icy

The *Artificial* dataset: In order to deal with the uneven sample distribution of the classes, especially the lack of snowy images, an image generator was applied. The image generator is based on Open Source Computer Vision Library (OpenCV), which processes the input images by flip, rotate, mask adding, cropping and filter with random possibilities [56]. As a result, 241 artificial snowy images were built from available snowy images. Therefore, by adding these images to the *Artificial* dataset, the *Snowy* class has 529 samples in total.

The *Split* dataset: the image size of the *Base* dataset is 188×368. To reduce the number of model's parameters and increase the samples, the original images were horizontally split to

two equal size images of size of 188×184.

Table 12: Summary of Class Distributions of the Four Datasets

Classes	Base			Artificial			Multiple			Split		
	Sum	Training	Test	Sum	Training	Test	Sum	Training	Test	Sum	Training	Test
Snowy	288	238	50	529	408	121	288	223	65	576	459	117
Icy	1636	1302	334	1636	1314	322	1636	1319	317	3272	2629	643
Wet	1505	1204	301	1505	1208	297	1505	1228	277	3010	2407	603
Slushy	815	651	164	815	658	157	815	634	181	1630	1295	335
Multiple	-	-	-	-	-	-	551	432	119	-	-	-
Sum	4244	3395	849	4485	3588	897	4795	3836	959	8488	6790	1698

4.2.3 CNN Models implementation

This paper integrates CNN models to build an automated winter road detection and classification system. The CNN models are popular for image classification applications. Their promising performance has been proven in a variety of applications. The proposed model is an extension of the VGG network [57].

In the proposed CNN, there are four types of layers were used, including convolutional, max-pooling, flattening, and fully-connected layers. Each convolutional layer mainly includes feature maps, kernels, and padding. The first feature map is the original image, the kernel is kind of filter whose size and stride steps are hyper-parameters and each kernel will only process over its receptive field, and the padding is used to preserve the input size in cases that some part of the receptive field of the kernel is outside the feature map boundary. Some of the convolutional layers are followed by a max-pooling layer that builds a reduced size feature map by combining the outputs of a group of neurons at the previous convolutional layer into a single neuron in the next layer. This is done by picking the maximum value of a window (with the size of 3×3 or 2×2) and inserting it in the reduced feature map. After the last pooling layer, there is a flattening layer that flattens the two- or three-dimensional feature maps into a one-dimensional array (vector) and passes the vector to the fully connected layers. The fully connected layers are used to abstract the feature map into a vector with the size of the number of classes, which shows the probability of belonging to each class.

The proposed model followed the structure of the VGG network except for the number of layers and parameter size. The proposed network has three blocks. The first block includes a convolution layer with 16 kernels (3×3 receptive field) and a max-pooling layer (3×3 pool size). The second block includes a convolution layer with 32 kernels (3×3 receptive field) and

a max-pooling layer (3×3 pool size). The third block includes 2 convolution layers with 32 kernels (3×3 receptive field) and a max-pooling layer (3×3 pool size). The Rectified Linear Unit (ReLU), formulated in (3), has been used as the activation function of the convolutional layers [17].

$$f(x_i) = \max(0, x_i) \tag{3}$$

x_i is the output of the feature map and input to the i^{th} neuron of the activation map, and $f(x_i)$ is the output of the same activation unit. Compared to other activation functions, *ReLU* is sparser, better gradient propagation and efficient computation.

The SoftMax function [47], formulated in (4), is used in the neurons of the last fully connected layer. The number of neurons in the last layer is equal to the number of classes and each SoftMax function calculates the probability of its corresponding class. Then the class with the maximum probability will be chosen as the decision or label.

$$P(Y = i|x_j) = g(x_i) = \frac{exp(x_i)}{\sum_{j=0}^{K} exp(x_j)} \tag{4}$$

where x_i is the input to the activation unit of i^{th} neuron, $g(x_i)$ is the output probability of i^{th} class.

To avoid overfitting in early iterations, two regularization methods, Dropout and Batch-Normalization, have been used in the structure of the proposed CNN. Dropout (DO) operation helps avoiding overfitting toward training set when the model is learning the parameters of the neural network by temporarily and randomly removing some of the learned parameters from the network [48] Batch-Normalization (BN) was proposed by Ioffe & Szegedy in 2015 [49] as an alternative regularization method. This method deals with the internal covariate shift and prevents the training from getting stuck in the saturated regimes of non-linearities [58]. In addition to Dropout and Batch Normalization, Image Augmentation (IA) has also been employed. IA is used to help the classifier work better on the limited number of images by modifying an image and inserts it into the dataset as a new image. Flipping on vertical or horizontal edges, rescaling, rotating, whitening, and shifting are some examples of these techniques.

Three different CNN models have been applied to the collected datasets (from visible light and thermal cameras). First, only GoPro images are used from which a single stream CNN is built. Second, a CNN model for only thermal images is trained. Third, to capture the

combined both resources, GoPro and thermal images, the double stream CNN was applied. The double stream CNN has two independent input and convolutional blocks. The results of each stream are flattened, mixed, and passed to a fully connected block. A comparative analysis is then executed based on single-stream GoPro CNN, single-stream thermal CNN and double stream CNN of GoPro-ThermiCam. Both single steam CNN, either GoPro or ThermiCam, have the same structure and input size, which has been presented in Table 13.

