Automated Crack Classification for Underground Tunnel Infrastructure Using Deep Learning

Author: Darragh O' Brien

November 2021

Department of Civil and Environmental Engineering
National University of Ireland, Cork

This dissertation is submitted for the degree of

Master of Engineering Science (by Research)

Supervisor: Dr Zili Li

Head of School: Dr Jorge Oliveira

I dedicate this thesis to those who helped me achieve.

my parents, my brothers and my Girlfriend Sharon

DECLARATION

This is to certify that the work I am submitting is my own and has not been submitted for another degree, either at University College Cork or elsewhere. All external references and sources are clearly acknowledged and identified within the contents. I have read and understood the regulations of University College Cork concerning plagiarism and intellectual property. This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration, except where specifically indicated in the text. This dissertation contains less than 65,000 words including appendices, bibliography, footnotes, tables and equations and has less than 150 figures

Darragh O Brien

November 2021

ACKNOWLEDGEMENTS

I would like to appreciatively acknowledge funding of this research by the School of Engineering, University College Cork and the European Centre for Nuclear Research. I am eternally grateful to my supervisor, Dr Zili Li from the University College Cork for his patience, wisdom, endless enthusiasm and assistance throughout my studies

Equal thanks go to my supervisor John Andrew Osborne and my colleagues Roddy Cunningham and Eliseo Perez-Duenas, for their recurring support and vital guidance throughout my placement at CERN. Furthermore, I would like to thank the BE department at CERN for providing the image data used to perform the field test.

I would also like to thank the Department of Civil and Environmental Engineering, for providing access to The Google Collaborator Pro computing service enabling me to write and execute arbitrary python code through a browser along with providing access to Powerful GPU's and enormous quantities of Ram.

Finally, I would like to thank my parents, girlfriend and my friends, for always being there for me, for their encouragement, support and love.

ABSTRACT

One early sign of tunnel structure deterioration originates in the form of cracking, and therefore crack detection and resultant classification is integral for tunnel structural inspection and maintenance. Conventionally tunnel cracks are manually recorded and classified by trained professionals, which is costly, time-consuming and inevitably subjective. Recent advances in the deep learning space have allowed for automatic cracks detection algorithms to be developed and subsequently utilized in surface structural health assessment of surface buildings, bridges, roads and other civil infrastructure. Nevertheless, these methods of development underperform when implemented for a tunnel structure in an underground environment due to the disparity of illumination combined with the congested image data caused by pipes, steel mesh, wires, and other tunnel amenities. This thesis develops an intuitive crack directional classification approach that increases the accuracy, reduces time and subjectivity in comparison to traditional inspection methods. The detection of cracks by utilising CNN's is antiquity investigated by in literature however little of the writings develop the algorithm further for classification purposes.

The novel of this research is centred on the development of a crack classification algorithm that adheres to the directional classification rationale. The output information of the crack classification is correlated to the structural movement of the lining providing a deeper understanding of the tunnel behaviors.

To surmount these challenges, this thesis constructs a Convolution Neural Network (CNN) image-based crack detection method accompanied by an innovative crack classification for underground infrastructure environment. Conventional CNN's are developed from scratch, the proposed CNN incorporates transfer learning in the form of the VGG16 model with weights transferred from ImageNet. The transfer model was trained under various scenarios to determine

the optimal model for the operational task in the tunnel environment. The various models are trained using over 10’000 images validated on 2’500 images all of which are 256 x 256 pixels in size, these models are all subsequently tested using 30 images 3072 x 4096 pixels in size. The transfer learning model used outperforms that of the traditional CNN training method of training from scratch. The optimum transfer model accomplished testing metrics of 96.6%,87.3%,92.4%,89.3% for Accuracy, Precision, Recall and F1 score respectively. The proposed CNN appraises images regarding the existence and subsequent location of cracks. Detected cracks are subjected to the secondary classification CNN where the crack is categorized into one of the four crack classes which include the three directional classes of Horizontal, vertical and diagonal with the last crack classes incorporated to represent complex crack regions. The secondary classification CNN attains an Accuracy of 92.3% a Precision of 83.9% a Recall value of 82.3 % and an F1 score of 81.5%.

The performance of the manufactured integrated detection and classification method is analysed by performing a field test to evolve the research from a controlled theoretical setting into a realistic tunnel environment. The field test is performed on three separate tunnel sections with an amassed distance of 150 meters with the section testing the robustness, speed and ultimately prospect of application in the CERN inspection scenario. The outcome from this testing demonstrates that the established CNN crack detector/classifier can effectively overwhelm the unfavourable tunnel environment and accomplish results to a high standard.

Table of Contents

TABLE OF FIGURES

TABLE OF TABLES

TERMINOLOGY

List of Abbreviations

CERN- European Organization for Nuclear Research

CNN - Convolutional neural networks

VGG – Visual Geometry Group

SVM – Support vector machine

IPT – Image processing techniques

ML – Machine learning

DL – Deep learning

LS – Long shutdown

TP – True Positive

FP – False positive

TN – True Negative

FN – False Negative

LHC– Large hadron collider

CV – Computer vision

AI – Artificiel intelligence

CLC – Compact Linear Collider

FCC – Future Circular Collider

ASI – Artificial Superintelligence

KNN – K nearest neighbours

SVM – Support vector machines

DT – Decision trees

FC – Fully Connected

ILSVRC – ImageNet Large Scale Visual Recognition Challenge

GPU – Graphics processing unit

ROV- Remotely Operated Vehicle

CHAPTER 1

Introduction

In recent decades, the periodic growth concerning urbanization in the majority of countries across the globe has led to an increase in the necessity for underground infrastructure, for example, tunnels for public transportation and utility services. The characteristics associated with underground structures prove difficult to replace, with these structures likely to be forced to serve beyond their original design lifespan, even decades after construction, in comparison to their above-ground infrastructure counterparts. The tough environmental influences that these underground infrastructures are subjected to regarding corrosion, carbonation among numerous other detrimental deterioration factors inevitably cause the structures to worsen over time potentially posing a risk to the functionality in the long term. To ensure tunnel serviceability, intermittent tunnel inspections are required to be performed. Conventionally, the inspections are undertaken by trained professionals, examining the surface of miles of tunnels in search of defects such as water leakage, spalling, cracking along with monitoring any deformation. If a defect is found during an inspection it is subsequently logged concerning the severity, location and categorized with type defect. This logging of defects allows for ease of comparison to prior and future surveys of the tunnel. Though this conventional manual method is easily implemented it is a costly, time-consuming and inevitably subjective inspection technique.

Artificial Intelligence has been experiencing monumental progress in bridging the gap between the capabilities of humans and machines. Researchers and enthusiasts alike are tirelessly working on numerous facets of the Artificial Intelligence realm to make incredible things possible with the domain of Computer Vision being the perfect depiction of such progress. The purpose of Computer Vision is to empower machines to process visual data and view the world as humans do. The specific perception the machine gains on the world is distinct to the space in which it was constructed for operation with successful application in a multitude of scenarios from gaming to medicine and even agriculture. The advancements in Computer Vision with Deep Learning have been constructed and perfected with time the evolution is primarily attributed to the increased power and popularity of the Convolutional Neural Networks (CNN's). These ever-developing capabilities of the available technology in conjunction with the increased computation speeds has led to ideas which once were seen to be unattainably expensive to be developed are now being researched and executed.

In the last number of years, the civil engineering spectrum has embraced technological advances with standout developments being made in surface structural health monitoring. The computer-driven vision-based defect detection encompasses the vast majority of the innovation within surface structural health monitoring with image processing techniques and convolution neural networks the two most prominent approaches. Recent studies have accomplished astonishing results with (Cubero-Fernandez, et al., 2017) achieving an 80% accuracy crack detection on roads with the image data simply accumulated using cameras equipped on an ordinary car driving at a speed of 130km/h. The defect detection in bridges has also achieved extremely high accuracies as recently reported in (Chul Min Yeum, 2015), where a 96% accuracy of faults in bridge bolts was attained. Yet, in this surface structural health monitoring surge, the image-based technology for underground infrastructure is dwindling behind that of other civil structures equivalent above ground. One of the rationales behind this lack of development in comparison to other civil structural counterparts is due to the difficulties that an underground tunnel environment proposes: the lighting or lack thereof creates a particularly challenging environment consequently impacts the image acquisition as the dark environment makes it extremely difficult to collect quality image data. This lack of quality data has the knock-on effect of making it difficult to create a vision-based defect detection in the civil environment.

The European Organization for Nuclear Research (CERN) deep particle accelerator is a complex series of machines with progressively higher energies. Each machine along the series increases the energy to a set level before passing it on to the next step in the series and so on till it reaches the last accelerator. This last step is the large hadron collider, which was first used on the 10th of September 2008, has a circumference of 26.7km and varies in depth from 50 to 175m. The tunnel infrastructure that makes up the particle accelerator in CERN mainly has a horseshoe cross-section design which consists of four parts: one circular arc in the top roof, two side circular arcs of a radius of and a flat bottom and is concrete lined as seen in. While CERN is at the forefront of the development and research in the area of physics it tunnels structure is still subject to that of any other tunnel in the form of tunnel lining deterioration with time. The nature of the activities that take place within the tunnels in CERN creates temporarily extremely radio-activated environments not accessible by humans and means that structural maintenance checks can only be done during complete shutdowns such as from long shutdown 1(LS) and long shutdown 1 (LS2). The structural maintenance checks carried out still operate using the traditional visual inspection method which, as mentioned previously, is hugely time-consuming considering the vast extent of the tunnel infrastructure. This inspection method would become a more daunting prospect in the future with there being two supplementary accelerators presently under investigation to add further acceleration capability to the LHC. These accelerator studies are the Future Circular Collider (FCC) that would add 100km and the Compact Linear Collider (CLC) adding another 11-50km. This enormous additional infrastructure calls for an evaluation of the conventional examination of the tunnel infrastructure at CERN. This additional infrastructure demands a better and faster evaluation strategy to reduce the inspection cost and more importantly the exposure time of inspectors to hazardous environments

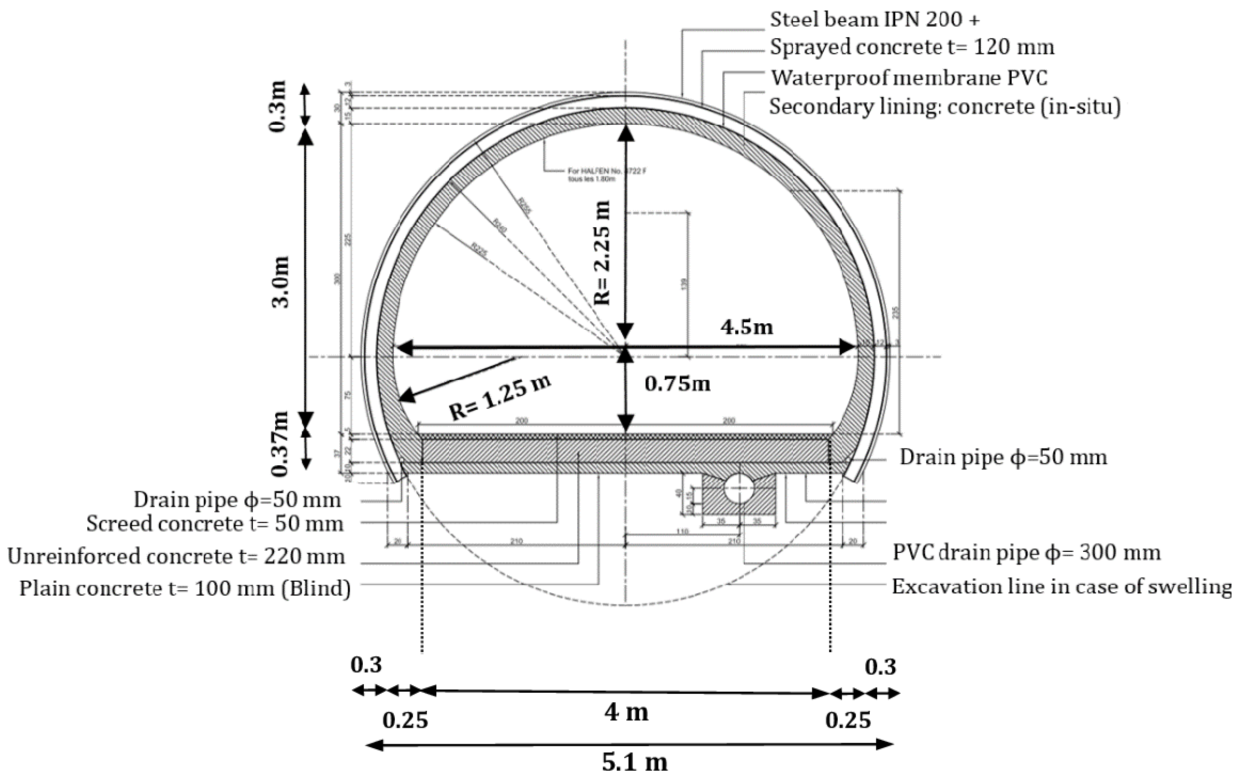


Figure 1- LHC Tunnel Geometry (Civil engineering department at CERN)

1.1 Aims and objectives

The inspections of underground tunnel infrastructure are conventionally undertaken by trained professionals employing visual examination which is considered a straightforward task for human cognition. Albeit a straightforward task for experienced inspectors, the lack of discerning characteristics of cracks produces challenges for deep learning methods to learn differentiating features to make accurate predictions. The objective of this research is to contribute to the field of tunnel inspection by providing a remotely operated comprehensive framework to monitor the CERN tunnel infrastructure. This thesis aims at producing an automated faster and more objective evaluation system than the currently deployed traditional visual inspection. A detailed case study on the current implemented and researched technology is conducted to improve design assumptions and procedural steps for a deep learning model to remove as many implementation problems as possible. This thesis pursues the following objectives:

1. Development of a binary crack detection convolution neural network algorithm that will be trained and validated on a dedicated database created from images of the CERN tunnel infrastructure. Furthermore, the additional adaption of the model for deployment in a twofold scanning window evaluation strategy will be used in the substantiation process by utilizing specific cracked tunnel lining images to test the accuracy and subsequently the robustness of the CNN model in the tunnel environment.

2. To construct a novel crack classification approach in conjunction with crack detection to provide a quantitative objective evaluation of tunnel lining data. The crack classification methodology will adhere to directional rationale with the output information of the crack directional formation being correlated to structural movement providing a deeper understanding of the tunnel behaviors.

3. To assess the applicability of the intergraded crack detection and subsequent directional classification method a field test on a continuous tunnel sectional data with an accumulated length of 150 meters will be performed. The appraisal of the developed system in a scenario such as this enables processing speed, accuracy, robustness and proficiency to simultaneously be analyzed. The developed algorithm will provide a global crack percentage in combination with subsequent crack class percentage with the refinement of these results will provide a sectional summary of the tunnel. The refinement of the data to sectional information is intended for use in

the discovery of underlying segmental structural behaviors in addition to highlighting problematic areas of the tunnel.

1.2 Organization of thesis

The thesis is comprised of six chapters, which are focused on the research development and application of a deep learning convolution neural network for the structural assessment of the CERN tunnel infrastructure using a novel unified detection and directional classification procedure.

Chapter 1 summarizes the background of conventional visual inspection and introduces the concepts of automation of said inspection with new technological constructs. The presenting of concerns and current pitfalls of automation in the tunnel environment in addition to stating the research objectives is discussed.

In Chapter 2 a broad literature review is performed for crack detection for both Deep learning (DL) and Machine Learning (ML) processes. As studies of crack detection specifically in tunnels using this technology are quite limited the literature review is expanded to incorporate other civil infrastructure utilizing this expertise for structural assessment.

Chapter 3 demonstrates the necessary steps taken in the creation of the substantial crack dataset. The training procedure and precautionary structural steps taken within the CNN structure to solidify the computer vision goal while reducing the possibility of overfitting are expressed and justified.

Chapter 4 exemplifies the optimization procedures via refinement methods in conjunction with the enhancement process on complexity and the volume features of the data. This data optimization is coupled with architecture/training experimentation to create multiple varying crack detection CNNs. To determine the optimal CNN model each model is examined under identical specific conditions to evaluate the best performing CNN for the crack detection task on CERN tunnels.

Chapter 5 establishes the rationalization of the alteration to the data design process necessary for the formation of a secondary innovative directional crack classification algorithm. The subsequent testing of the integrated detection/classification model is conducted and scrutinized under identical conditions to the singular detection model.

Chapter 6 performs a field test to transition the research, from the experimental setting into a realistic tunnel environment. The aforementioned field test is performed on three separate tunnel sections with an accumulated length of 150 meters to test the robustness, speed and ultimately prospect of application in the CERN inspection scenario.

Chapter 7 concludes the thesis by summarizing the significant elements concerning the performance and operation of the developed interrogated detection/classification novel algorithm constructed in this research. In addition to the summary, guidelines are outlined on what aspects have the greatest potential for enhancement to develop the classification algorithm before implementation.

CHAPTER 2

Literature review

Artificial intelligence (AI) has had an extraordinary evolution over the last decades growing from a simple ‘buzzword’ to a colossus within industry (Artificial Intelligence Market Size & Share Report, n.d.) refers to the global market value of AI being 39.9 billion dollars 2019 with a projection of 733.7 billion by the year 2027. This enormous growth within industry is reflected in the research world with AI becoming one of the most comprehensive and extensively researched areas of science expanding out from the traditional implementation within computer science out into philosophy, linguistics, psychology and many other areas. This widespread nature leads to AI having not one singular definition that is universally acknowledged by all experts.