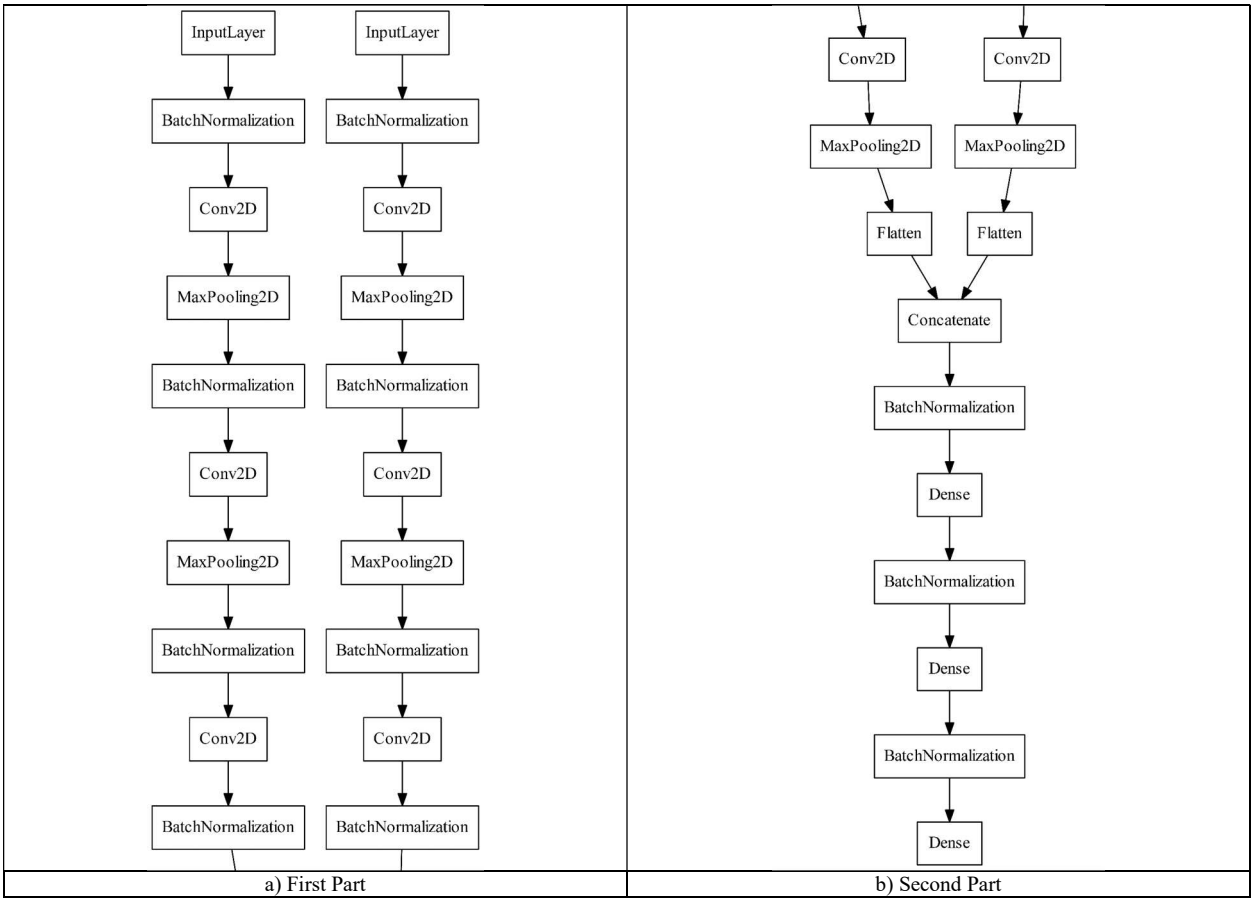
Table 13: The Proposed Single Input Deep Neural Networks with 188×368 Size Input

Layer Type	Filter Size	Number of Feature Maps	Stride	Output Size	Number of Parameters
BatchNormalization				1×188×368	1472
Convolutional	3×3	16	1×1	16×188×368	(16×(3×3)) +16 = 160
Max-Pooling	3×3	-	1×1	16×62×122	-
BatchNormalization				16×62×122	488
Convolutional	3×3	32	1×1	32×62×122	(32×(3×3)+1)×32) = 4640
Max-Pooling	3×3	-	1×1	32×20×40	-
BatchNormalization				32×20×40	160
Convolutional	3×3	32	1×1	32×20×40	((32×(3×3)+1)×32) = 9248
BatchNormalization				32×20×40	160
Convolutional	3×3	32	1×1	32×20×40	((32×(3×3)+1)×32) = 9248
Max-Pooling	3×3	-	1×1	32×6×13	-
Flattening	-	-	-	32×6×13=2496	-
BatchNormalization					9984
Fully Connected	500	-	-	500	(2496×500) +500 = 1248500
BatchNormalization					2000
Fully Connected	50	-	-	50	(500×50) +50 = 25050
BatchNormalization					200
Output	6	-	-	4	(50×4) +4 = 204
Total Number of Parameters	-	-	-	-	1,311,514

For combining the two technologies, a joint dataset made of GoPro images and ThermiCam images is used to build a double steam CNN model. The joint model has two independent convolutional streams, which is the same as the convolutional part of the single-stream network in Table 13. The last feature map of each stream is then flattened and joint together to build a one-dimensional feature vector of size 4992 elements, which is twice the flattened vector of each one stream CNN. Afterward, the flattened vector is given to a fully connected layer with more neurons than the single stream version. The structure of the double stream CNN is shown in Table 14 and Figure 9.

Table 14: The Proposed Double Stream Deep Neural Networks with 188×368 Size Input

Layer Type		Filter Size	Number of Feature Maps	Stride	Output Size	Number of Parameters
BatchNormalization_T	BatchNormalization_G				1×188×368	1472
Convolutional_T	Convolutional_G	3×3	16	1×1	16×188×368	160
Max-Pooling_T	Max-Pooling_G	3×3	-	1×1	16×62×122	-
BatchNormalization_T	BatchNormalization_G				16×62×122	488
Convolutional_T	Convolutional_G	3×3	32	1×1	32×62×122	4640
Max-Pooling_T	Max-Pooling_G	3×3	-	1×1	32×20×40	-
BatchNormalization_T	BatchNormalization_G				32×20×40	160
Convolutional_T	Convolutional_G	3×3	32	1×1	32×20×40	9248
BatchNormalization_T	BatchNormalization_G				32×20×40	160
Convolutional_T	Convolutional_G	3×3	32	1×1	32×20×40	9248
Max-Pooling_T	Max-Pooling_G	3×3	-	1×1	32×6×13	-
Flattening_T	Flattening_G	-	-	-	32×6×13x2=4992	-
BatchNormalization						19968
Fully Connected		500	-	-	500	(4992×500) +500 = 2496500
BatchNormalization						2000
Fully Connected		50	-	-	50	25050
BatchNormalization						200
Output		6	-	-	4	204
Total Number of Parameters		-	-	-	-	2,595,074



a) First Part b) Second Part

Figure 9. Structure of double stream Convolutional Neural Networks

A comparison between two single-stream networks and one double steams network is made in Section 4.3. Additionally, the double stream network with different weights for each of its inputs, ThermiCam and GoPro, is built and analyzed. To tune the weight of one input, the filter size of the last max-pooling layer was changed. For instance, if the filter size of the last max-pooling layer of GoPro stream is set to 5×4, the input size becomes 32×4×10 instead of 32×6×13. After flattening, the feature vector size is then reduced by 50% from 2496 to 1,280, which means that the weight of GoPro input is 0.5 or half of the ThermiCam input.

4.3 Results

In this work, Keras with TensorFlow [52] backend is used to implement CNN with different structures. As mentioned in Table 12, the training set of each dataset is used to train the model and tune its hyperparameters. The hyperparameters include learning rates, kernel size, learning iterations and batch size. The coefficients of the deep neural network was from optimizing the cost function using the Adadelta method [53]. Adadelta, which is embedded in Keras module, uses an adaptive learning rate and has a lower computational cost than Stochastic Gradient Descent (SGD). For this research, a GeForce GPU (GTX1060 3GB) is used to implement CNN with Keras, and the average time for training a single stream CNN model is developed on the base dataset on a GPU - this was 44 seconds per iteration and for training a double stream CNN model, this was 48 seconds per iteration. The results in this section are obtained after 300 iterations of training CNN model.

4.3.1 Performance Measures

To evaluate the proposed system performance, a variety of error measures including multiple-class average and weighted average *Precision*, *Recall* and *F1-Score* are employed. Table 15 shows the confusion matrix obtained by applying the learned and tuned double-stream CNN model on the test set of the base dataset. In the confusion matrix, the rows are the observed labels or real classes of each pair of RGB-Thermal images and the columns are the predicted labels by the CNN model.

The diagonal values show the number of true classification samples; for example, the value on the first row and the first column shows the total number of snowy images that are classified correctly - which is 42 in this case. The non-diagonal values show the misclassifications, for example, the value on the first row and the second column shows that eight samples were *Snowy* while they are classified as *Icy*. Similarly, the value on the second

row and the first column shows that five samples were Icy while have been classified as *Snowy*.

The precision of each class is the ratio of the correctly classified samples of that class to the total samples classified as that class. For example, the Precision of Snowy class is 42 divided by 47. The recall of each class is the ratio of correctly classified samples of that class to the total observed samples of that class. For example, the Recall of Snowy class is 42 divided by 50. The Precision and Recall can be calculated for each class individually. To simultaneously consider both Recall and Precision measures, the F1-Score (5) measure is used - this is the harmonic mean [59] of both.

$$F1\ Score\ = 2 \times \frac{Precision \times Recall}{Precision + Re} \tag{5}$$

To easily compare different CNN models, the aggregate of each measure is calculated in two ways. First, the average of each measure over four classes is calculated in formulas (6-8). For example, the average Precision is the sum of the precision values of all the classes divided by the number of classes.

$$Precision_{average}\ = \frac{\sum_{i=1}^{4} Precision_i}{\sum_{i=1}^{4} 1=4} \tag{6}$$

$$Recall_{average}\ = \frac{\sum_{i=1}^{4} Recall_i}{\sum_{i=1}^{4} 1=4} \tag{7}$$

$$F1 - Score_{average}\ = \frac{\sum_{i=1}^{4} F1-Score_i}{\sum_{i=1}^{4} 1=4} \tag{8}$$

Additionally, the weighted average of each measure is also calculated in formulas (9-11), where weights are proportional to the number of samples of each class. For example, the weight of the Snowy class is total snowy samples (50 samples in this case) divided by the total number of samples (849 samples in this case).

$$Precision_{weighted}\ = \frac{\sum_{i=1}^{4} Precision_i \times n_i}{\sum_{i=1}^{4} n_i} \tag{9}$$

$$Recall_{weighted}\ = \frac{\sum_{i=1}^{4} Recall_i \times n_i}{\sum_{i=1}^{4} n_i} \tag{10}$$

$$F1 - Score_{weighted}\ = \frac{\sum_{i=1}^{4} F1-Score_i \times n_i}{\sum_{i=1}^{4} n_i} \tag{11}$$

where i is the class ID and it ranges from 1 to 4 corresponding to *Snowy*, *Icy*, *Wet* and *Slushy* classes and n_i is the number of samples belonging to the i^{th} class.

Table 15: Confusion Matrix of Double Stream Network over the Base Test Set

		Classified (Predicted) Label					Error Measures		
		Snowy	Icy	Wet	Slushy	Total	Precision	Recall	F1-score
Observed Label	Snowy	42	8	0	0	50	0.894	0.840	0.866
	Icy	5	317	0	12	334	0.922	0.949	0.935
	Wet	0	2	297	2	301	0.983	0.987	0.985
	Slushy	0	17	5	142	164	0.910	0.866	0.888
	Total	47	344	302	156	849			
Average	-	-	-	-	-	-	0.927	0.910	0.918
Weighted Average	-	-	-	-	-	-	0.940	0.940	0.940

4.3.2 Input Configuration Evaluations

The developed CNN models, two single-stream models and one double stream model were fine-tuned and learned with the training set and evaluated over the test set. Table 16 shows the results of different input configurations. These include using only GoPro as the input data stream, using only ThermiCam, and using both resources in a two data streams configuration. In addition to the classification results of each configuration, the ground truth of training and test sets show the actual number of samples in each category. The size of the input images is 188×368 pixels and images are taken from the base dataset (Table 11).

The average and weighted values of precision, recall and F1-score indicate that the double data stream CNN model has the best result among the three configurations over the test set. This means that a combination of the visible light and thermal image resources provides more information for the CNN model regarding classifying. The detailed results of the double data stream CNN model are shown in Table 16.