“Some define AI loosely as a computerized system that exhibits behaviour that is commonly thought of as requiring intelligence. Others define AI as a system capable of rationally solving complex problems or taking appropriate actions to achieve its goals in whatever real-world circumstances it encounters.” (National Science and Technology Council, 2016)

Essentially all of the variety of definitions that are associated with the term artificial intelligence is centred on the same core ideology of a computer to preforming a task that resembles that of a human minds cognitive function such as learning, reasoning, problem solving and perception. This theory of AI is believed to have started in the 1940s when Alan Turning started contemplating “CAN MACHINES THINK ?” which eventually led to the creation of the “Turning test” a test of a machine’s ability to exhibit intelligent behaviour equivalent to, or indistinguishable from, that of a human (Turning, 1950). As with the nature of AI being intertwined with technology the genera of AI has drastically changed with the technological renaissance that is currently in progress. The increased ease and access to the technology that can quickly and efficiently crunch and store data as well as the increasing availability of data has drastically reduced the barriers in applying AI to a required task with current technologies like driving, aviation, medicine, online advertising, image recognition, and personal assistance all utilizing AI.

The excitement, intrigue and interest surrounding AI have become global in recent years due to advances made in robotics with the Sophia (robot) as well as the popularisation of driverless cars like Tesla and the Waymo Google car. However, as with everything it is not all positives with fears of AI replacing vast numbers of the workforce even Stephen Hawking warning that it “could spell the end of the human race.” These fears can be reduced maybe even elevated due to the research carried out at Stanford in 2018 (Grosz & Stone, 2018) where it was found that there was no “imminent threat” to humankind, as Hawking feared.

All these fears around the sector of AI stem from the depiction of the technology to the general society which usually formed from fear-mongering journalism and on a wider scale the science-fiction depiction of Artificial Superintelligence (ASI) where the AI does not just mimic the human behaviour but becomes self-aware and exceeds the capability of human intelligence. This narrative of “inevitability” and “devastation” that AI is and will cause is skewed as there is no reason to anticipate ASI having the ability to demonstrate human emotions and become malicious or dangerous. This rationale is due to the current success level that AI has achieved being virtually exclusive in narrow tasks and not on the comprehensive widespread application. This essentially means that the majority of currently developed AI can only operate the specific task it is designed for and are exceptionally good at their specific task often better than the human equivalent such as facial recognition, translations and even calculus. For example, a simple checkers computer system can beat a human playing checker, but the same program could not solve a complex maths question or translate a phrase. These General AI systems that can learn and apply their intelligence to solve any problem have yet to be archived. The astronomical challenge that creating such an AI system is not surprising when considering the complexity of the human brain to which this computer program must resemble. The breakthrough for General AI is highly reliant on the comprehensive understanding of the complete functionality of the brain which is currently not fulfilled. The inability to create a one model fits all scenario has not reduced interest nor appetite to apply the technology into diverse areas with the prominent sectors in which AI is utilised are computer vision, robotics, natural language processing, speech processioning, planning scheduling and optimisation. The prominence of AI in these sectors is not coincidental as each of the sectors embody isolated but primary responsibility of the human brain. For instance, robotics reflects the characteristics of movement, speech processing demonstrative of the hearing and computer vision (CV) enabling computers to observe and understand visual data.

In these prominent sectors, there is still enormous diversity concerning the training methods, the data needed as well as output and the implementation possibilities this diversity have a secondary effect on the continuous growth of the AI umbrella. This evolution has caused the term AI to form an all-encompassing title over the diverse tools, techniques, and algorithms which embody computational intelligence. These diverse tools, techniques, and algorithms can be split into two very distant categories of AI of ML and DL. The relationship between these categories and the title of AI is quite a starlight forward one in essence as AI is the overarching title that encapsulates ML. With the area of DL being explained as a subset of the ML spectrum. The connection between the prominent sectors of AI and the subsequent subset's of AI is illustrated in Figure 2.

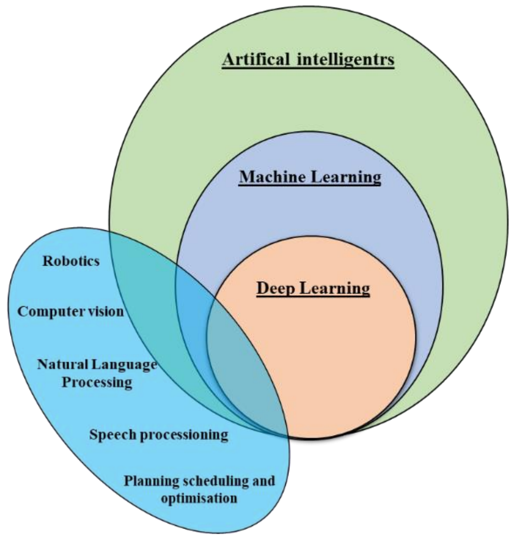


Figure 2 - Venn diagram of the sub-fields of Ai and the genera's of application.(Created by Author)

2.1 Machine learning

As machine learning is a subset of AI the associated literature on the subject is quite substantial causing there to be several definitions used to describe the main principles of ML similar to AI. The first and most commonly used definition was introduced in 1959 by (Samuel, 1959) who defined machine learning as "the field of study that gives computers the ability to learn without being explicitly programmed'. Although these algorithms can be effective in a vast array of fields, they all observe the same general structure as shown in Figure 3. This generic structure is divided into 3 discrete parts of data preparation, training and testing.

The first and most important part of the development of a ML algorithm is the data preparation. This typically entails the collection of quantity but more significantly quality relevant data. As conventional ML techniques operate with a predetermined number of input features the collected data undergoes pre-processing to allow for feature selection. This feature extraction process allows for the user to extract features from the data which they perceive to be relevant to the execution task. The feature extraction allows the model to train faster and more efficiently as the complexity is reduced by way of reducing the number of relevant features. The collected features are subsequently split into subcategories to be used in differing ways in the model development. These categories are training, validation and testing with the common ratio split being 80:10:10 respectively.

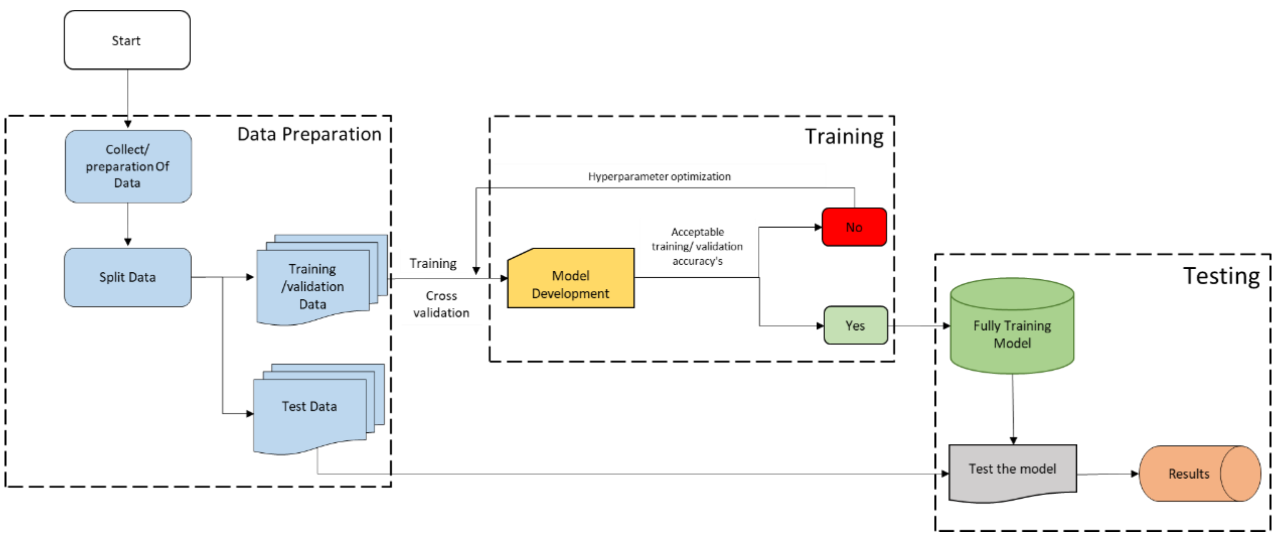


Figure 3- machine learning structure flow diagram(Created by Author)

The data collected is refined and subsequently split into their corresponding categories, the development of the ML algorithm moves on to its next stage the model creation and subsequent training. The creation of a ML model is the manifestation of a hypothesis where the model is fundamentally a data structure that stores a representation of a dataset in the form of weights and biases. The training of such models is accomplished by an iterative process of inputting the training data to the model to facilitate the learning of the relevant features that have been extracted and deemed relevant. Depending on the volume of features utilized in the solution due to the simplicity of the workflow structure of the ML model it is possible to trace the information throughout the entirety of the model training. This enables the user to intervene at mid-phases of the structure to facilitate model optimization. Inevitably as the number of feature increase so does the complexity of the problem and therefore it becomes harder to trace the relevant information and follow the connections.

Once the model is judged to be completely optimized on the training data it is considered to be fully training and can be subjected to testing. This is traditionally carried out to truly test the robustness and accuracy of the model as it is being exposed to new data that it previously has never seen and forced to make a trained judgement. Since ML algorithms are created optimized and trained using the features that the user defined as relevant the algorithm test results should reflect exactly how the user approaches the problem. This in theory is good as the design of the model is representative of the expertise in this section it is utilized in, however, in some scenarios, this proves to be problematic as the algorithm is limited to executing the explicit task it is designed for and not gaining a deeper understanding of the task.

2.1.1 ***Crack detection using machine learning techniques***

Civil engineering inspection is an area in which maximisation of precision and efficiency are of utmost importance due to the repercussions that are associated with an error. In this regard, machine learning algorithms are becoming increasingly popular and more widely used to carry out inspections. During visual inspections, the inspector is examining the structure under criteria but is predominantly looking for degradation of the structure. This degradation is all dependant on the material, environment and structure being examined. Cracks are the most important feature to be located during these inspections as they are the first and most significant sign of structural deteriorating. Examinations of civil structures employing machine learning can be conducted with numerous data types such as infrared thermography data enabling the thermal effectiveness of a building can be quantified, acoustic-based data allowing to test a pipeline or a pressure vessel for a possible cracking/leakage. However, the majority of ML crack detection algorithms in a civil structural inspection are created using Photographic data (i.e., images and video) due to ease of acquisition and supporting the direct comparison to the current method of visual inspection. As crack detection commonly uses visual data for the training and subsequent implementation is considered to be within the scope of the computer vision (CV) sector. Computer vision is "the enterprise of automating and integrating a wide range of processes and representations to gain a higher understanding of video or image data" (Ballard & Brown, 1982). A CV algorithm is designed to resemble that of human vision the distinguishing factor between the two is the cognitive ability of the human allowing for a head start over the computer to gain a higher understanding of video or image data. This advantage is formed over a lifetime of experiences that allow the human brain to gain context relating to distance an object is away, the overall size of an object, whether the object is moving and even if the is something noticeably wrong occurring.

While this experience is the separating attribute between human and computer vision it can be overcome with adequate quality data and sufficient training. In the context of crack detection particularly in underground infrastructure, the detection proves to be a challenging task for any CV and ML method. It is problematic for ML algorithms to be established for this crack detection task because these frameworks are created with the purpose of learning characteristic feature that defines a class of objects which in the case of cracks are quite limited with the primary features being the irregular and jagged nature of the crack combined with high contrast to the surrounding area. These defining characterises of cracks are not exclusive and are easily confused with typical objects that occur in the civil structural environment such as construction joints. During a visual inspection performed by a trained professional, the structural integrity is intuitively deciphered based upon training and experience. Whether unbeknown to the inspector or not they are carrying out three subconscious decisions of detection, calcification and interpretation when examining the structure. Each of these decisions has increasing difficulty in implementation for ML with the majority of the current literature in the fields of detection and classification. The programming of a ML algorithm to interoperate the data and make a nuanced decision on the structural integrity is exceedingly difficult due to the need for reference to location sounding information etc. which cannot be provided strictly by image data alone. This leaves the reaming two inspection decisions to make up the bulk of the literature with there being plentiful diverse approaches used in achieving the same direct goal of crack detection. This research focuses on investigating the most commonly implemented ML approaches of k nearest neighbours (k-NN), support vector machines (SVM) and Decision trees. The aim of this is to provide an expansive perspective of the areas of operation while focusing on the three most prominent methods.

2.1.1.a K nearest neighbours (K-NN)

A k-NN framework is a simplistic approach to machine learning as the framework is trained with only inputting labelled data. This method of k-NN classification utilizes the proximity of training points to make a prediction classification. The classification of a point is reliant on the arbitrarily K number which is a user-determined number. This arbitrary K number sets the number of data points to be considered to make a classification decision with the majority of the data points determining the class of the unlabelled test data. This is illustrated in Figure 4 where the unlabelled data point is illustrated by the red star. The variation in the K values in the Figure alters the Class of the unlabelled data point as when K is set at 3 the class of the data point is determined as class b but when the K value is increased to 6 the class is also changed. Owing to the simplicity of training a model like this for an arbitrary task such as crack detection/classification in literature the method of K nearest neighbours is commonly used for benchmarking and comparative analysis of the performance rather than exploited as the primary approach.

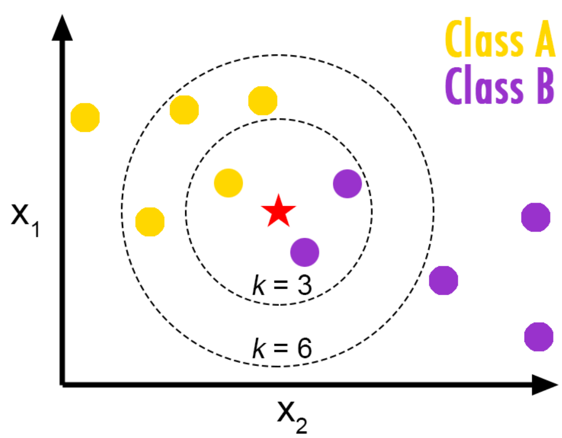


Figure 4 - k-NN Classification (Anon., 2021).

The research of (Zhang, et al., 2014) applies 4 separate models to classify cracks in a subway tunnel environment two of which are Machine learning models in the form of k-NN and a support vector machine with the reaming two being deep learning networks using the radial basis function and extreme learning machine techniques. These models were trained using data collected via high-speed complementary metal-oxide-semiconductor (CMOS) industrial cameras which collected images of the subway tunnel lining. This image data is segmented using morphological image processing techniques of smoothing, Black Top hat Transform and lastly thresholding. After these manipulations on the image are complete the feature extraction process was carried out by using a distance histogram technique that effectively deciphers the spatial difference between the irritant object and cracks. The four methods used in this study perform externally well when tested on images from the Beijing Subway Line 1 with the extreme learning machine achieving the highest test accuracy of 91.6% only slightly above that of the support vector machine result of 91.3% trailed by the radial basis function neural network attaining a 90.1% accuracy and lastly k-NN classifier scoring an 88.7% accuracy for this experiment.

(Sinha & Fieguth, 2006) uses a neuro-fuzzy network technique to ultimately detect defects in a buried pipeline the method is essentially an adaptation on the conventional neural network. Similarly, to (Zhang, et al., 2014) this research uses the k-NN as a level to measure against the primary method. Before the feature extraction can occur, the pipeline data is segmented using a combination of mathematical morphological to reduce the ‘fuzzy, nature of the data. This enables the feature extraction practice to be created around a statistical analysis of the data using five determinant features of the area of the regions, the number of objects, the major and minor axis length and lastly Mean and variance of image projections. In this comparative analysis for crack classification accuracy the k-NN sores a testing accuracy of 81% with the optimum model of the research achieving a 94.1% accuracy.

(Mokhtari, et al., 2016) strictly conducted a comparative analysis with no new ml approach used the study concentrates on the methods of Artificial neural networks (ANN), Decision trees (DT), k-NN, and Adaptive Neuro-Fuzzy Inference System (ANFIS). For this study, 26 pavement images, including 11 images in good pavement condition, seven images in an intermediate condition, and eight images in poor condition which underwent processing and feature extraction resulting in 3090 crack image components that are used for the training of the various models. Once trained the models were tested where this study deemed the k-NN and Dt methods to be unactable/ unsuitable for implementation for pavement crack detection with the remaining two methods considered acceptable with the ANFIS chosen as the preferred option.

In comparison with the other examples (Younes, et al., 2009) uses a k-NN as the singular ML algorithm of the research with the model constructed to detect deterioration in asphalt pavement. The database of images is attained with a simple digital camera to reduce the possibility of noise in the images. These images undergo 4 simple processing steps to prepare the data for training. The aforementioned steps are scaling the images to the desired size, conversion from RGB images to grayscale images, subjecting the images to median filter and finally, k means clustering operation. Experimentation is carried out on the user-defined k number with the results hugely varying with the optimum model with a K value of 8 achieving a classification accuracy of 80%.

2.1.1.b Support vector machines (SVM)

Support vector machines (SVMs) were initially proposed in 1963 by (Vapnik, 1963). A support vector machine is a supervised learning algorithm that separates/classifies data using a linear classification structure as shown in Figure 5. This classification structure is based on the margin maximization principle which improves the complexity of the classifier intending to achieve excellent generalization performance. The SVM achieves the classification task through training on labelled data to create an optimal Hyperplane to separate the categories of data along with constructing the maximised margin. An alternative to the conventional linear SVM is the Kernel SVM developed in (Boser, et al., 1992) which operates in the same manner as linear SVM, but the data is processed with a kernel function thus allowing for the execution of nonlinear classification problems. Both methods of SVM operate in feature space and therefore, require the use of a feature extraction process when dealing with image data. Since the system of SVM functions strictly in feature space combined with the reality of cracks having very few distinctive characteristics, the means of implementation of SVM for crack detection are quite similar with any variation of application occurring in the approaches of data collection and feature extraction.

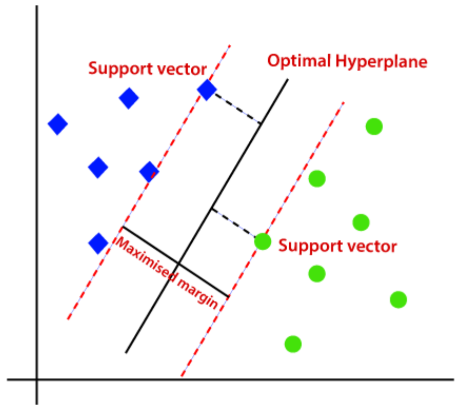


Figure 5- (Here, 2021) support vector machine depiction.

(Sari, et al., 2019) creates a SVM to detect cracks in the road, collecting the image data from the road infrastructure in Banjarmasin, Indonesia using a simple setup with a smartphone. Before feature extraction, the images are segmented using the OTSU thresholding method where a global threshold of the image is calculated along with the standard deviation of the filter. The final threshold value is the sum of the OTSU threshold values and half of the standard deviation this enables the enhancement of the crack features making them more distinguishable and simpler to extract. Gray Level Co-occurrence Matrices is the feature extraction technique utilized which works on the principle of texture calculations in the second order. Measurement of textures in the first order applies statistical calculations which are based on the pixel value of the original image. In the second order, the relationship between the two-pixel pairs of the original image is considered. GLCM employs five quantities, in the form of the angular second moment

(ASM), contrast, inverse different moment (IDM), entropy, and correlation. This method is highly effective and achieves impressive results when tested with a testing accuracy of 96.25%.

(Hadjidemetriou, et al., 2018) conducts a comprehensive evaluation on the use of patch detection of road damage while utilizing low-cost technology. This is carried out by experimentation with the data collection type to the patch size analysis. The data is collected via 2 different methods the first smartphone attached to the dash of the car, the second via a go pro attached to the rear with the last using the same go pro setup but the variation being the data was collected at night. This video data was segmented into non-overlapping image data which in turn is converted into grayscale images as this reduces the computational time and complexity without losing any critical information since pavement colours are mainly shades of grey. These images are subsequently divided up into smaller image patches processed and manually classified to create ground truth data that can be utilized to train the SVM. Through experimentation on image data collection type the size of the image patches from 4x4 to 40x40 the optimal SVM model is established where the block size of 20 x 20 reaching a block-based detection accuracy of 87.3% where the images were collected via the go pro during daytime. This proves to be an innovative and accurate system for automated detection and quantification of patches in pavement frames, utilizing the fast classification method of SVM.

(Chaiyasarn, et al., 2018) contrasting to the other examples concentrates on masonry structural cracks. The data collection was carried out by flying a drone in the vicinity of these masonry structures to collect video which is subsequently converted into images patches of size 96 x 96 pixels. This resulted in a formation of a database of 6’002 image patches which were manually separated into training/validation and testing sets with the training/validation comprising of 3162 image patches and the remaining 2840 images used for testing. Also contrasting with the other example, the SVM used in this study was used in conjunction with a deep learning CNN model. These combined techniques known as CNN-SVM is implemented to automatically classify image patches and to localize crack regions on masonry images for inspection. The SVM is used in replacement of the conventional SoftMax classification layer of the CNN. The main objective in SVM is to find a hyperplane that separates the largest fraction of a labelled dataset for binary classification. This CNN-SVM is directly compared with a CNN with a conventional last classification layer with the CNN-SVM outperforming the CNN in every testing metric achieving results of 74.9% 82%,78%,78 % in the testing metrics of Accuracy Precision Recall F1 score.

2.1.1.c Decision trees

Decision trees (DT) are the most simplistic and easily understandable machine learning technique with this classification type following a tree-like structure making it conducive to comprehending the decision-making process to specialists and non-experts alike. These decision trees are trained in a supervised manner with a predefined target variable with there being two categories that decision trees are used in the first and predominate type is classification trees used to predict a class with the second type being Regression trees used when an outcome is a real number and not a class. The classification of a new input resembles an object moving through a flow chart with the process starting with the top node (the root) with the value of this top node being ‘outlook’ in Figure 6. The decision tree illustrated in Figure 6 determines are the conditions suitable for playing golf. The input example is then moved down the branch that corresponds to a particular value of this attribute, arriving at a new node with a new attribute. This process is repeated until one arrives at a terminal node—a so-called leaf —which is not labelled with an attribute but with a value of the target attribute. The ultimate goal is to create a predictive model that can make observations about a sample (the branches) and make accurate conclusions about the sample's target value (the leaves).

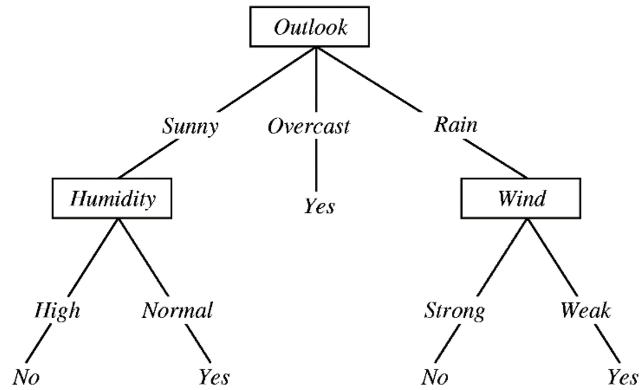


Figure 6- (Quinlan, 1986) well know decision tree example of are the conditions suitable for playing golf.

(Cubero-Fernandez, et al., 2017) uses decision trees to detect and successively classify crack images into three categories based on the directional profile of the perceived crack. This study resembling that of (Hadjidemetriou, et al., 2018) operates the data collection using a camera attached to the back of a vehicle where the distance between the camera and the pavement was 1.2 meters with a field of view of 45 degrees resulting in the collection of images of size 320 x 320 pixels representative of 99×99 cm area of pavement. The resulting database was made up of 1,000 images 600 used for the training and cross-validation of the model and the residual 400 comprising of 100 images from each class used for testing purposes. The images are undergoing several image processing stages of Bilateral filtering followed by edge detection in the form of the Canny operator and finally exposed to Morphological filtering to enhance the crack feature which is extracted using a projective integral technique. This projective integral technique separates the data concerning the directional information in the rows/columns thus providing information for 2 classifiers to be created corresponding to the vertical and horizontal data. Both models are applied on unclassified images, resulting in the classification of images as no crack (no model detects a crack), horizontal or vertical cracks (when only one model detects a crack), and alligator crack (when both models detect cracks) with overall testing accuracy being 80.25%.

(G. Decor, et al., 2019) conducts a comparative analysis between three statistical learning algorithms a random forest decision tree method and two CNN models. Each of these models is trained on tunnel lining images data roughly split 50/50 between concrete and masonry. The random forest algorithm (L. Breiman, 2001) is a meta-classifier, consisting of a set of decision trees. In this ensemble learning approach, each tree classifies a sample, represented by its feature vector, into a category. The final classification of the sample by the forest is obtained from the majority vote of all trees. In this application, the number of trees in the forest has been empirically set at 200 and the minimum number of elements in each leaf of the tree is set at 1. The selected features should best represent the characteristics observed on the samples of each class. In the case of tunnel images the three features of texture, orientation and binary relationships between a pixel and its 8 neighbours are used to classify an image. This method proves to be highly effective achieving an accuracy of 91.73% for Concrete tunnel images and 92.81% for Masonry. Even though it is an effective method of detection it is still below that of the alternatives posed in this research with both the CNNs accomplishing higher results in testing on all metrics.

2.2 Deep Learning

As aforementioned Deep learning (DL) is a subsection of Machine Learning (ML) causing there to be very few distinguishing factors that separate the two methods of Ai. The distinctive discrepancy is the omission of the prepossessing phase where the relevant feature is determined from the DL framework as depicted in Figure 7. This exclusion means that the focus of the user is altered from extracting features and is now aimed at the selection of data that represents the task. As shown in Figure 7 the feature extraction in the context of DL is embedded within the framework allowing for optimization through the process. Another basic difference comes in the ability to track information as it passes through the framework. In traditional ML the models use of selected features represent drastically less data than the Raw data that is needed and used for Deep learning. The increase of the volume of data that is utilized means the complexity of the data connections made is more often than not inexplicable challenging to follow leading to DL models to be referred to as black-box algorithms. This inability to follow the data step by step through the model reduces the possibility of intervention within the process the only possibility for such involvement is situated at the end of each iteration. Therefore, the output of the algorithm is conditional on the data that it is exposed to as the algorithm concentrates on finding patterns, correlations and features within the data to create a representative function for the input data.

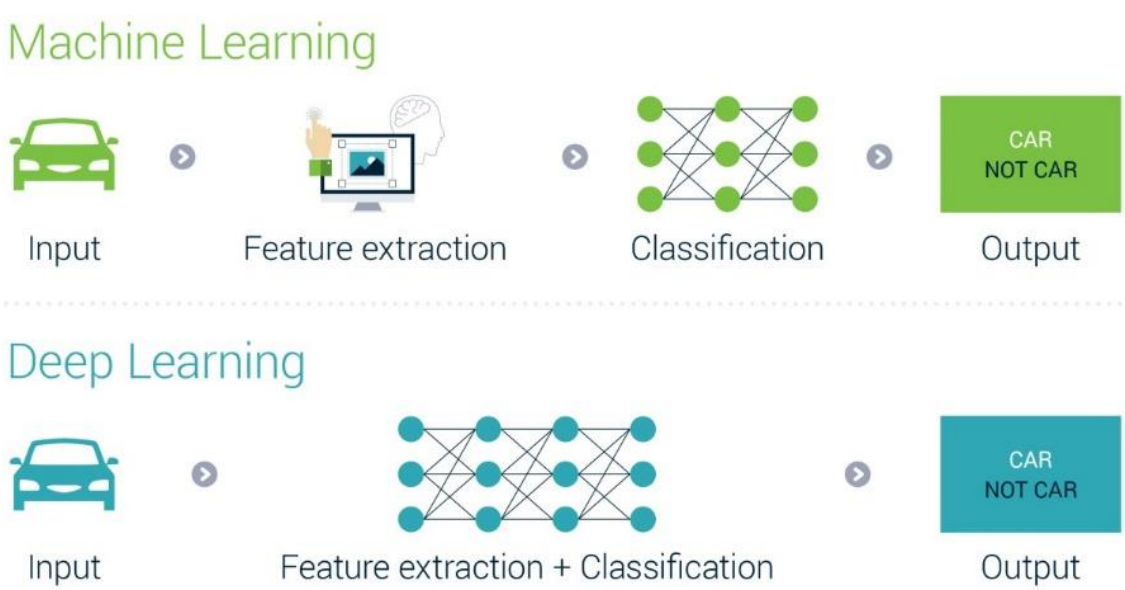


Figure 7- Contrast between the Machine and Deep learning training methods. (Bapat, 2021)

"More data beats clever algorithms, but better data beats more data" Peter Norvig

This quote illustrates the importance of the data from a DL perspective as the input data directly influences the output and ultimate evaluation of the model. There is no definitive answer to say which data is appropriate for the completion of a task as each dataset will result in a corresponding distinctive solution. This further emphasises the importance of data selection for DL with it continuing to be the most notable challenge. Despite this crucial role that data design has on the success of a DL approach it is not considered an explicate process within the framework of a DL algorithm. Therefore, a holistic understanding of the operational process within a DL framework is essential in the creation of a successful DL algorithm. This Deep Learning (DL) literacy is integral in the determination of a framework compatible with the available data as distinct frameworks are more receptive to various needs. For instance, recurrent neural networks are best suited for comprising time data, whereas Convolution neural networks (CNN) are more appropriate for tasks requiring conservation of spatial relations typically. Knowing the suitability of CNN's in the spatial relations tasks i.e. visual data the decision was made to concentrate the remainder of this section on gaining a holistic understanding of the concept's that formulate a CNN along with the various permutations of CNN's used civil engineering specifically for crack detection.

*2.2.1 **Convolutional neural networks (CNN) Architecture***

Neural networks are simply described as a series of algorithms that attempts to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates. Convolutional neural networks are essentially a sub-category of Neural networks with the inherent difference being that a CNN uses the mathematical procedure of convolution at least once in their framework which has made them extremely proficient when handling Visual data.

The catalyst for the current CNN revolution was the ILSVRC computer vision competition in 2012 where a CNN named 'Alex net' definitively won the competition beating the runner up by over 10%. This quintessentially was the breakthrough point for CNN's showing their efficiency in computer vision over the other computer classification alternatives.

The architecture of a CNN model can be vaguely split into two blocks of feature learning and classification as shown in Figure 8. The first of these block performs template matching by applying convolution filtering operations. The first layer filters the image with several convolution kernels and returns "feature maps", which are then normalized via an activation function and/or resized. This process is repeated several times dependant on the number of layers with the final of the layers in the block condensing the final feature maps into a single vector with this vector defining the output from this initial block and forming the input into the subsequent classification block. The condensing of the features into this vector allows the second block to function as a traditional neural network. The input vector values are transformed with several linear combinations and activation functions to return a new vector to the output. This last vector in the block contains as many elements as there are classes for the specific problem with each element having a probabilistic value that the input image belongs to each class. Therefore each element has a value between 0 and 1 with a cumulative sum of all the elements equalling 1. Contingent on the design specification needs the type of activation function varies as binary classification uses a logistic function but in the case of a multi-class classification model the SoftMax activation function is predominantly used. The architecture of a CNN can be summarized as a stack of layers executing differing computational operations on the data input to make a prediction. The typical CNN layered structure is constructed using 4 major types of layers as the building blocks these layers are convolution layers, activation layers, pooling layers and fully connected layers.

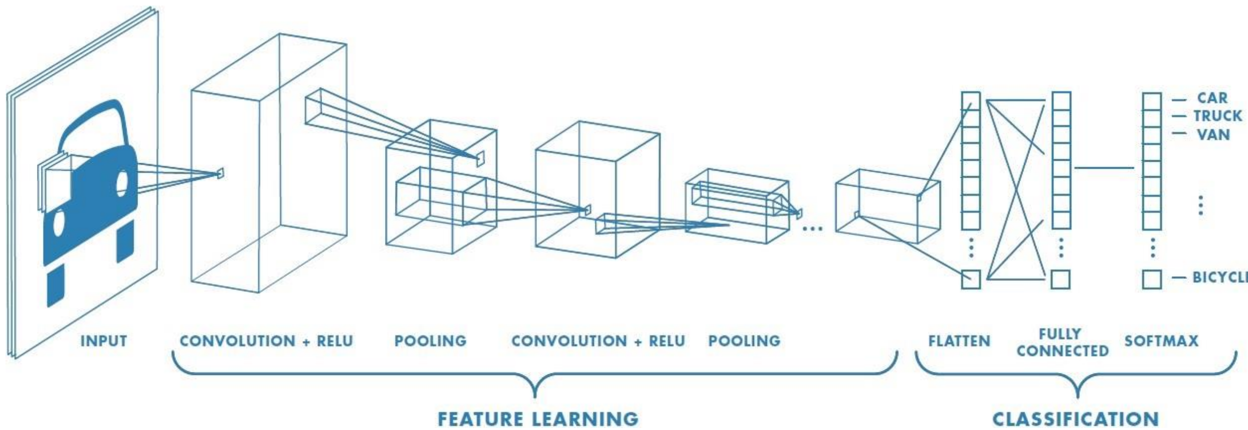


Figure 8 - Illustration of a Vehicle detection CNN architecture.

2.2.1.a Convolution layers

The very first layer and in essence the foundation of a CNN is the convolution layer. In the area of computer vision images are represented as numerical information that corresponds to column and row values where colour and or brightness data is stored. The numerical information of an image is defined by the colour mode as colour images (RGB images) are stores more information than a single channel greyscale image. A convolution operation is a mathematical operation on two functions to produce a third function that expresses how the shape of one is modified by the other. When this operation is utilized in a CNN the two products used are arrays in the form of the input Image intensity array and the convolution operating array also known as a kernel or filter where the dot product of these two arrays produces an output known as a feature map. Figure 9 illustrates this numerical convolution process where the stride of the kernel is set at 1 with no bias attached. By design, these kernels/ filter matrices are smaller in dimension than the input image array to facilitate the iterative process of convolving over the image to produce the feature maps. In the CNN application there are multiple convolution operations used in a single layer to obtain distinctive features from the input images. These filters are normally initialized at the beginning of the CNN construction and then via the learning process are adapted and optimized to extract the desired features. The layered structural nature of CNN's allows there to be multiple convolution layers used throughout the model.

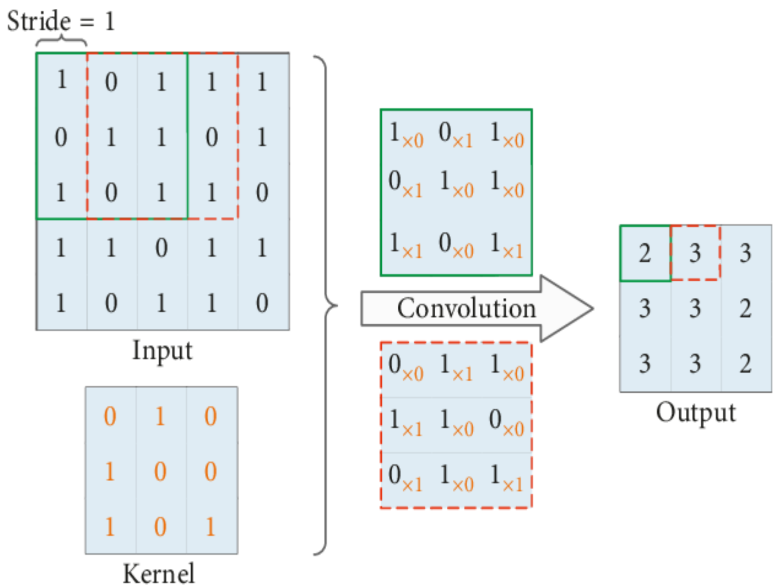


Figure 9- convolution operation

As the data is propagated through the network of multiple convolution layers the extracted feature map from a previous convolution layer is fed into the succeeding convolution operator. This results in the initial convolution layers to detect rudimentary information from the image such as edges, colours blobs and primitive shapes. However, this propagation of data allows for the later convolution layers to concentrate on more abstract features and spatial relationships that are specific to the task. This phenomenon is illustrated in Figure 10 where the feature maps from various layers at differing stages of a crack detection model are shown.