Between single data stream CNN models, the model built by the GoPro data stream performs better than CNN built by using the ThermiCam data stream. Specifically, the true positive predicted image (TP) values imply that CNN model with GoPro images classifies "*Wet*", "*Slushy*" and "*Snowy*" better than CNN model with ThermiCam images. This happens since the images of these three classes are richer in color information. On the other hand, for the "*Icy*" patterns are better preserved in thermal images where the TP of the CNN model built by ThermiCam is higher than the other built by GoPro images.

Table 16: Prediction Results of Single Input Network and Multiple Input Network

Set	Scenarios	Predicted (TP)				Performance (Average)			Performance (Weighted)		
		Snowy	Icy	Wet	Slushy	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Train-ing	GoPro	236	1302	1204	651	1.000	0.998	0.999	0.999	0.999	0.999
	ThermiCam	235	1288	1198	641	0.990	0.990	0.990	0.990	0.990	0.990
	Double Stream	238	1302	1203	650	0.999	0.999	0.999	0.999	0.999	0.999
	Ground-truth	238	1302	1204	651	-	-	-	-	-	-
Test	GoPro	35	299	296	144	0.890	0.860	0.875	0.912	0.912	0.911
	ThermiCam	34	307	271	112	0.845	0.796	0.816	0.855	0.853	0.851
	Double Stream	42	317	297	142	0.927	0.910	0.918	0.930	0.940	0.940
	Ground-truth	50	334	301	164	-	-	-	-	-	-

4.3.3 Tuning Input Ratio for the Double Stream CNN model

In the double data stream CNN model, the ratio of combination of visible light images and thermal images can be changed based on the concept mentioned in section 3.4. Table 17shows the results of the same CNN configuration while the weights of each stream is reduced. First, the thermal stream weight is set to 0.5, which means that the number of neurons in the first layer of the fully connected block of CNN model (corresponding to thermal image stream) is reduced by 50%. Second, the visible light stream weight is set to 0.5 while the thermal stream weight is set to 1. Third, both stream weight is set to 0.5 – which means that the total number of neurons in the first layer of the fully connected block is reduced by 50%. The fourth row of Table 17 shows the results when both weights are set to 1 and no change is applied. These results are similar to those mentioned in Table 15 and Table 16.

The results show that reducing the weight of the thermal image stream increases slightly in the precision, recall and F1-score values over the test set since the true positive value of the *Snowy* and *Slushy* are slightly increased by 1 and 5 samples, respectively. Besides, decreasing the weight of both image streams increases the weighted precision, recall and F1-score by 0.8%, 0.8% and 0.7%, respectively. This suggests that the same CNN model with a smaller fully connected block, (i.e. halving the first layer of fully connected block) provides almost the same promising results. This can be interpreted as a sensitivity analysis over the size of the fully connected block of the double stream CNN model.

In general, reducing neurons of one data stream has no significant effect on the

performance measures suggesting that the other data stream compensates the possible error.

Table 17: Prediction Results of Different Input Weight of Multiple Input Network

Set	Stream with Reduced Neurons	Predicted (TP)				Performance (Average)			Performance (Weighted)		
		Snowy	Icy	Wet	Slushy	Precision	Recall	F1-score	Precision	Recall	F1-score
Train-ing	Thermal Stream	238	1301	1204	651	0.999	1.000	1.000	1.000	1.000	1.000
	GoPro Stream	238	1302	1204	651	1.000	1.0000	1.000	1.000	1.000	1.000
	Both Sreams	238	1300	1204	650	0.999	1.000	0.999	0.999	0.999	0.999
	None	238	1302	1203	650	0.999	0.999	0.999	0.999	0.999	0.999
	Ground-Truth	238	1302	1204	651	-	-	-	-	-	-
Test	Thermal Stream	43	315	296	147	0.923	0.921	0.922	0.943	0.943	0.943
	GoPro Stream	36	316	297	149	0.924	0.890	0.905	0.940	0.940	0.939
	Both Streams	45	318	298	143	0.936	0.929	0.932	0.948	0.948	0.947
	None	42	317	297	142	0.927	0.910	0.918	0.940	0.940	0.940
	Ground-truth	50	334	301	164	-	-	-	-	-	-

4.3.4 Adjusting input combination weights or double stream CNN model

The double data stream CNN model is chosen as the best model for further analysis over the different configuration on the database – in this model, the weights of the input combination are set to 1 for each of the image streams. In addition to the base dataset, three variations, mentioned in Table 12, are built and used for training and evaluating the selected CNN model. Table 18 shows the performance measures of the four different modeling scenarios. The F1-score of the CNN model over the test set of the base dataset is 94.0% while creating and adding artificial snowy images to the database increase the F1-score to 94.4% - which mostly increases the true positive predicted value of the Snowy class.

As expected, adding an extra class of multiple patterns into the database decreases the F1-score of the test set by 6.1%. This happens since the samples of the *Multiple* class has the patterns of *Snowy*, *Icy*, *Wet* and *Slushy* classes which make the learning and optimizing CNN model's parameters difficult.

Finally, splitting each original image into two smaller images with the size of 188×184 pixels is tested. This helps reduce the size of the CNN model since the input image size is reduced and providing the opportunity of classifying two different regions in one image simultaneously. Since the number of parameters of the model has increased and also the smaller images may have less informative pixels, the weighted F1-score of the CNN model is decreased by 3.4%. Although this decrease implies that the performance is degraded, the capacity of the system is increased because it can classify more images at the same time.

Table 18: Prediction Results of Input Dataset of Multiple Input Network

Set	Scenarios	Performance (Average)			Performance (Weighted)		
		Precision	Recall	F1-Score	Precision	Recall	F1-Score
Training	Base	0.999	0.999	0.999	0.999	0.999	0.999
	Artificial	1.000	1.000	1.000	1.000	1.000	1.000
	Multiple	0.998	0.995	0.997	0.998	0.998	0.998
	Split	0.998	0.999	0.998	0.999	0.999	0.999
Test	Base	0.927	0.910	0.918	0.940	0.940	0.940
	Artificial	0.936	0.939	0.937	0.944	0.944	0.944
	Multiple	0.858	0.859	0.858	0.869	0.891	0.879
	Split	0.873	0.872	0.873	0.906	0.906	0.906

4.4 Conclusion

The winter road condition monitoring and classification, especially in cold-winter countries like Canada, is crucial for winter maintenance operations. In this research, a camera-based road monitoring methodology based on the fusion of the thermal and visible light video is developed and tested using Montreal winter data as an application environment. The GoPro, which is small and easy to install, water-resistant, robust to vibrations, affordable, and high-resolution, is chosen as the visible light imaging system. The ThermiCam Wide by FLIR is chosen as the thermal imaging system since it has a fast response to the thermal pattern changes, portable and easy to install, and can perceive the wide temperature range of road surface with high precision. The assembled and mounted system on an instrumented vehicle used to collect data in Montreal during winter.