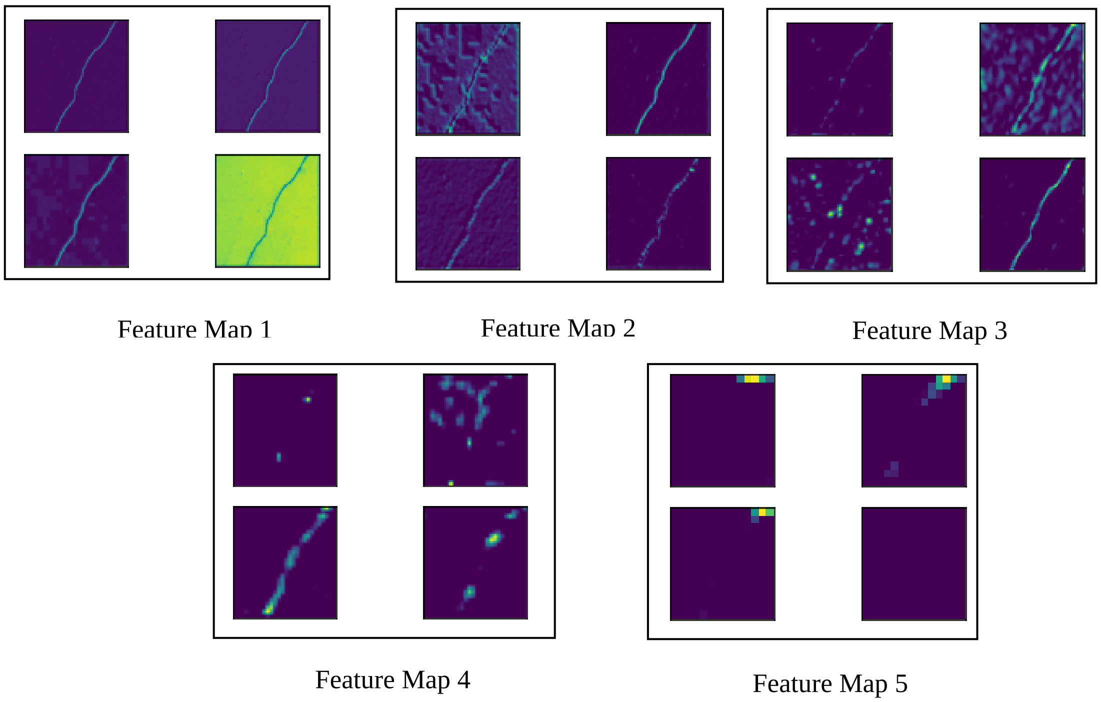


Figure 10 - Feature maps from varying layers of a crack detection model

2.2.1.b Pooling layers

The objective of pooling layers is the sub-sampling of the feature maps. These feature maps generated by the convolutional operations all include some information on the input neurons thus causing information redundancy and inevitably increasing the computational cost. This subsampling of the feature maps essentially shrinks large-size feature maps to create smaller feature maps referred to in the context of CNN's as downsampling. The downsampling of the data reduces the spatial size however reserves the valuable information (or features) to minimize the reduction in possible overfitting while optimizing computational speed. In a similar manner to the convolutional operation, both the stride and the kernel are initially size-assigned before the pooling operation is executed. In addition to the size assigned the type of pooling is also predetermined with numerous options available which include tree pooling, gated pooling, mean pooling, min pooling, max pooling, global average pooling (GAP), and global max pooling. The most familiar and frequently utilized pooling methods are mean pooling and max pooling with the latter considered to be the optimal pooling type for image data processing per (D. Scherer, 2010). These two prominent methods of pooling are illustrated in Figure 11 below where a 2 x 2 kernel is used with a set stride of 2 where the colour of the output section is representative of the corresponding area in the input.

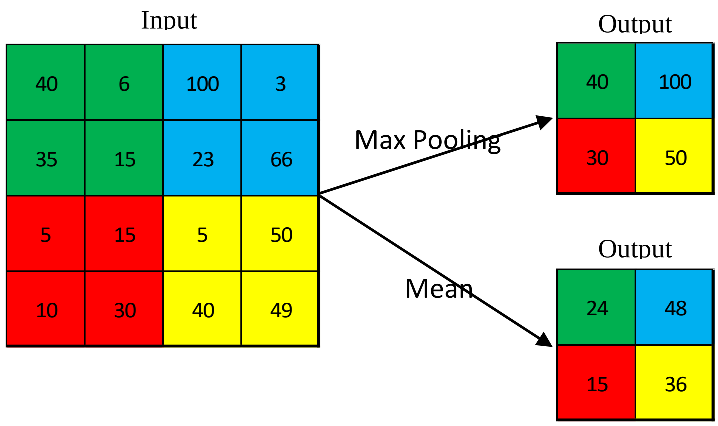


Figure 11- Mean and max-pooling Process.(Author)

2.2.1.c Activation function

Mapping the input to the output is the core function of all types of activation function in all types of neural networks. Non-linear activation functions are universally accepted as the optimum function variety for the creation of a successful CNN. Theses Non-linear activation layers are employed after all learnable layers i.e. layers with weights such as fully connected layers and convolutional layers. The use of these functions provides a nonlinear mapping of the input to the output facilitating the models the ability to learn extra-complicated things. In addition to complicated learning, the activation function capability to expel data is crucial in the training process as it allows the back-propagation of error during training an integral part in the optimization of a model. In the realm of CNN's, three functions are typically employed as the nonlinear activation function for a model development the sigmoid function, tanh function and the Rectified Linear Units function (ReLU) being function in question. The sigmoid function ($f(x) = (1 - e^{-x})^{-1}$) and tanh function ($f(x) = \tanh(x)$) are two saturating nonlinear functions meaning that the values are compressed within a range such as [-1,1] and [0,1] for tanh and sigmoid respectively. The ReLU activation function was introduced in (Hinton, 2010) as a new non-saturated nonlinear function as a substitute to that of the sigmoid and tanh functions. One of the greatest advantages ReLU has over other activation functions is that it does not activate all neurons at the same time. Figure 12 illustrates the distinguishing factor that separates the non-saturated ReLU function from the other two saturated functions as the ReLU function converts all negative inputs to zero. In the context of image data, ReLU is the optimal activation function due to its management of negative numerical information. The ReLU's transformation of negative values to zero is ideal when dealing with image data as any negative image brightness data that may have been created during the matrix multiplication in the convolution process is subsequently nullified. This makes it very computational efficient resulting in the ReLU function convergence six times faster than tanh and sigmoid activation functions.

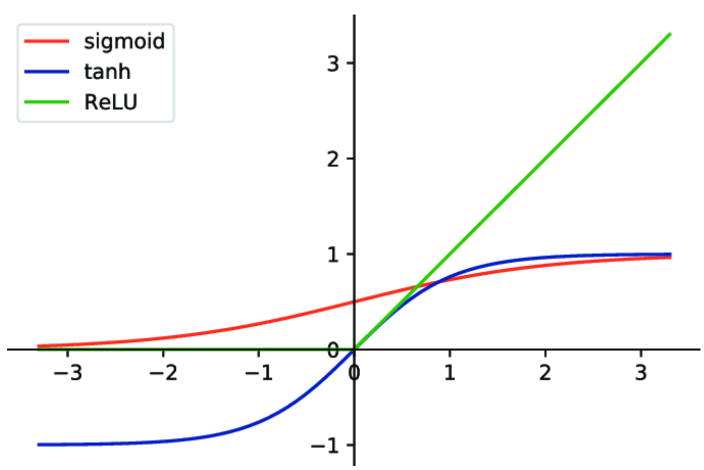


Figure 12- Nonlinear Activation Functions. (Anon., 2021)

2.2.1.d Fully connected layers

Fully connected layers are the portion of the CNN where the system reduces the output of the preceding layers to the number of classes for which the system gives a prediction score. Fully connected layers are the last block of layers before prediction, so the network combines all the information gathered to make an integrated prediction, each neuron is connected to all neurons of the previous layer hence the name Fully Connected (FC) layer. Due to the Fully connected nature of the neurons in a FC layer, the number of parameters dramatically increases compared to a convolutional layer. The input of the FC layer comes from the last pooling or convolutional layer where the feature maps are condensed to form a vector. This vector is subsequently input into a FC layer and an output vector produced.

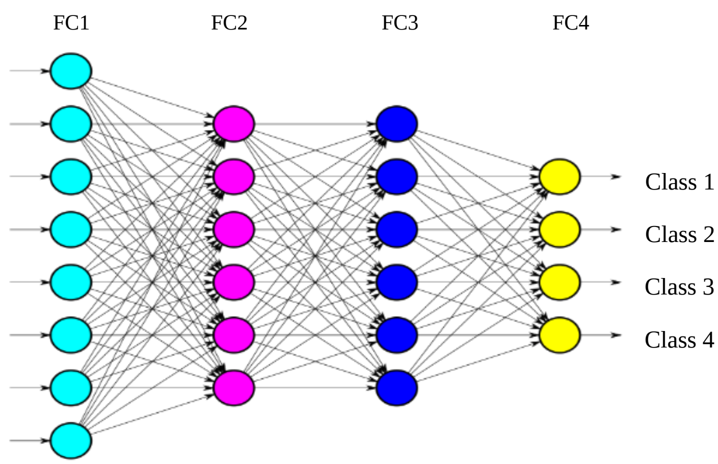


Figure 13 - Fully connected layered structure for a 4-class problem. (Anon., 2021)

The output vector is altered during the passing through the layer as a result of the layer implementing an activation function. Traditionally the FC layers similar to the convolution layers utilize the ReLU function as the activation function except for the classification layer. The classification layer is the last layer of the CNN model with a vector output size of N where N is the number of classes in the image classification problem with Figure 13 displaying a FC layered structure with FC 4 representative of the classification layer of a 4-class problem. The preferred activation function in the classification layers is the SoftMax function owing to the nature of it being a saturated nonlinear function permitting each element of the classification layer output vector to indicate the probability for the input image to belong to a class.

2.2.1.e Training methodology

Two distinctive training practices can be implemented in the development of CNN for a particular task. The more conventional of the two approaches utilized in the construction of a CNN is training a model from scratch. This method is essentially where every parameter encapsulated within the model regarding the number/size of layers, number convolution operations and determination of convolution operations are determined by the author. This being the conventional method of CNN construction the task appears a very daunting and complicated undertaking however with a brief understanding of the theories mentioned previously in addition to some rules of thumb the task hugely reduces in complexity. To create a model utilizing this training strategy the success of the model is highly dependent on the Data as there is a need for an abundant amount of labelled data that resembles the distribution of the testing set of data permitting for the ease of quality training i.e. not overfitting the Convolution neural network. Yet in most situations, this idyllic availability of sufficient quality data to facilitate the training is simply unrealistic. As a result of this idealistic view on the direct availability of data to train a CNN coupled with the expensive and time-consuming nature of gathering sufficient training data makes it incredibly difficult/expensive to develop a unique CNN classifier using this method.

The alternative technique of training tackles the problems that arise when training a model from scratch and is referred to as Transfer learning (TL). The concept of TL refers to the situation of knowledge learned in one domain being exploited and implemented in a new domain to improve the generalization. We as humans have an innate capability of transferring knowledge across tasks.

The stronger the correlation between tasks the easier it is to transfer our knowledge. If we take an example involving sport someone who has played a lot of badminton in the past attempts to play tennis, due to the similarities in strategies and skill execution this person will have the capability to learn tennis faster than a complete beginner. Essentially the use of transfer learning is shifting the location of the start point on the learning curve from the beginning to a latter part of the curve. When referring to CNN's the knowledge transferred comes in the form of weights bias and feature extraction ability. The transfer of weights and bias will enable the data to navigate through the CNN at a more efficient pace allowing for faster more efficient training along with enabling better results.

This transfer learning strategy is the frequently adopted approach of training when using image data to train/test a CNN. This is primarily due to the success that has been achieved over the years with image classification tasks such as that of the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). Some of the most famous Image classifiers have come from the ImageNet completion such as AlexNet, Visual Geometry Group (VGG), ResNet and GoogLeNet to name a few. The ILSVRC is an object detection and image classification task across a wide variety of objects spanning 1’000 classes. The models mentioned have been trained on millions of images spanning the 1’000 classes for days or even weeks. This rigorous training of the model over a vast variance of images causes the model to learn and become adept at extracting features from photographs. The layered structural nature of CNN’s supports the extracting features at different areas of the structure the generic features such as edge and colour information are extracted in the earlier layers with the later layers encoding shape and textures specific to its classification metrics. Knowing this information the decision on how much or how little of the information is required to be transferred for their specific task is made in advance of training commencing. However, even though these models are trained for generic object detection of over 1’000 classes spanning animate and inanimate objects it does not guarantee that the transferred model will perform well when on a custom task. The fulcrum to the success of a model is still unchanged to that of a scratch model with success dependent on the data however the transfer learning method requires exponentially less data to learn the particular feature of the custom class. This reduction in the volume of data needed drastically reduces the training time as well as the data collection volume making CNN even more accessible for everyday tasks. Civil engineering has benefited from the development/ availability of this technology with significant growth of research utilizing this technology for external structural health monitoring of civil structures thus showing a desire for more automated inspired inspection approaches.

In (Feng, et al., 2017) the area of damage detection in Hydro-Junction Infrastructure is investigated. The architecture of Inception v2 is transferred which is another example of a successful model that has emerged out of the ILSVRC. The model is implemented to classify images to be intact or to be considered damaged subsequently classifying any damage like a crack, spalling, seepage or to contain exposed rebar. The model has trained off 14’873 images validated off 1’903 images and finally tested on 1’829 images. This model was trained using over 100 epochs attaining training and validation accuracies upwards of 98% and test accuracy of 96.8%.

A short comparative study is carried out in (Sharma, 2020) where three crack detection CNNs are created using transfer learning. The three models that are transferred are AlexNet, GoogLeNet and ResNet. All 3 of the Models in question are trained using the SDNET dataset from (Sattar Dorafshan, 2018). Transfer learning is being applied in a wide variety of areas in society with exceptional results being achieved especially in medicine where transfer learning models were able to achieve better results and thus outperform a CNN trained from scratch such as in (N. Tajbakhsh, 2016), (Dongmei Han, 2018).

*2.2.2 **Crack detection using CNN***

The removal of obstacles such as the availability of data combined with the effortless access to substantial computational power has made the creation of CNN’s for novel tasks easier than ever before. This reduction in difficulty has permitted the technology to gain a lot of traction in the last number in diverse aspects of civilization. However, the field of civil engineering still falls far behind that of its societal counterparts. The volume of literature utilization deep learning for surface structural health monitoring is growing but still quite modest in comparison to other object detection tasks. The current accessible literature is quite diverse in practices whilst striving to achieve a similar end goal of detecting the defects as accurately and quickly as possible. In this section, the author has compiled a comprehensive but not complete list of research of deep learning being implanted in tunnelling and other civil infrastructure for crack detection. Prior to a CNN model being employed in a sector for any task, the algorithm must undergo stringent training to become capable of performing the desired operational task. There are two primary approaches commonly utilized for such training are referred to as training from scratch and transfer learning. The two training methodologies in question have pros and cons for implementation with the preferred or optimal method determined by the data available or active assignment.

The studies of (Attard, et al., 2019), (Dung & Anh, 2019), (Gopalakrishnan, et al., 2017), (Mei & Gül, 2020) are examples of research that implement transfer learning to create a crack detection classifier all of which have a distinct approach in achieving the identical goal. In (Attard, et al., 2019), a region-based approach referred to as mask R-CNN is utilised in the detection of cracks from concrete surface images. This study implements transferer learning using the ResNet 101 architecture with the weights pre-trained on the COCO dataset trained and validated on only 160 images of 256 x 256 pixels in size for training and validation. This ResNet 101 model serves only as backbone feature extraction processes to create feature maps for the Region Proposal Network (RPN). This RPN examines the feature maps provided and outputs region of interest (ROIs) with crack probability scores. These ROIs are subjected to a Faster R-CNN classifier to make a classification prediction and create a corresponding mask of any crack objects. The mask R-CNN strategy proves a highly effective training strategy attaining a precision value of 93.94% and a recall of 77.5%.

(Dung & Anh, 2019) performs a comparative analysis of the three pre-trained CNN models of VGG, ResNet and Inception with each of the models being retrained on 40’000 227 x 227-pixel resolution images. The purpose of this evaluation is to quantify the best pre-trained CNN to perform the encoding for a Fully convolution network (FCN). A FCN is a variation on the commonly used method of training that typically feeds the data into the CNN architecture before passing the encoded information into a fully connected (FC) layer to perform the classification task. The FCN still performs the encoding process with the CNN but instead of passing this information into a FC layer, the encoded data is input into a secondary decoding CNN to make the classification. The evaluation resulted in the VGG-16 model structure being used for the

FCN which is trained end-end on 500 annotated images achieving roughly 90% in training validation and testing max f1 and precision.

The studies of (Ren, et al., 2020), (Li, et al., 2020) are examples of this encoder-decoder method being used to train a CNN in the tunnel environment with the models implement the training from scratch strategy as opposed to the transfer learning method used by (Dung & Anh, 2019)The research conducted by (Zhang, et al., 2017) operates a distinctive five-layer CNN in which the conventional pooling layer following a convolution layer is not being employed to maintain the structural size integrity of the image through the entirety of the CNN model. This methodology of space invariance operates on a similar ethos of the encoding decoding technique of having the desire to have the image being classified at the backend of the CNN model to be the identical size as the input image

Comparable to (Attard, et al., 2019)the studies of (Gopalakrishnan, et al., 2017) use a transfer learning model in this case, a VGG16 model with the weights transferred from ImageNet as a feature extractor. The features extractor generates semantic image vectors which are used to perform extensive experimentation to discover the best possible final classification layer with solutions varying from neural networks to logistic regression. The best performing classifier was found to be a neural network implementing the Adams optimizer achieving 90% for accuracy, precision, recall and f1 score.