The different CNN models were built and tested, including a single data stream model built by visible light or thermal image resources and a double data stream model built by both visible light and thermal image resources. Additionally, four dataset variations were built from the collected images and used for the sensitivity analysis, including i) the *base* dataset, ii) the *artificial* dataset expanded by adding artificially built snowy images, iii) the *multiple* dataset where a class with multiple road condition patterns in one image formed an extra category, and iv) the *split* dataset where each original image was split to two separate images.

The performance results over the test sets show that the F1-score of double data stream CNN model on the base dataset is better than the two single-stream CNN models with an accuracy of 0.866 for *Snowy*, 0.935 for *Icy*, 0.985 for *Wet* and 0.888 for *Slushy*. Besides, a

comparison between the single-stream models and the double stream model reveals that the classification of *Snowy*, *Wet* and *Slushy* images rely more on color information from visible light image, but *Icy* images are better detected and classified by thermal images since they have sharper temperature map. Moreover, the comparison of adjusting different weights for the input streams on the double data stream CNN setup shows that reducing weight of thermal stream makes the model perform a little better than the original model while reducing weight of the GoPro stream makes the model perform worse than the original model. This result shows that visible-light images input has more useful information than thermal images. Besides, we find the network whose all inputs were reduced has almost same performance as original model. That shows reducing the weights have limited influence on model performance.

In the comparison between original dataset and artificial dataset, we find the performance on *Snowy* significantly improved the performance; however, it caused the overfitting problem. Meanwhile, splitting original images into small sub-images had a negative effect on classification performance.

There are some limitations in this research. At first, the setup of the thermal camera used in this study is not quite suitable for the permanent setup since it needs a wired connection to a laptop. Besides, the resolution and field of view of the thermal camera could be increased to match with those of the visible light camera to have a better combination. The difference in perspective of the visible light and thermal cameras may lead to an unwanted error in image matching process. Few hours of winter data were collected for this research – longer periods of data collection in different street types could be collected in the future. Additional classes could also be added, such as black ice. The method of labeling the whole image restrict the potential of the dataset, we cannot tell how much snow or ice covered the road image.

As part of future work, more image data should be collected during day and night under different weather conditions. Moreover, instead of labeling the entire image with a single category, images could be labeled by type of cover, cover area and depth (using 3-D images). The performance of multiple sensor-based systems could be studied to identify not only cover types and areas but also the cover depth (e.g., snow depth). A long time period of data is also recommended to increase the training data set and add more categories. Detection and

classification algorithms could also be adapted for real-time applications as part of future work.

5 Concluding Remarks

Automated monitoring systems are essential for government and transportation agencies to reduce high road maintenance and snow removal costs by applying modern autonomous technology. Defective pavement conditions such as cracks or potholes deteriorate road user's comfort, damage vehicle, generate crashes and increase emissions. In addition to the pavement deterioration, northern cities suffer from adverse weather and then road surface conditions because of the long periods of snow and ice during winter months. In the past, the government and transportation agencies relied on costly and labor-intensive manual procedures to collect road information. To reduce associated costs, the use of automated monitoring solutions is crucial for the systematic pavement distress detection as well as the detection of hazardous road surface conditions during winter.

The objective of this research is to build a video-based CNN methodology for collecting road condition data and employ machine learning models to automate the classification of images to detect pavement deterioration or icy conditions. In the second part, the fusion of visible light and thermal images was investigated and tested for winter surface conditions. Then, a bi-image dataset for winter road conditions was collected and labeled. Afterward, different CNN models included a single data stream model built by visible light or thermal image resources and a double data stream model built by both. All CNN models were built and tested based on the collected data from the City of Montreal.

Among the main findings of this research, we can highlight the following:

For pavement distresses, the comparison between different input sizes and resizing factors showed that both input size and resizing factors have a significant effect on the performance of visual spectrum images. Thus, choosing the proper input size and resizing factor to process the image before inputting them into CNN is necessary. Then, the comparison results of different regularization schemes showed Batch Normalization could work as the Dropout method to anti over fitting. The performance is enhanced when the Batch Normalization layers are combined with low-rate Dropout layers.

For winter surface conditions, the results proved that the double stream input network had a much better performance by combining the temperature information from thermal

image and color information from RGB images. Meanwhile, the true positive results of single input stream networks indicated the classification between 'Snowy', 'Wet' and 'Slushy' rely on color information while the 'Icy' class relies more on the temperature information. In the comparison results of the double stream input network with different input weights, it showed the classification performance relies mostly on RGB input, but there is no significant difference between varying input weights. Afterward, when comparing the final results between the datasets with different processing strategies, this research part revealed that adding artificial images would improve the performance over the special class but would also lead to overfitting problems. Moreover, the results showed that when only reducing the size of the image in order to have more samples, it cannot improve the model performance because smaller images less information is available for input. Compared with the state-of-the-art research, the F1-Score of the CNN model in the first study reached 0.813 and 0.838 for over the 150×150 (A) dataset. For each class, the F1-Score of the model is 0.808 for "Pothole", 0.802 for "Patch", 0.860 for "Marking", 0.796 for "Crack-Linear" and 0.813 for "Crack-Network". However, by merging linear and network crack classes, the F1-Score over the merged class increased to 0.916.

In the second study, the F1-score and accuracy of the double stream input network over base dataset achieved 0.930 and 0.940 for precision and F1-Score, respectively. Meanwhile, the F1-score on the original dataset for each class is 0.866 for snowy, 0.935 for icy, 0.985 for wet, and 0.888 for slushy conditions.