Unlike the aforementioned transfer learning models, (Mei & Gül, 2020) focused on creating a 201 layered structure to be trained on the ImageNet database directly just like the VGG 16 or ResNet models mentioned previously. This approach permits the transfer of knowledge but this time by the users created model. This allowed for a direct comparison between the two training methodologies. This research used just 70 images which were cropped into 128 x 128-pixel images. These images were then manipulated with images augmentation and flipped horizontally and vertically resulting in a database of size 62’790 image patches. These images were used to train the user-defined 201 layered structure from scratch and the transferred model. The model trained from scratch achieved roughly 90% accuracy as regards the performance metrics of recall, precision and f1 score but was outperformed on all these metrics by the model with the information transferred.

Like transfer learning models the models executing training from scratch as the approach for the creation of crack detection algorithms are incredibly diverse in their techniques to extract the desired features to categorise a crack. (Cha, et al., 2017) uses a framework trained from scratch operating with 4 convolution layers to detect cracks in concrete buildings. The study investigates the influence the training database size has on the network attained validation and training accuracies. The experimental database sizes vary from 2k to 40k with the investigation advising the use of more than 10k images for the training of a model. The optimal performing model from the training set is deployed in a scanning window method to test its performance utilizing 55 images with resolutions of 5,888 × 3,584 pixels achieving a 97% testing accuracy.

(K. Makantasis, 2015) proposes a fully automated tunnel assessment approach. The constructed examination procedure subjects the raw input image data collected from a single monocular camera to a series of low-level feature extraction algorithms. The multiple feature maps created during the extraction process are all combined with the intensities of the raw pixels to create a feature vector encompassing visual information that distinguishes each one of the image pixels. The Convolutional Neural Network (CNN) deep learning-based approach where the low-level feature vectors are used to hierarchically construct high-level features to describe the tunnel defects and a Multi-Layer Perceptron (MLP) performs the classification task. This tunnel assessment approach was trained from scratch on over 100’000 sample images from the 3.5km of the Metsovo motorway tunnel in Greece attaining 88.3%, 89% and 88.6% for accuracy recall and F1 score respectfully when tested.

(Protopapadakis, et al., 2019) similar to (K. Makantasis, 2015) created a deep learning crack detection mechanism for concrete tunnel surfaces. The designed methodology leverages a deep Convolutional Neural Network classifier trained from scratch to output binary annotate images containing suspected crack areas from the input image. A heuristic image post-processing step is applied to the annotated image to eliminate noise and pinpoints the (x, y) coordinates of a crack position within the image. The presented CNN architecture overall outperforms the conventional methods of KNN’s, SNM’s and many more in terms of detection performance and computational cost.

The studies of (Chen & Jahanshahi, 2018) proposed a new deep learning framework called NB-CNN to detect cracks on underwater metallic surfaces from nuclear inspection videos. This CNN is assembled using 4 convolution and pooling layers concluded with 2 fully connected layers performing the classification. The CNN is trained on 5326 image patches of size 120x120 pixel resolution. To facilitate the training from scratch this smaller database is subjected to heavy image augmentation and multiplied by truncated Gaussian random variable resulting in 147’344 image paths. The originality of this research is formed in the data fusion scheme of accumulation of crack patches referencing the same location from multiple video frames being stored and referred to as tubelets. These tubelets are subjected to the Naıve Bayes decision-making algorithm discarding non-crack tubelets more effectively than other methods significantly improving accuracy. This proposed data fusion scheme successfully discards false positives and generates a bounding box surrounding the remaining crack tubelets truly representing the real cracks.

The research portrayed in (Menendez, et al., 2018) presents the ROBO-SPECT European FP7 project which epitomises the definitive goal of automation of tunnel inspection. The project implements a multi-degree-of-freedom robotic system composed of a mobile vehicle with an extendable crane. Located at the head of this crane is a pan and tilt mechanism holding two sets of cameras, a lighting system, a laser profiler and a high precision robotic arm with an ultrasonic sensor attached. The multitude of sensors mounted on the crane enables various operations to be performed concurrently. The inspection procedure applies the first set of cameras in conjunction with the CNN developed in (K. Makantasis, 2015) to perform a real-time defect inspection of the tunnel lining simultaneously the laser profiler sensor is inspecting the tunnel structural deformation with an accuracy of ± 2 mm. If a crack is detected by the CNN this data is passed onto the robot to move the crane head to the proximity of the crack area. Once in the crack vicinity, a secondary close up image is captured of the crack before the robotic arm is moved to touch the tunnel lining to collect ultrasonic data on the width and depth of the crack. All the data that is gathered during the inspection is sent to a site referred to as the control room for processing by the Structural Assessment Tool. The inspection system is tested on a 20 m tunnel segment where 2 hours and 9 minutes was necessary for the completion of the full inspection attaining an overall maximum error of 11 cm.

Even though the number of studies utilising CNNs for surface structural health monitoring is growing and gaining momentum the majority of the literature is focused singularly on detecting the cracks. This thesis concentrates on developing this concept of crack detection utilising a

CNN a step further to classifying crack directionally in the CERN tunnels while still exploiting the powerful CNN structures.

CHAPTER 3

Model Development

The variability within the domain of Ai possesses is impressive with the frontiers advancing at breakneck speeds with methodologies considered innovative today potentially outdated tomorrow. The catalysts to the development over the last number of decades are centered on the increased ability to store and analyses data. This technology enhancement of data storage in particular proved revolutionary as an exponential volume of data could now be cumulatively stored and is referred to as Big Data. This term of Big Data generally denotes a large quantity of data with the types and format varying from structured numerical data in traditional databases to unstructured data in the form of text documents, emails, videos, audio, stock ticker data and financial transactions. The ever-developing technological landscape of the world that we live in has enabled more and more aspects of our lives to become tracked/ monitored creating a specific data profile of ourselves.

This collection of data has become extremely useful to us individually while on the other hand has become extraordinarily valuable to industry so much so that data such has been referred to by Peter Sondergaard as “the oil of the 21st century and analytics the combustion engine”. The value in the data is attributed to the essential role it plays in powering machine learning and artificial intelligence models in carrying out numerous tasks from the seemingly small aspects of life such as Netflix's and YouTube recommendation engines to the higher end in the creation of driverless cars. The power that data holds should not be underestimated with data being the singular most influential element surrounding model development. In essence, an artificial intelligence model can only ever perform as well as the data to which it is exposed.

Data is the overriding feature in model development as the volume, type, variability, reliability and resemblance to the task are all mitigating factors when deciding upon the nature of the model being created followed closely by the training technique that can be exploited. Since Deep Learning (DL) has the feature extraction phase integrated into the algorithm in comparison to Machine Learning (ML) where the features are explicitly defined by the user the data selection surrounding tasks implementing DL is of increased importance. The significance of the data selection is due to the need for the model to learn the distinguishing aspects of the task which is incredibly difficult if the data does not resemble the task. The data selection is not the solitary characteristic of data that influences DL models with the volume of the available data for the required task likewise a hugely influential factor. The volume of the data is a definitive factor concerning the training of a DL model with enough data the model can be trained in any way the user feels adequate however if there is not sufficient quantity to train from scratch then alternative options must be utilized to perform training. The prominent alternative to this conventional training structure is referred to as transfer learning (TL). This phenomenon of TL is previously described in the last chapter of this thesis as the method of model manipulation of a formerly trained DL model to a new task. A secondary course of action of using publicly available data that resembles the specific task to bulk up the database can be utilized in addition to using transfer learning or if sufficient volume of publicly available data the traditional technique of training from scratch can be availed of.

Public data sets are incredibly popular in addition to being enormously useful for model development in seemingly niche areas. The University of California Irvine (UCI) Machine Learning Repository is extremely popular in the machine learning community due to variation caused by 588 datasets making up the repository. This variation has empowered the creation of an immense number of models with hugely different purposes from classifying wine to the characteristics of Solar flares with the repository being cited over 1’000 times since its creation.

The sector of Computer vision in particular, has benefited from the publicly availability of copious quantities of data famously with the ImageNet database of 1.2 million images spanning 1’000 classes forming the database for the ILSVRC. This competition holds a particular space in the history of computational development as the success of deep learning neural network techniques traces back to innovations used in submissions to the ILSVRC. The ImageNet database spans 1’000 classes of a very wide spectrum from food to transport even to animals despite this comprehensive range of classes cracks are unfortunately not included. The availability of such public data sets which include cracks is extremely limited mainly due to the specificity of use of such data. The largest and most popular data set of its kind is the SDNET2018 dataset, which contains 230 images of cracked and non-cracked concrete surfaces (54 bridge decks, 72 walls, 104 pavements) segmented into 256 × 256-pixel resolution sub-images. Each sub-image is labelled as 'Cracked' if there is a crack in the sub-image or 'Non-cracked' if there is not a crack. The dataset includes cracks as narrow as 0.06 mm and as wide as 25 mm. The images within this dataset contain a variety of obstructions, including shadows, surface roughness, scaling, edges, holes, and background debris.

3.1 Building a Database

“The goal is to turn data into information, and information into insight.” – Carly Fiorina

The desired information sought from the dataset designed is to define the characteristics that quantify a crack. This information will support the inspection and help provide an unbiased insight into the surface structural health of the tunnel infrastructure. For computer vision to be effectively applied in a certain scenario a well-defined goal, dataset and structure are essential for a successful implementation. This research aims to create an artificial intelligence algorithm of sufficient precision enabling non-subjective structural heath monitoring inspection of the tunnel infrastructure at CERN. Due to the intricacies and specificity, an algorithm like this requires to perform such a task a new specific database is required to resemble said task. The physical characteristics of tunnel infrastructure provide a quite distinctive and difficult environment for the collection of quality image data, where the lighting conditions and obstructions are the more noticeable obstacles for image processing. Furthermore, this collection difficulty that the environment leads there to be no publicly available dataset for cracks in tunnels to facilitate in helping the training of a model. Although the SDNET2018 dataset is available and contains an abundance of crack data from concrete bridge decks, walls, and pavements, the vast majority of data there does not resemble that of a tunnel environment for numerous reasons illustrated in Figure 14. Demonstrated in this figure are the four prime reasons why the crack data from the SDNET2018 dataset was deemed unsatisfactory to be used in the development of a CNN model for crack detection in concrete tunnel linings.

- The first reason presented in Figure 14 as (A) is the lighting conditions. As the SDNET2018 dataset is created using images from above ground civil infrastructure the lighting condition in these images is non-representative of the darker/ bleaker underground environment of a tunnel as shown in Figure 15.
- The second reason portrayed as (B) Figure 14 is that these civil infrastructure of the footpaths and bridge decks surface roughness are considerably rougher than a concrete tunnel lining at CERN. This roughness causes there to be a significant volume of noise in the images which creates unnecessary complications in the image analysis during CNN training.

- The last motive for the exclusion of this data from the database for this research is the dimensional variation of the cracks in the dataset. The majority of the cracks from the wall section of the SDNET dataset are quite small in size with some as small as 0.06mm. The miniature cracks of these cracks depicted in Figure 14 (C) make it difficult for a model to extract crack features from the said image to decipher them from a non-crack image.
- On the other extremity of dimensional variation, a portion of the cracks in the dataset is vast in size up to 25 mm wide illustrated in Figure 14 (D). The size of the cracks is not the problem it is the fact that cracks this size are uncommon in tunnel linings and therefore this proportion of the dataset is unrepresentative of a tunnel structure.

Figure 14- SDNET samples ((Sattar Dorafshan, 2018)

Figure 15 – Examples Cracks from tunnel environment with difficult lighting conditions (Captured by Author)

The issues described in this dataset drastically reduce the quantity of data considered to be suitable and the extraction of beneficial data is extremely tedious and time-consuming. Furthermore, an assignment such as this requires a precise dataset that incorporates all the intricacy's that are associated with such an environment. To facilitate these needs the dataset was created consisting of one hundred per cent images of tunnel linings images more specifically the tunnel lining at the European Centre of nuclear research (CERN). Owing to the radioactive and distinctive nature of the tunnel environment at CERN the customary practice of periodic inspection is not as simply executed in comparison to a standard tunnel. In the CERN tunnels, as Large Hadron Collider (LHC) along with other underground technologies need time to cool down before the areas become deemed safe to inspect. These periods for inspection in question are known as long shutdowns (LS) with the first taking place from February 2013 until March 2015 with the current shutdown LS2 beginning in December 2018 and is scheduled to finish at the end of 2021.

During the current shutdown (LS2) and inspection period all the cracks, spalling, leakages and any other faults found were logged and photographed for reporting purposes. This collection of image faults in the tunnel created a significant database of crack images however as the images captured were photographed for reporting purposes with a dual purpose of logging the existence of a crack but also highlighting the location within a section of the tunnel. This causes the cracks in a portion of images to become the secondary and not the focal point of the image. This dual-purpose during the image capturing inevitable diminishes the volume of the images considered suitable for this research, as substantial number of images are not of significant-quality primarily due to the features of the crack not being in satisfactory focus to be conceded useful. To create a dataset considered beneficial in the creation of a CNN algorithm this reporting dataset had to be refined down to images where the cracks are of sufficient focus and quality. Once this refining process had concluded it was found that the volume of quality images were not sufficient bolster the dataset the author compiled images that were collected with the focal point of the image the cracks. These images predominantly came from the TT61 and TT10 tunnels as seen in Figure 16. These tunnel sections in particular where chosen as they are inclined tunnels consequently making them more prone to cracking enabling collection of the appropriate quality data .

The resultant compiled data constituted of 360 images where the crack is the central figure of the image with there being variation in the sizes, lighting conditions, the volume of obstruction, angle and distance to the tunnel lining spanning the whole database. The disparity contained within the dataset is an integral part of the developmental process of a robustness algorithm by facilitating the exospore to as many variations of a tunnels environment as possible thus creating a better-adapted CNN model. To optimally utilize a database concerning a CNN it is common to segment the database into three separate subsets of training, validation and testing each of which performs a distinctive role in model development. The training subset is repetitively feedback into the model to learn the distinctive features associated with each class of the given task to properly fit the model. The validation data performs a slightly different task with the data being used to provide an unbiased evaluation of a model during the tuning model hyperparameters. The last subset of the dataset is the Test set these samples are used to provide an unbiased appraisal of the fully trained model's performance.

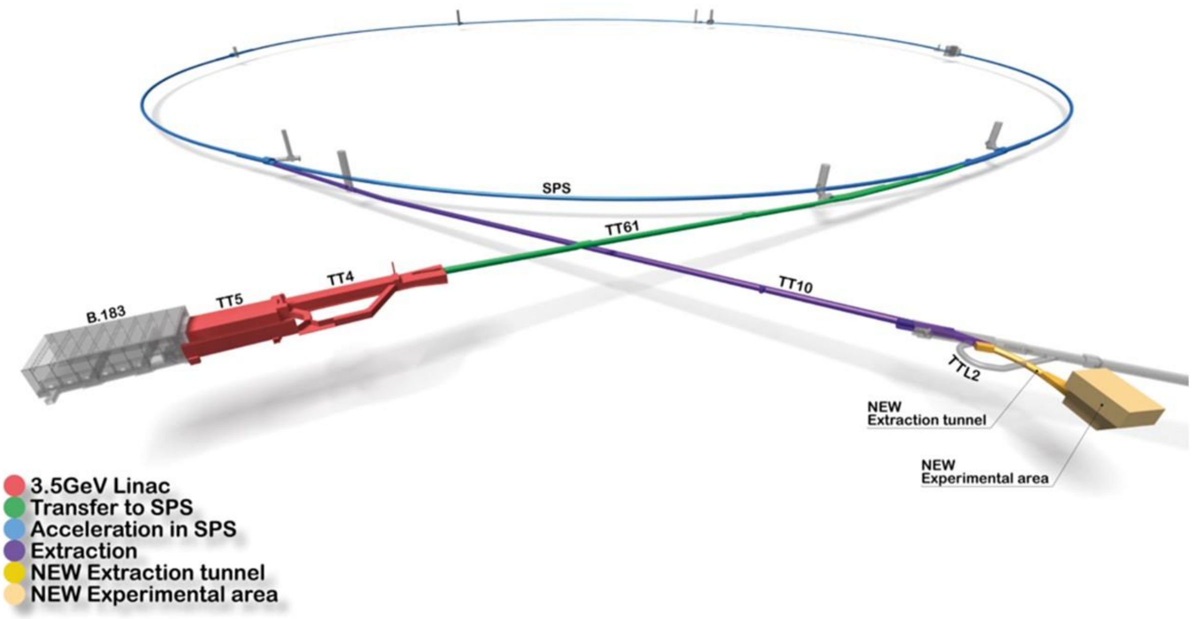


Figure 16 - CERN Tunnel schematic (SPS, n.d.)

This particular dataset was split with 330 images used for training and validation purposes with the remainder of the 30 images set aside for the objective of testing of the CNN. The 30 selected imaged set aside for the testing of the model were predominantly (2/3’s) selected from authors collected data with remainder for varying sections of the CERN tunnel infrastructure to best test the models ability to decipher crack across the whole of CERN. The images selected for training and validation were subjected to a segmentation process where these large images were cropped into much smaller images of 256 x 256 pixels resolution (to resemble the size of the images used to train ImageNet models) as illustrated in Figure 17. The segmentation process of the 330 images resulted in roughly 43’000 sub-images yet due to the shape characteristics of cracks being long and slender the proportion of these images do not contain cracks for example in Figure 17 only 26 out of the 144 segmented images contain a crack. These 43’000 sub-images were subsequently manually classified into two categories: contain a crack, and not containing a crack with just over 6’600 of these images contained cracks. This procedure of segmenting the larger images creates several positives concerning the ease of training a model. The first of these positives is the subdivision of these images inherently increases the database that is available for training. Secondly, the reduction of the size of the images facilitates the diminishing of the computational time as larger images occupy more space in the memory but also result in a larger neural network. Lastly, this technique of training a CNN on smaller images enables the CNN to be implemented in the scanning of any image greater than the size it is trained upon.

As with everything, negatives accompany the positives in this scenario the major negative is the time-consuming nature of having to categories the 43’000 sub-images. A further drawback of using such small images for training is the lack of context each image provides in comparison to that of a larger image. This is problematic when dealing with tunnels inspections where the aim is not only to detect but also to understand the reasoning behind the crack formation. The positives of increasing the dataset and reducing the computational time however in this scenario outweigh the negatives of context and characterization of the sub-images.

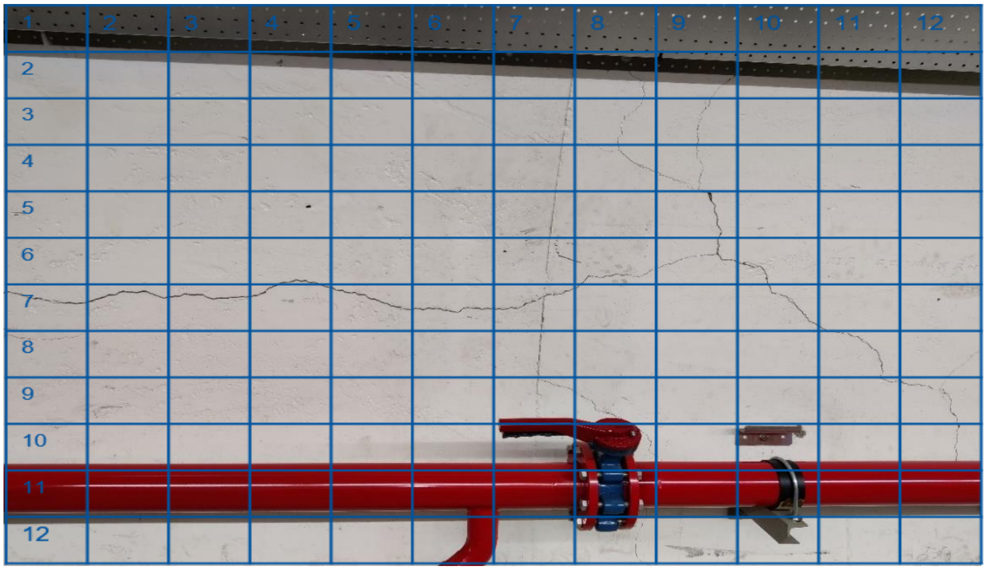


Figure 17- Illustration of a tunnel-lining image being segmented into 144 separate 256 x 256-pixel resolution images. (Created & created by Author)

3.2 Training the Model

All the training/testing of Models in this research are performed using the Google Collab cloud computing service. The Google Collaborator computing service is a service that allows anyone to write and execute arbitrary python code through a browser along with providing access to powerful GPU's. The convenient availability of pertained models such as VGG16 and GoogLeNet in conjunction with the access to powerful GPU's provide a space in which anyone can execute code in a quick/effective manner thus removing some of the obstacles to model development for innovative obscure tasks. As the pre-trained models have been previously trained on a Big Data dataset (i.e. ImageNet) for completely different classification tasks, it has difficulty in tackling our specific tunnel defect classification problem before being trained on the task data set. This necessitates the model to be retraining for the specific task with their three possibilities concerning said retraining.

There is a handful of styles of training that can be implemented during the transfer learning adaptation of a model. The first option is to retrain the model fully on the specific database. This is essentially training the transfer model from scratch, the difference being that the transfer model weights will not be randomly initialized instead, they will instead be adjusted using the transferred model's weights. The second option is the inverse to that of the first thus indicating that all the knowledge that is transferred is set as untrainable. This signifies that the information transferred cannot be altered for the duration of the model training. The final of these three possibilities lies in between the two extremes shown by the others. Meaning that the transferred model has a set volume of the knowledge set as untrainable with the remainder of the knowledge as trainable. The ratio split between trainable/untrainable knowledge is decided upon before the training can commence. All these options are examined in this paper along with some slight variations on these possibilities which are detailed in Section 4.2. All the models trained in this research implemented the same simple training procedure illustrated in Figure 18 and described in detail in the remainder of this section.

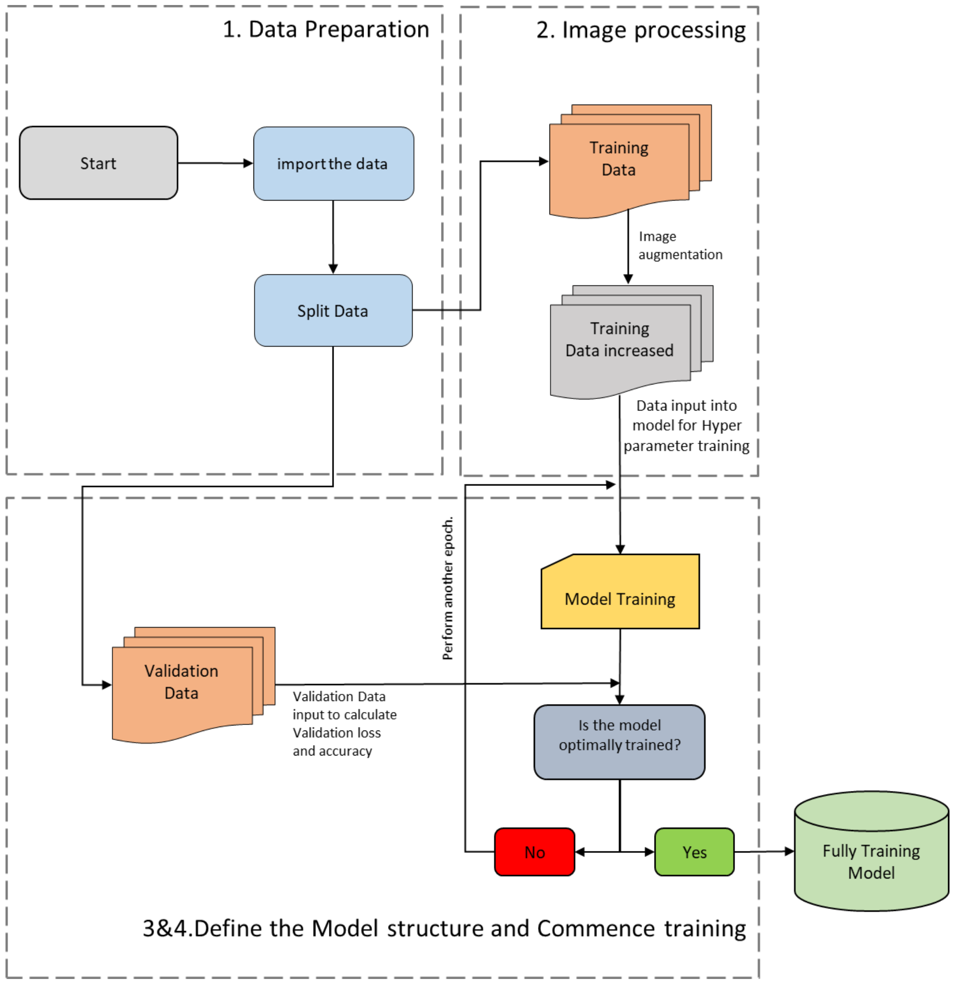


Figure 18- Model Training Flow Diagram (Created by Author)

1. Data Preparation: Load the tunnel crack data images into the code for pre-processing before being fed into the CNN Model. To avoid an imbalance in the database, which would cause bias CNN model training the equivalent number of cropped images that did not contain cracks were randomly selected to create a database of 13’200. This database consists of 10’400 for training and 2’800 for validation images, with a split in of 1:1 to images of images with and without cracks. The batch sizes for the training and validation sets are established as 1% of the size of the respective dataset. The value for batch size determines the number of images used in

each update and a complete update of a batch is considered as an iteration. Consequently, the number of iterations needed to complete the full database is regarded as an epoch for the model training.

Once the image is selected from the database and put into the corresponding subset of data i.e training or validation it is unable to be selected again resulting in to cross contamination of the data. This method of data selection allows for a model to be trained on a randomly selected data resulting in a more robust model in comparison to a model created using a favorable hand-crafted data set. This however results in the images not easily being relatable back to areas of the CERN tunnels

2. Image processing: Concerning image processing, numerous different methods can be implemented to manipulate the images to characterize and enhance the images features. However, some processing methods may transform the morphological features of the raw RGB image data and reduce the credibility for tunnel crack classification. As an alternative processing method, data augmentation is adopted to increase the relatively smaller database size of tunnel data images in this study. This augmentation flips the original image both horizontally and vertically as well as one rotation of through 90 degrees with no change of the tunnel image morphology (see Figure 19). The augmented images are not saved but just used within the code to bulk up the training data images threefold.

3. Define the Model structure: In this step, the model's architecture is decided concerning the training type being utilized. An in-depth description is provided in the later Section 4.2.

Original Image Horizontal Flip Vertical Flip Rotated 90º

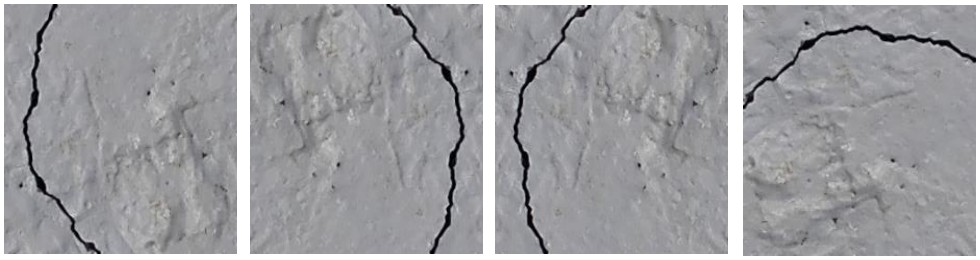


Figure 19-Data Augmentation used in the training of the Crack detection CNN(Created by Author)

4. Commence training: The model training commences after the definition of both data set and model structure. During the training process, each training batch passes through the CNN and the results are evaluated against some performance metrics in the form of accuracy and loss. The accuracy metric measures the correct number of predictions in percentage, while the loss is calculated to determine how much the prediction varies from the true value. Backward propagation of errors known as backpropagation is applied as the last phase in each iteration of the training as a method of fine-tuning the weights of a neural network based on the error rate. The backpropagation fine-tuning of the weights facilitates in reducing the error rates as well as making the model more reliable by growing its generalization. This process can also create a problematic phenomenon known as overfitting, a widespread problem that occurs when repetitively updating a CNN model on the same data. To minimize the influence of overfitting, three approaches are integrated into the model/training structure:

1) Validation data: the use of validation data allows corresponding validation accuracies and loss to be calculated from the CNN model. As this data is not included in the CNN training, it independently validates the model training results.

2) Early stopping: This is a training technique commonly used as a breakpoint for the training. During the training of a CNN model, the training loss will constantly decrease due to

the repetition of updates to the model on the same data the model learning the database labels in addition to the features. Conversely, the validation loss is calculated from out of sample data and will inevitably reach an optimum value and will only plateau or rise from this point. This approach of early stopping works by using the validation loss as a barometer to quantify the quality of training. This is done by equating the validation loss after every epoch to previous epochs if the loss reduces the model's parameters from that precise epoch are saved to be retrieved and used at a later stage. However, if the validation loss does not see a reduction for a set number of epochs the model will conclude the training. In this research, the model is set up to train for a max of 100 epochs with an early stopping threshold of 20 epochs.

3) Dropout: Throughout a CNN's training the neurons making the model are learning features, gaining context on the data and adjusting their weights accordingly. This becomes problematic when the weights of the neurons are becoming tuned for specific features providing some specialization. This causes the neighboring neurons to become dependent on the specialization provided by its adjacent neurons if left alone will result in a fragile overfitted model too specialized to the training data. The technique of dropout implemented during training elevates the codependence of adjacent neurons by randomly dropping a percentage of the network during training illustrated in Figure 20. This results in the responsibility of each neuron as during the training it must step in and handle the representation required to make predictions for the missing neurons. This in effect reduces the network sensitivity in turn resulting in a network that is capable of better generalization and is less likely to overfit. The ratio of dropout is fixed at 50% for the models in this research and is utilized strictly in the fully connected layers of the model.

A B

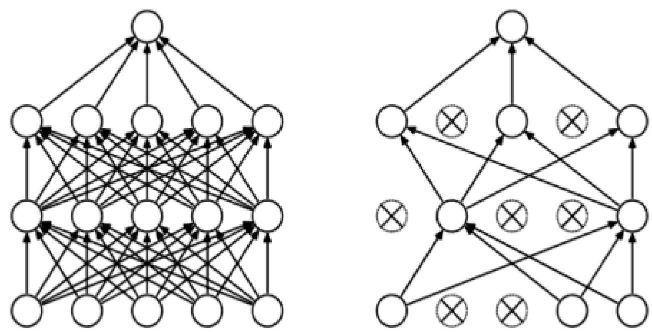


Figure 20 - illustration of (A) before dropout and (B) after dropout is applied

CHAPTER 4

Model optimization

Model optimization is often concentrated on the optimization of hyperparameters of the CNN to
minimize the possibility of overfitting/underfitting. Considering that these optimization techniques are already integrated into the training procedure in the form of validation data, early stopping and dropout the potential problems are severely curtailed or possibly removed. This forces the focus of the optimization of the models to be focused on characteristics of the model development as an alternative to the typical hyperparameters optimization. The characteristics that this research concentrated on are the Data and the Architecture owing to the influence they hold on the development of a successful algorithm. In terms of the choices concerning optimization, the features of complexity and volume are those of the utmost significance.

The scope of the following section is aimed at detailing the rationale and methods used in locating and optimizing the finest CNN crack detection algorithm possible. The chapter is split into three distinct sections with the first detailing the methods used in the refinement of the created database. Followed by the multidimensional analysis of varying experimentation CNN models all with differing numerical structures regarding the volume of knowledge transferred from a pre-trained network. The chapter is subsequently concluded with extensive testing of the various CNN models with the testing comprising of the analysis of problematic areas of the CERN tunnels.

4.1 Data optimization

Database optimization refers to a variety of strategies used to ultimately increase the accuracy, efficiency and applicability of the model. Commonly the use of data optimization reduces the complexity of the process within the model while optimizing the available resources in essence reducing physical processing needs. Data's influence on a model cannot be understated with the quality being the singular most dominant factor in determining the performance of a model in the operational task. The creation of a dataset is fundamentally the easy part as all that is required is the collection of the data. However, the creation of a quality dataset creates more obstacles as "With too little data, you won't be able to make any conclusions that you trust. With loads of data, you will find relationships that aren't real" --Douglas Merrill. Essentially, it is a balancing act of quantity versus quality. This balance causes there to be quite a bit of ambiguity surrounding data especially questions relating to "exactly how much training data do you need ?''. The uncertainty within the question is reflected in the answer with no concrete solution. There are so many influencing factors relating to the task you are looking to perform that affect the volume of data you need such as the performance you want to achieve, the input features you have, the noise in the training data, the noise in your extracted features, the complexity of your model and so on. In the case of this study, the volume of data available is not in question with 6'600 256 x 256-pixel resolution combined with the image augmentation surpassing the 10'000 image recommendation formed during database experimentation in (Cha, et al., 2017). The quality however is an unknown quantity as previously highlighted poor data quality leads to the inability to develop accurate insights/models. To quantify, the quality of the database an initial training/testing of a CNN model is undertaken. The results from this initial CNN model were extremely promising to attain a training accuracy of 82.2% and corresponding validation accuracy of 79.0% illustrated in Figure 21.

Although this initial model archives quite a high accuracy upon further examination of the accuracy graph, slight overfitting is noted. Overfitting is a phenomenon when a model learns the details and noise in the training data to the extent that it negatively impacts the performance of the model on new data. This means that the noise or random fluctuations in the training data is picked up and learned as concepts by the model. Essentially what is occurring is the model is correlating features to classes that are not genuine causing the fitting of the model undesirable well to the dataset of insufficient quality and failing to learn generic features that separate the classes. In the case of this model, the overfitting is not drastic but is still a significant factor in reducing the quality and hence must be managed moving forward.

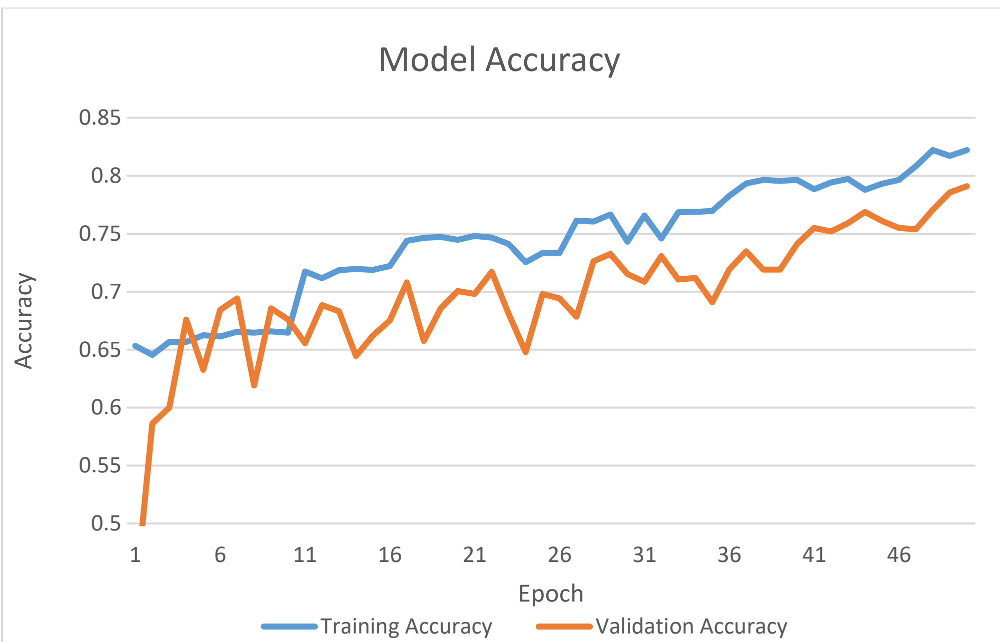


Figure 21- Initial models accuracy metric graphed

Several steps in the form of reduction in complexity and size of the dataset are taken to diminish if not elevate the impact of overfitting. The reduction of complexity follows very straightforward criteria of extracting the crack data where the crack is located on the extremity of the images as depicted in Figure 22. These simple criteria were chosen primarily due to the small nature of the images used combined with the downsampling that inherently occurs to the images as they pass through the CNN during training. This combination considerably increases the difficulty of classification/detection problem as the number of pixels where vital features are present are significantly reduced. This means that the images with cracks strictly on the extremities are less likely to be detected as a crack by the network than those whose crack lies in the center of the image. A secondary motive behind this decision is due to the crack features in these images only taking up a small proportion of an already small image. This increases the difficulty of classification as it is difficult to definitively decide on whether it is a crack, a scratch or only a crack like feature. Following these specific refinement criteria, the datasets complexity and subsequent size was reduced from roughly 6,600 images to 6’100 images.