There are a few limitations to this research that can be addressed as part of future research. Firstly, timing restricted the amount of image data that could be used for CNN training. From the dataset description, it showed the 'Pothole' class in the crack dataset and the 'Snowy' class in the winter road condition dataset had fewer images than other classes. Therefore, as future work we recommend the collection of more data during both day and nighttime, and for winter and non-winter seasons. Additionally, the data collected from different vehicles and camera positions could also be considered to study algorithm performance.

Secondly, alternative types of sensors should be explored. Although the visual spectrum (GoPro) can provide high-quality images, future work could explore 3D Lidar to capture the 3D information of the road surface together with the RGB images to increase the test

accuracy. The resolution and field of view for thermal cameras could be increased to match with those of the visible light camera in order to have a better combination and increase the covered road area. The difference in the perspective of the visible light and thermal cameras may lead to an unwanted error in the image matching process.

Lastly, the labeling method for winter road image dataset limited the potential of the dataset. In our winter road condition work, each image has only one label, which is based on the condition occupying the maximum area. If each image can be labeled by the various small polygons matched to different road condition areas, the dataset would have more small images with more precise labels by splitting the original image. The performance of new dataset generation strategies could improve the outcome and can be tested as part of future studies.

Appendix A: Image datasets

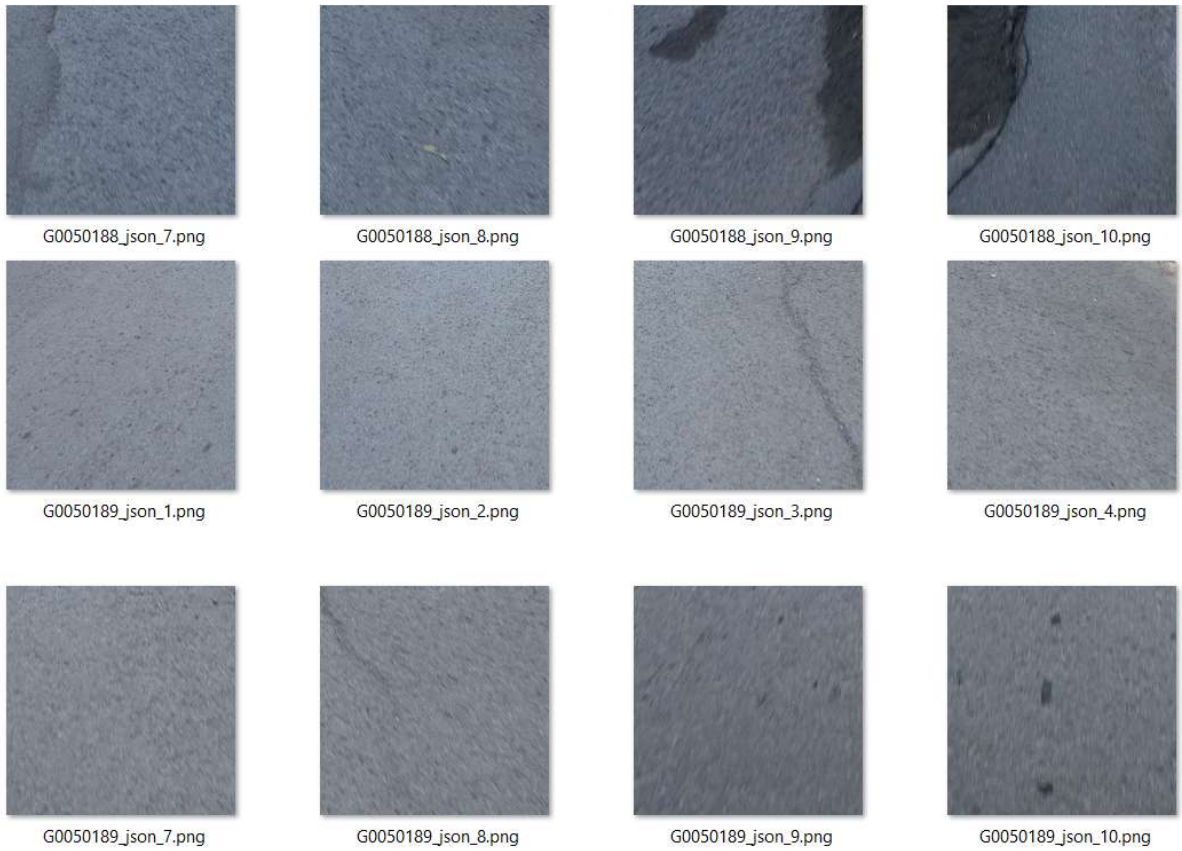


Figure 10. Samples of Montreal Pavement Dataset

Figure 11. Samples of Cropped GoPro Dataset

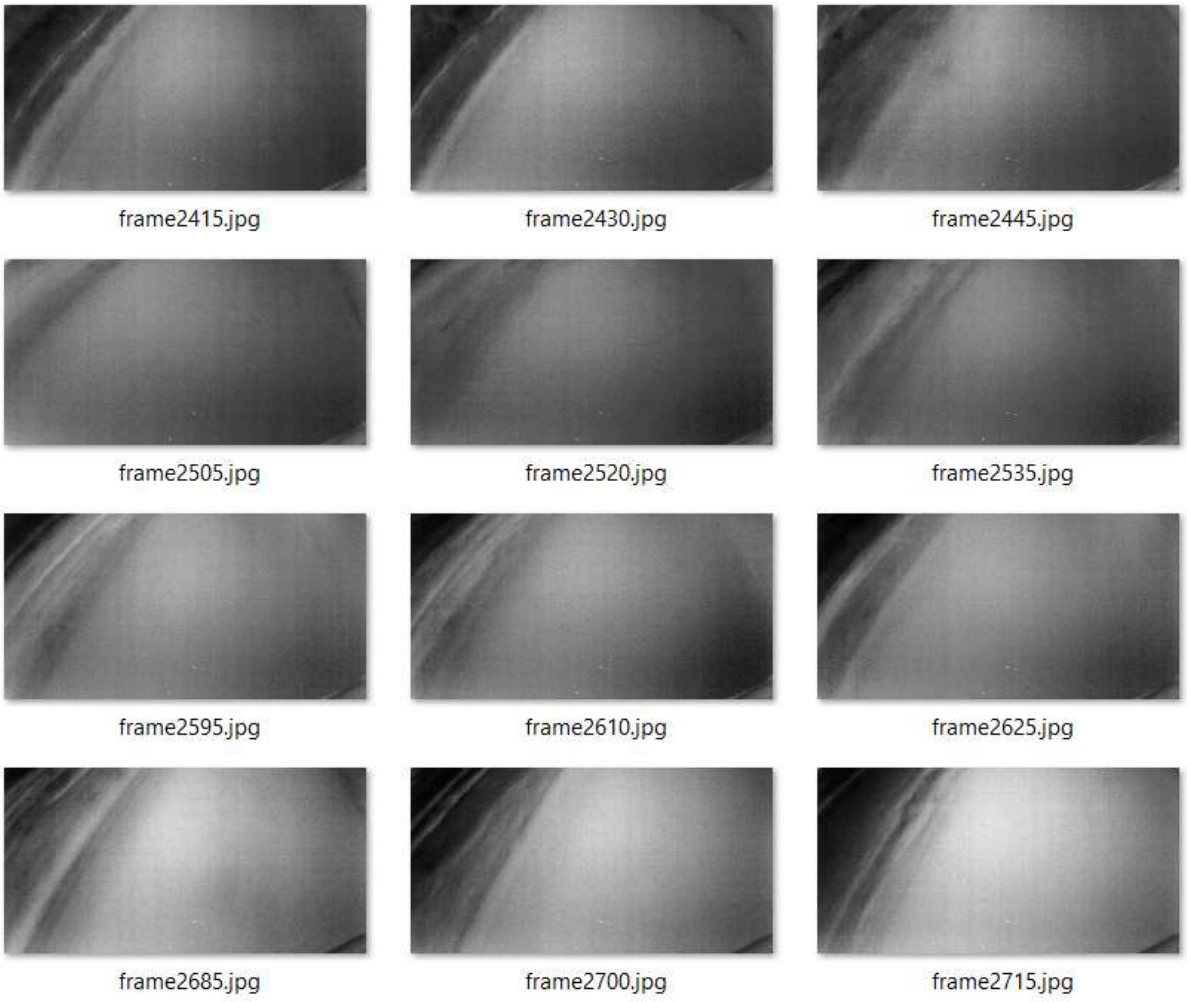


Figure 12. Samples of Cropped Thermal Camera Dataset

References

1. Lee, J., B. Nam, and M. Abdel-Aty, *Effects of pavement surface conditions on traffic crash severity.* Journal of Transportation Engineering, 2015. **141**(10): p. 04015020.
2. Ho, K.-Y., et al., *The effects of road surface and tyre deterioration on tyre/road noise emission.* Applied acoustics, 2013. **74**(7): p. 921-925.
3. Norrman, J., M. Eriksson, and S. Lindqvist, *Relationships between road slipperiness, traffic accident risk and winter road maintenance activity.* Climate Research, 2000. **15**(3): p. 185-193.
4. Qiu, L. and W.A. Nixon, *Effects of adverse weather on traffic crashes: systematic review and meta-analysis.* Transportation Research Record, 2008. **2055**(1): p. 139-146.
5. FHWA. 2020; Available from: https://ops.fhwa.dot.gov/weather/q1_roadimpact.htm.
6. Shahin, M.Y., M.I. Darter, and S.D. Kohn, *Development of a Pavement Condition Index for Roads and Streets*. 1978, CONSTRUCTION ENGINEERING RESEARCH LAB (ARMY) CHAMPAIGN ILL.
7. Eisenbach, M., et al. *How to get pavement distress detection ready for deep learning? A systematic approach*. in *2017 international joint conference on neural networks (IJCNN)*. 2017. IEEE.