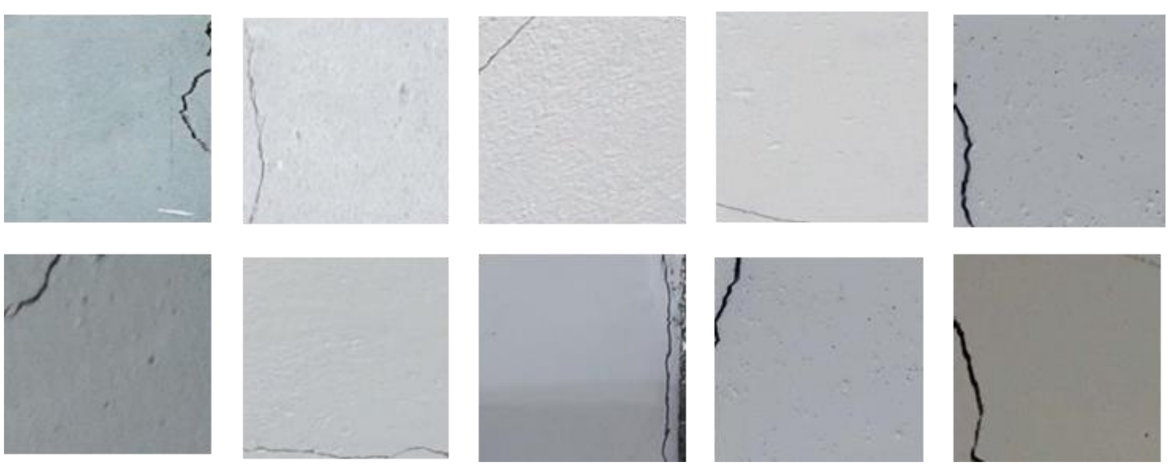


Figure 22- cracks on the extremity of images (Created by Author)

In addition to the reduction of size to the no crack database, a further examination of the data was needed following the repetitive misclassification of no crack images during the training of the model. This continuous misclassification during the initial models training highlighted the model's limited ability in dealing with the common obstructions in the CERN Tunnel environment in the form of pipes and cable trays with examples of miss classified images shown in Figure 23. On further examination of the non-crack data subset, the rationale behind the incapability of the model to recognize the obstruction is clear. Owed to the randomly selected nature of the non-crack subsection in the database not enough of the common tunnel environment obstructions such as the ones shown in Figure 23 are incorporated in the database. These omissions thus produce a model, which is underexposed to the obstructions and unable to learn the intricacies related to these objects. This obstacle to optimization is easily removed by increasing the inclusion of the data representative of these entities. As the introduction of these obstruction images increases the non-crack data subset the corresponding number of images must be removed to keep the unbiased ratio of data intact.

Now with the problematic crack images removed and obstruction images included the database is optimally refined consisting of 9’600 images for training and 2’400 for images for the validation of the model. This concludes optimization of the database and therefore is prepared for application in the architecture developmental stage.

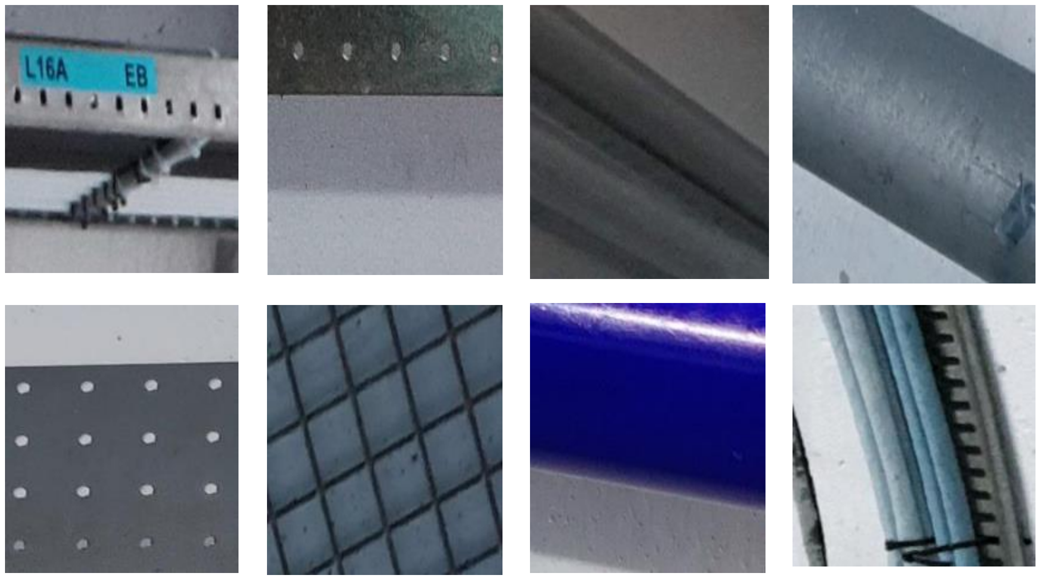


Figure 23- tunnel obstructions

4.2 Architecture optimization

This study implements deep learning methods using Cloud computing. This gives access to pre-trained models structures along with the equivalent trained weights. Specifically, this project implements transfer learning by using the VGG16 model structure a deep convolution neural network developed by the Visual Geometry Group (VGG) in oxford with the weights of the model pre-trained on ImageNet. This ImageNet Database is comprised of 3.2 million annotated images with over 5’000 categories described in (Russakovsky, et al., 2015). Throughout the training of the VGG16 model on these 3.2 million images it becomes proficient at deciphering images classes attributable to its feature extraction capabilities. The proficiency of this feature extraction in conjunction with the malleable characteristics of the model structure has enabled numerous variations on the VGG model to accomplish great results when applied in other areas on other datasets such as (Dung & Anh, 2019) and (Gopalakrishnan, et al., 2017). The adaptability accompanied with accuracy’s reached using the VGG 16 model structure is the motivation behind its implementation in this research. The Vgg16 model depicted in Figure 24, is made up of 14.7 million parameters, contains 13 convolutional layers with small Kernels of 3 x 3, 5 Max Pooling layers with kernels of size 2 x 2 to carry out the spatial pooling. The optimizer referred to as Adam is incorporated into the VGG structure as the optimizer for this research. It is proven by (Gopalakrishnan, et al., 2017) an effective and efficient variant of gradient descent generally does not require hand-tuning. The Adam optimizer leverage's the power of adaptive learning rates methods to find individual learning rates for each parameter. This is done by Adam using the gradients of the loss to make alterations to the parameters to reduce the error in the system. The adaptability of the optimizer allows for better and faster results in comparison to traditional methods as it has taken the best parts of the traditional methods and merged them into one. Thus, making it strong enough to deal with noisy data and sparse gradients and the ideal optimizer for our problem.

The objective of Architecture optimization is to locate the best possible CNN model structure to perform the task of crack detection and attain a comparable accuracy to that of the studies of (Attard, et al., 2019), (Dung & Anh, 2019), (Mei & Gül, 2020). To find which method of training is best suited to the constructed dataset for this research, experimentation on the architecture is performed to find the optimal CNN structure This experimentation consists of training five models under differing training structures as well as training type between transfer learning and learning from scratch with the numerical breakdown of each structure depicted in Table 1.

Two of the five experimental models implemented use the methodology of training from scratch with each model availing of similar but differing strategies. The first is a user-defined model created by the author referred to as the scratch model is made up of 4 convolution layers, 3 max-pooling layers, 2 dense ReLU layers with dropout included and finally a SoftMax classification layer. The structure of this model was created to represent a simple deep learning model as a baseline comparison to that of the complex deep learning model of the VGG16 structure. The second of these two models referred to as Transfer model 1, this model resembles that of the model implemented in (Islam & Kim, 2019) where the Vgg16 structure is applied however the model's transfer trainable parameters are set to zero. The application of the model in this study will allow for a direct comparison between the training methods as well as within the training method of training from scratch. As the difference between the scratch model and this transfer model 1 besides the depth/size of the model is that the transfer model 1 has the weight initialized from the VGG16 structure while the scratch model has these weights randomly selected to begin with. Figure 26 exemplifies the discrepancy between the models emphasizing the influence of the initialization of the weights has on the performance concerning training time and accuracy. The Scratch model takes the full 100 epochs to train which is 58 longer than the transfer model 1 while achieving a 2.73% lower model accuracy. The remaining 3 models in this research all perform the conventional transfer learning methodology like that of (Gopalakrishnan, et al., 2017), (Silva & Lucena, 2018) and (Yuvaraj, et al., 2021). Each of the three models leverages the VGG16 model slightly differently to evoke a distinctive reaction from the model each having varying levels of success. The transfer Model 2 uses the VGG 16 model strictly as a feature extractor. The method of feature extraction sets all five of the Vgg16 Convolution Blocks illustrated in Figure 22 as untrainable making the weights in said layers incapable of updating after each epoch. This enables the extensive training of the transferred model via Big Data to be leveraged to extract notable features at every level of the model. These features are subsequently fed forward to train fully connected layers to decipher crack and non-crack images.

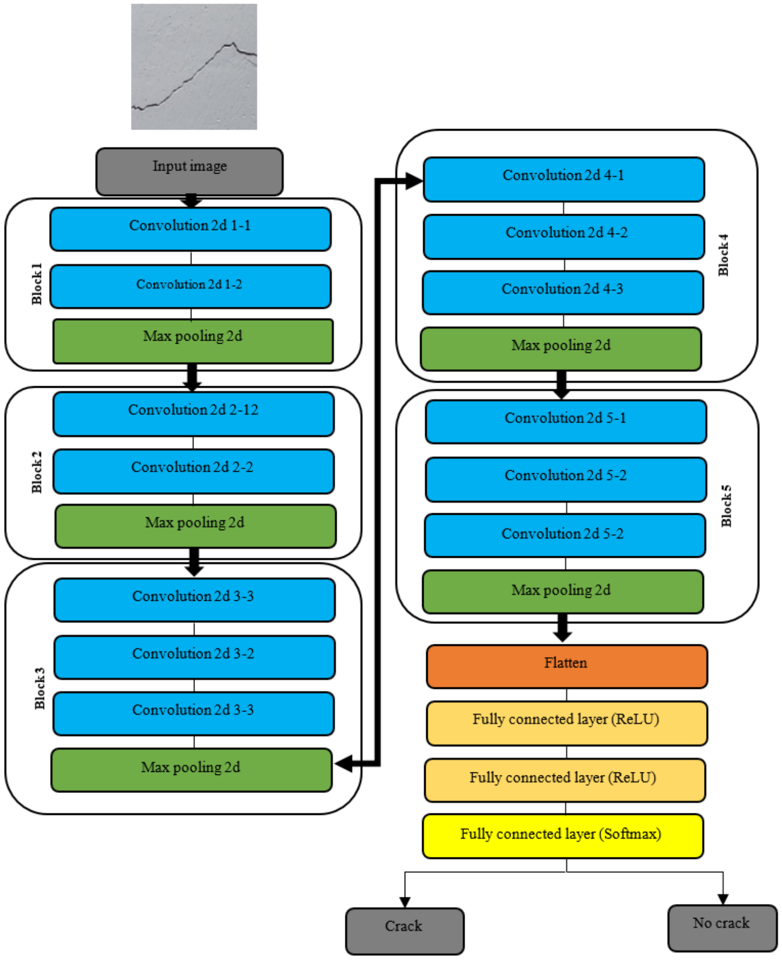


Figure 24- The architectural structure of Transfer Model 2 (Created by Author)

The alternative to this use of a transfer learning model as a standalone feature extractor is to integrate the feature extraction into the model otherwise known as fine-tuning the transfer model. This fine-tuning method of transfer learning refers to the incorporation of the learning process of the data set into the VGG16 model which is carried out by altering the VGG16 trainable parameters by unfreezing some of the Convolution Blocks. The motivation behind unfreezing later layers is that deeper layers in Block 3&4 encode high-level features while the lower layers in Blocks 1,2 &3 detect simple edges and shapes as depicted in Figure 10. This allows for deeper feature maps that contain less information about the image and more about the class of the image. Hence means the later Convolution Blocks detect fewer features but help encode features to their respective classes. By unfreezing the later Convolution Blocks the important structural information of the image can be extracted by the skilled VGG16 model lower layers and the feature encoding to be fine-tuned on the dataset. The two transfer models 3&4 use this fine-tuning model method only varying on the number of Convolution Blocks set as Trainable with dissimilarity highlighted in Table 1. With the model parameter settings, the training and validation accuracy of all five models is plotted in Figure 26 and Figure 26. This figure shows the influence of the optimization of the data and architecture on the accuracy when compared to the initial model results of 82.2% and corresponding validation accuracy of 79.0% demonstrated in Figure 21. Each of the five models achieves extremely high results with the scratch model proving to be the worst while still achieving a training accuracy of 95.7% with the model Transfer 4 attaining the highest results of 98.7%, 98.3 % for training and validation accuracy, respectively.

Model Name	Total Model Parameters	Model Trainable Parameters	Model Non-trainable Parameters	Convolution Blocks set as Trainable
Scratch model	13,340,418	13,340,418	0	User defined structure
Transfer model 1	31,756,098	31,756,098	0	All
Transfer model 2	31,756,098	24,120,834	14,714,688	None
Transfer model 3	31,756,098	24,120,834	7,635,264	Block 5
Transfer model 4	31,756,098	30,020,610	1,735,488	Block 4 & Block 5

Table 1- Parameter breakdown for the varying experimental CNNs

Figure 25 -Plot of Training accuracy per epoch for the 5 experimental CNN models

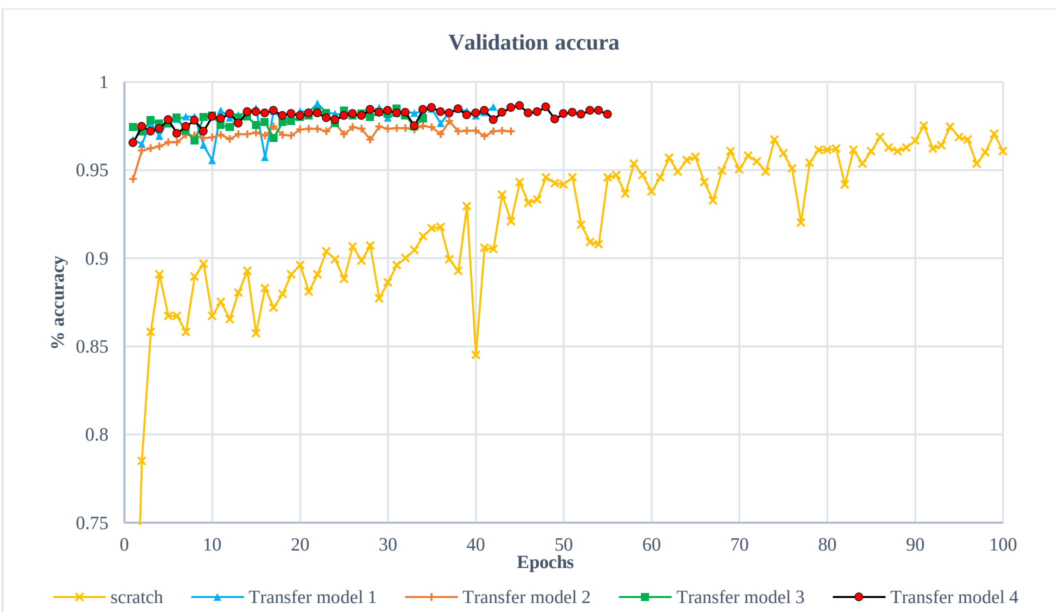


Figure 26- Plot of and validation accuracy per epoch for the 5 experimental CNN models

4.3 Testing the CNN's

The five variations of CNNs described in the previous section each offer something different concerning the training of the CNN and all show fluctuating levels of accuracy. All the five trained models were tested under the same circumstances to find the optimal possible model for our specific task of crack detection in tunnel images with the best model subsequently used moving forward for the crack classification. Furthermore, this method permits uniformity in testing the models as well as providing a view of the robustness and accuracy of the trained and validated CNN algorithms in the tunnel environment. To facilitate an extensive and equal evaluation of the CNN models the identical 30 images are input into each model these particular images were held in reserve for the specific purpose of testing and therefore have not previously been seen by the models during training.