8. Mahler, D.S., et al., *Pavement distress analysis using image processing techniques.* Computer-Aided Civil and Infrastructure Engineering, 1991. **6**(1): p. 1-14.
9. Jokela, M., M. Kutila, and L. Le. *Road condition monitoring system based on a stereo camera*. in *2009 IEEE 5th International conference on intelligent computer communication and processing*. 2009. IEEE.
10. Ouma, Y.O. and M. Hahn, *Wavelet-morphology based detection of incipient linear cracks in asphalt pavements from RGB camera imagery and classification using circular Radon transform.* Advanced Engineering Informatics, 2016. **30**(3): p. 481-499.
11. Tsai, Y.-C.J. and F. Li, *Critical assessment of detecting asphalt pavement cracks under different lighting and low intensity contrast conditions using emerging 3D laser technology.* Journal of Transportation Engineering, 2012. **138**(5): p. 649-656.

12. Lin, J. and Y. Liu. *Potholes detection based on SVM in the pavement distress image*. in *2010 Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science*. 2010. IEEE.
13. Marques, A. and P.L. Correia, *Automatic road pavement crack detection using SVM.* Lisbon, Portugal: Dissertation for the Master of Science Degree in Electrical and Computer Engineering at Instituto Superior Técnico, 2012.
14. Quinlan, J.R., *Induction of decision trees.* Machine learning, 1986. **1**(1): p. 81-106.
15. Bray, J., et al. *A neural network based technique for automatic classification of road cracks*. in *The 2006 IEEE International Joint Conference on Neural Network Proceedings*. 2006. IEEE.
16. Canada, G.o. 2019-12-04; Available from: https://climate.weather.gc.ca/historical_data/search_historic_data_e.html.
17. Montreal, T.C.o. *Rapport_financier_annuel_2019_fr*. [cited 2019; Available from: https://ville.montreal.qc.ca/pls/portal/docs/page/service_fin_fr/media/documents/rapport_financier_annuel_2019_fr.pdf.
18. Montreal, T.C.o. *Rapport_financier_annuel_2018_fr*. [cited 2018; Available from: https://ville.montreal.qc.ca/pls/portal/docs/page/service_fin_fr/media/documents/Rapport_financier_annuel_2018_fr.pdf.
19. Laurent, J., et al. *Using 3D laser profiling sensors for the automated measurement of road surface conditions*. in *7th RILEM international conference on cracking in pavements*. 2012. Springer.
20. Omer, R. and L. Fu. *An automatic image recognition system for winter road surface condition classification*. in *13th international IEEE conference on intelligent transportation systems*. 2010. IEEE.
21. Linton, M.A. and L. Fu, *Winter road surface condition monitoring: field evaluation of a smartphone-based system.* Transportation research record, 2015. **2482**(1): p. 46-56.
22. Zhang, C., et al. *Discrimination of highway snow condition with video monitor for safe driving environment*. in *2012 5th International Congress on Image and Signal Processing*. 2012. IEEE.
23. Pan, G., et al., *Winter road surface condition recognition using a pretrained deep convolutional network.* arXiv preprint arXiv:1812.06858, 2018.

24. Tabuchi, T., S. Yamagata, and T. Tamura. *Distinguishing the road conditions of dry, aquaplane, and frozen by using a three-color infrared camera*. in *Thermosense XXV*. 2003. International Society for Optics and Photonics.
25. Kuehnle, A. and W. Burghout, *Winter road condition recognition using video image classification.* Transportation research record, 1998. **1627**(1): p. 29-33.
26. McFall, K. and T. Niitula. *Results of audio-visual winter road condition sensor prototype*. in *Proceedings from the 11th standing international road weather congress*. 2002.
27. Jonsson, P. *Road condition discrimination using weather data and camera images*. in *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. 2011. IEEE.
28. Jonsson, P., J. Casselgren, and B. Thörnberg, *Road surface status classification using spectral analysis of NIR camera images.* IEEE Sensors Journal, 2014. **15**(3): p. 1641-1656.
29. Oliveira, H. and P.L. Correia. *CrackIT—An image processing toolbox for crack detection and characterization*. in *2014 IEEE international conference on image processing (ICIP)*. 2014. IEEE.
30. Ayenu-Prah, A. and N. Attoh-Okine, *Evaluating pavement cracks with bidimensional empirical mode decomposition.* EURASIP Journal on Advances in Signal Processing, 2008. **2008**: p. 1-7.
31. Zou, Q., et al., *CrackTree: Automatic crack detection from pavement images.* Pattern Recognition Letters, 2012. **33**(3): p. 227-238.
32. Medioni, G., M.-S. Lee, and C.-K. Tang, *A computational framework for segmentation and grouping*. 2000: Elsevier.
33. Ho, T.K. *Random decision forests*. in *Proceedings of 3rd international conference on document analysis and recognition*. 1995. IEEE.
34. Altman, N.S., *An introduction to kernel and nearest-neighbor nonparametric regression.* The American Statistician, 1992. **46**(3): p. 175-185.
35. Hassoun, M.H., *Fundamentals of artificial neural networks*. 1995: MIT press.
36. Krizhevsky, A., I. Sutskever, and G.E. Hinton. *Imagenet classification with deep convolutional neural networks*. in *Advances in neural information processing systems*. 2012.

37. LeCun, Y., et al., *Gradient-based learning applied to document recognition.* Proceedings of the IEEE, 1998. **86**(11): p. 2278-2324.
38. Simonyan, K. and A.J.a.p.a. Zisserman, *Very deep convolutional networks for large-scale image recognition.* 2014.
39. Szegedy, C., et al. *Going deeper with convolutions*. in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.
40. He, K., et al. *Deep residual learning for image recognition*. in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.
41. Zhang, L., et al. *Road crack detection using deep convolutional neural network*. in *2016 IEEE international conference on image processing (ICIP)*. 2016. IEEE.
42. Babkov, V.F., *Road conditions and traffic safety.* 1975.
43. Baladi, G.Y., E. Novak, and W.-H. Kuo, *Pavement condition index—Remaining service life*, in *Pavement management implementation*. 1991, ASTM International.
44. Huang, Y.H., *Pavement analysis and design.* 2004.
45. Siriborvornratanakul, T., *An automatic road distress visual inspection system using an onboard in-car camera.* Advances in Multimedia, 2018. **2018**.
46. ASTM, D., *Standard practice for roads and parking lots pavement condition index surveys.* 2011.
47. Jang, E., S. Gu, and B. Poole, *Categorical reparameterization with gumbel-softmax.* arXiv preprint arXiv:1611.01144, 2016.
48. Srivastava, N., et al., *Dropout: a simple way to prevent neural networks from overfitting.* 2014. **15**(1): p. 1929-1958.
49. Ioffe, S. and C.J.a.p.a. Szegedy, *Batch normalization: Accelerating deep network training by reducing internal covariate shift.* 2015.
50. Shimodaira, H.J.J.o.s.p. and inference, *Improving predictive inference under covariate shift by weighting the log-likelihood function.* 2000. **90**(2): p. 227-244.
51. Perez, L. and J.J.a.p.a. Wang, *The effectiveness of data augmentation in image classification using deep learning.* 2017.
52. Géron, A., *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. 2019: O'Reilly Media.

53. Zeiler, M.D., *ADADELTA: an adaptive learning rate method (2012).* arXiv preprint arXiv:1212.5701, 2012. **1**.
54. transportation, O.M.o., 2016.
55. Kietzig, A.-M., S.G. Hatzikiriakos, and P. Englezos, *Physics of ice friction.* Journal of Applied Physics, 2010. **107**(8): p. 4.
56. Bradski, G. and A. Kaehler, *Learning OpenCV: Computer vision with the OpenCV library*. 2008: " O'Reilly Media, Inc.".
57. Simonyan, K. and A. Zisserman, *Very deep convolutional networks for large-scale image recognition.* arXiv preprint arXiv:1409.1556, 2014.
58. Shimodaira, H., *Improving predictive inference under covariate shift by weighting the log-likelihood function.* Journal of statistical planning and inference, 2000. **90**(2): p. 227-244.
59. Xia, D.-F., S.-L. Xu, and F. Qi, *A proof of the arithmetic mean-geometric mean-harmonic mean inequalities.* RGMIA research report collection, 1999. **2**(1).