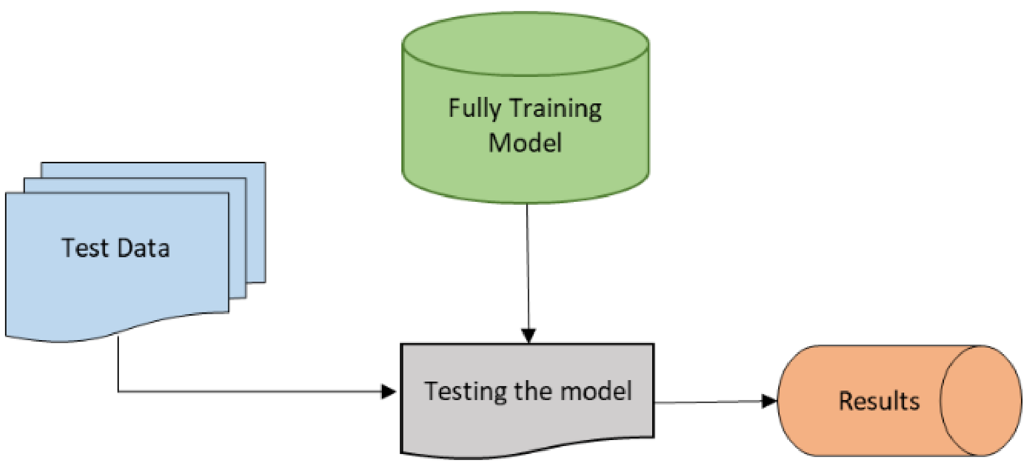


Figure 27- Testing of the Model flow diagram (Created by Author)

The separation of this data from the training process allows for true testing of the models, as the models have not learned any information specific to these images. As previously discussed, the images used in the training process are small in size with the dimensions of size 256 x 256 pixels. This permits the implementation of the CNN models in applying the scanning method to test all of the five iterations of the experimental CNN structures on the 30 large images of 3072 x 4096-pixel resolution As earlier mentioned in the data optimization section cracks that are located on the fringes of images have a higher likelihood of miss classified. To facilitate optimization of accuracy and reduction in miss classification two-pass scanning resembling that of (Cha, et al., 2017) of the test images is applied. The first pass starts at the top left of the image using a 256 x 256-pixel resolution window it starts scanning horizontally from left to right across the image as depicted in Figure 28, each window is classified using the CNN to whether it contains a crack or is considered to be intact. Once the scanning window reaches the furthest right extremity of the image it drops down 256 pixels and works horizontally from left to right across the image once again with this process continuing until the whole image is scanned. If a window is deemed to contain a crack during scanning it will be unaffected however if the window is considered to be intact the contents of the window will be removed and replaced with all white pixels as depicted in Figure 29. The second pass works under the same principles of moving the scanning window around the image however, this time scans within 128 pixels of the extremities of the image demonstrated in Figure 28. This assists cracks that were previously on the extremity of the window in pass 1 now be located in the central area of the window. These secondly scanned images are merged with the firstly scanned image to give a better depiction of cracks detected in the image with an example shown in Figure 29. This two-pass scanning method exposes each of the five CNNs to 10’710 256 x 256-pixel resolution images throughout the testing process

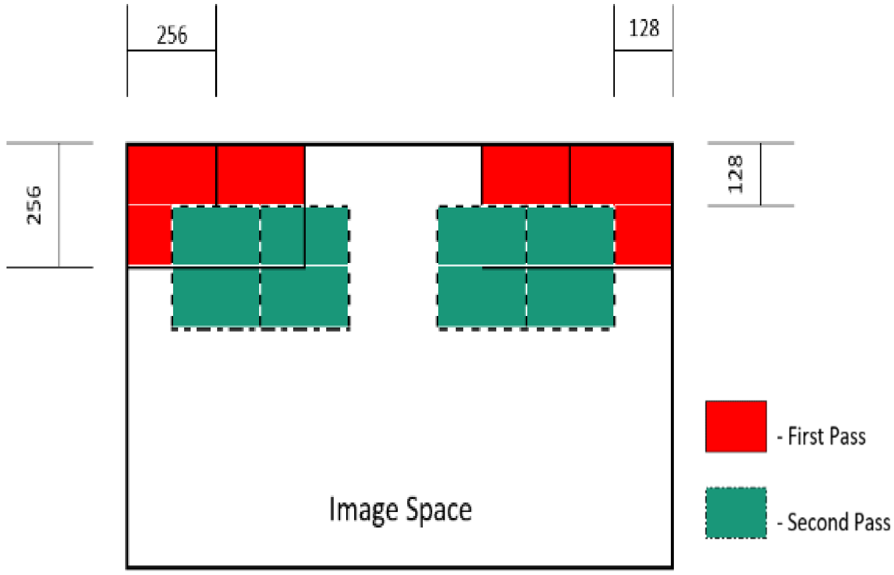
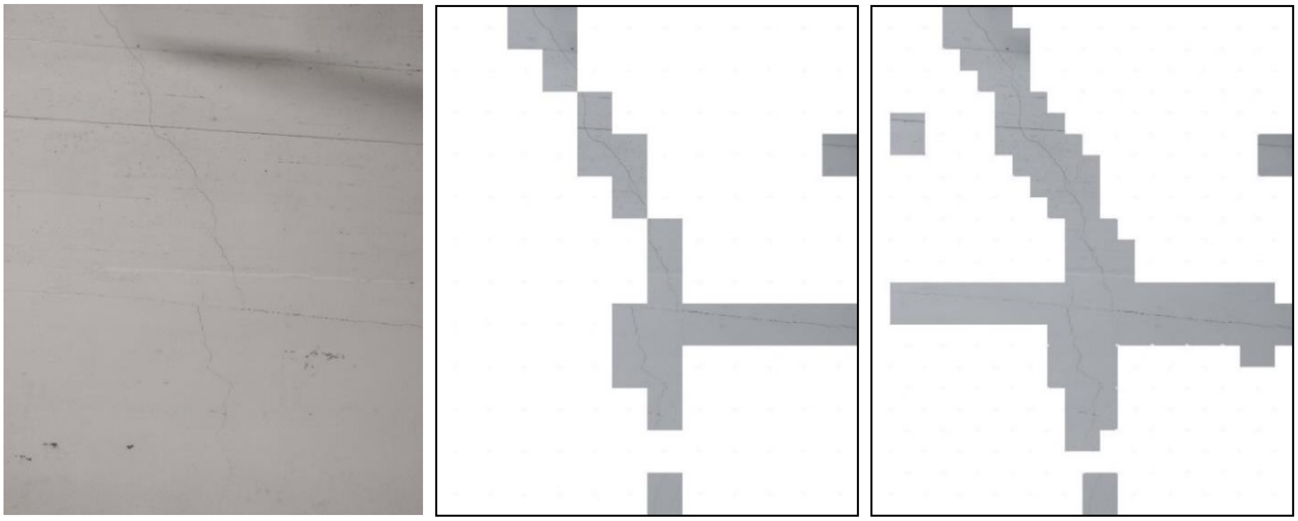


Figure 28- Illustration of the scanning window method (Created by Author)

The images used in the testing of the models were taken from different areas of the tunnel infrastructure at CERN displaying varying luminosity/obstructions within the image. This diversity of image conditions enabled to truly test the accuracy but more importantly the robustness of the CNN model concerning implementation for a full tunnel inspection. As detection is central though not the singular focus of this research with classification following detection also researched additional considerations were taken when collecting this test data with the prominent consideration being capture angle. The variation angle of raw images used for testing brings an extra variable into consideration for the classification of an image. For example, an image taken of a crack from two different angles has the potential to be classified differently. To reduce the influence of the factors of distance and angle on the results all the images used were collected by the author during inspections. This facilitated consistent distance and angle from the tunnel surface with an emphasis on use for CNN testing when being captured.



Original Image | 1st scan Image | Merged Output Image

Figure 29- Depiction of images that have passed through the scanning method.(Created by Author)

To quantify the results for each of the five variations of models used against one another along with other crack detected CNN's the universal performance metrics of Accuracy, Precision, Recall and F1 score is calculated. The metrics used are defined as:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{No. of cracks} + \text{No. of Noncracks}}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

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

These metrics offer a different insight into the performance of the model. Accuracy gives an overall understanding of the model as it is the number of correctly classified image windows over the total number of windows. Precision shows how much of the detected cracks were in fact cracks. While Recall is a measure of how much of the actual cracks were detected. The last metric F1 score is the weighted average of precision and recall this is the metric that is most useful in the appraisal of the model as in the calculation of the metric true positives (TP), False positives (FP) and False Negative (FN) are all used allowing for an overall view on the model's performance.

The importance of consideration of both FN and FP in the assessment of the models is demonstrated in the results. As if the overall goal of the model was strictly on the detection of as many cracks as possible (i.e. recall) the optimum model would be the scratch model but when FP is taken into consideration this model performs the poorest of the five models. This emphasises the importance of this metric in the appraisal of a model with the highest F1 score depicting the most well-adjusted model for the task. The optimum model chosen to move forward with the crack classification is the Transfer model 4. This is because this model outperforms the others regarding achieving the highest result in training (98.7%) as shown in Figure 25 as well as attaining the best testing results by displaying the best F1 score (93.5%) as shown in Figure 30. This model also shows no degradation of accuracy when faced with new images as the model achieves a testing accuracy of 98.4% only slightly below that of the training.

The results attained are comparable to other CNNs created for crack detection such as (Cha, et al., 2017) which attained a 97% testing accuracy, (K. Makantasis, 2015) reaching 88.3%, 89% and 88.6% for accuracy recall and F1 score respectfully when tested and (Gopalakrishnan, et al., 2017) achieving 90% for accuracy, precision, recall and f1 score. The testing, metrics accomplished indicates the model is comparable to its crack detection counterparts while being well adjusted to the complexity that is inherent to a tunnel environment especially the intricacies of the CERN tunnels.

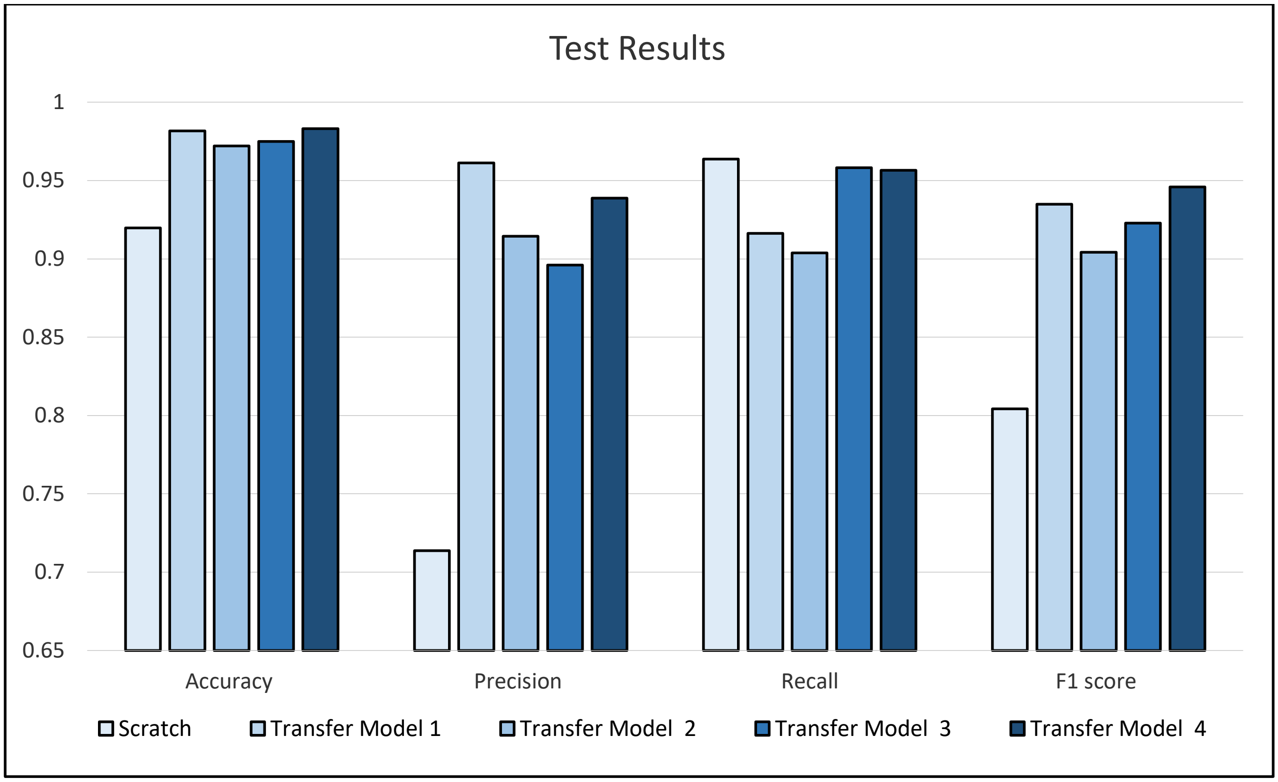


Figure 30- testing results from the variations of the CNN Models

CHAPTER 5

Crack Classification

A crack is widely considered as an initial indication of structural deterioration and the early detection of such a defect is critical in prolonging the working lifespan of the structure. Upon detection, the subsequent classification of this degradation is essential for the interpretation of the surface structural health condition of the structure. The type of classification utilized in the assessment of a structure whether it is based on severity, orientation or dimension is all reliant on the desired outcome, and each type gives a distinct evaluation of the tunnel surface structural health. Historically crack detected via visual inspection are classified by applying the inspector's vast experience which is an inherently flawed and subjective strategy in assessing a tunnel surface structural health/behavior. This subjective valuation of the tunnel lining becomes problematic over the lifecycle of the structure, as the true structural health of the tunnel is by no means categorically identified.

The purpose of this research is to create a crack detection model that is compatible with the existing method of visual inspection while striving to eliminate the problems caused by the traditional approach. The goal is the creation of an algorithm that over time can be optimized/customized to outperform these traditional methods in any tunnel environment. In addition to detection, the further classification of the cracks is conducted objectively to permit an unbiased evaluation of a tunnel's surface structural health plus allowing further insight into the structural actions of the tunnel lining. This unbiased evaluation additionally provides numerical data of the overall surface structural health of the tunnels as well as a further breakdown concerning tunnel chainage. This in-depth breakdown supports a factual view of the tunnel to be created which will highlight the problematic chainage of the tunnel. This sectional breakdown permits inspectors to concentrate inspections on these problematic chainages saving time while also contextualizing the data provided. This chapter concentrates on the novelty of this research and is split into two separate sections: the first one describes the rationale of selecting the crack classes, while the second section subsequently gives the explanation of the image classification as well as the testing results achieved.

5.1 Crack classification

Concerning objective crack clarification, the two prominent crack attributes that can be considered are dimension and direction. Each of these attributes gives a significantly unique perspective on the tunnel. If a tunnel is inspected via the dimensional approach all the cracks will be classified concerning their size. This as a result allows a threshold to be set up to elevate all the cracks under a given size only keeping cracks of significance thus providing a numerical assessment of substantial cracks in the tunnel. However, in a tunnel that is evaluated regarding directionality, all the detected cracks are classified concerning their direction. This approach is innately not as precise as the dimensional methodology yet can still provide an incredibly useful insight as the crack orientation correlates to behaviors in the tunnel. To classify cracks objectively, the traditional classification method was investigated via the tunnel inspection manual (Tunnels, 2015) where it was found that tunnel cracks deterioration were largely separated into five categories:

1. Crescent cracks
2. Shrinkage cracks
3. Horizontal cracks
4. Vertical cracks

5. Diagonal cracks

Owing to the nature of the cracks having very few distinguishing features all the five classes listed above share the majority of their feature with most only separated by subtle intricacies. When you examine these classes under the objective scope of dimension, these subtle intricacies can be extrapolated from the data, as the cracks and their features such as width length angle and curvature can be dimensionally defined. The necessity for dimension to decipher/separate the classes is evident particularly in the classes of shrinkage and crescent cracks. In the case of shrinkage cracks without the width of the crack being established, it is incredibly hard to decipher a shrinkage crack to that of horizontal and vertical cracks. While for the crescent-shaped cracks the defining attribute that separates it from a diagonal crack is the calculation of the curvature. When deprived of the ability to conclusively calculate the dimeson of these essential characteristics the complexity of deciphering a crack is hugely increased. Thus if the capability to dimension these features is unavailable the objective crack classification process must be altered. The modification of the procedure materializes by removing the dimensional depended classes of shrinkage and crescent cracks, therefore, leaving the residual classes to be applied in a directional classification approach.

This clarity of the execution task is essential in the classification process especially on smaller datasets like the one used in this research, as with ambiguity the increased risk of miss classification and reduction and model accuracy is unavoidable. This situation of uncertainty around model’s classes is demonstrated in (Gao, 2018) . This study implements four models in four experiments corresponding to the different tasks of component identification, spalling detection, damage level classification and damage type classification with a vast difference in results across each of the tasks. The first two of these experiments have binary classification results the first concerning component type and the second regarding if spalling is occurring in the image attaining 88.8 % and 85.4% in their respective test sets. The model achieving such high results in these tasks can be directly correlated to the model performing a clearly defined task thus allowing the model to train and learn the complexities relating to the specific tasks facilitating the capability to classify accurately.

However, when this clarity of the task is diluted, and more ambiguity is introduced the complexity and difficulty of classification for the model is amplified. This reduction of clarity is displayed in the remainder of the two experiments in (Gao, 2018) where the models are adapted to incorporate multi-class classification problems of damage level and damage type.

The damage level experiment with the classes illustrated in Figure 31 is the perfect example to illustrate this point of class clarity reduction. As demonstrated in Figure 31 separation between the classes is indefinite e.g. if a minor damage fault is photographed too close to the fault it resembles moderate damage more than minor damage. The uncertainty surrounding the metrics of class in these two multi-class classification experiments are the reason for the degradation of the results, the damage level model only achieving a 77% testing accuracy while the damage type model performs even worse merely attaining 57.7% testing accuracy. This ambiguity of the classes makes it increasingly difficult for the classification task to be executed accurately due to the parallels the classes share this predictable causes complications in the training of the model thus reducing the accuracy of the classifier. This underlines the importance of transparency of classes in the creation of an efficient effective computer vision CNN model.

Figure 31-Sample images used in damage level evaluation. (Gao, 2018)

5.2 Proposed method

In an ideal scenario, there would be a database with clear images of cracks with a constant dimensional aspect ingrained within them to aid in the creation of a dimensional CNN crack classifier. However, for this research such data is unavailable and the time-consuming nature besides overall difficulties that collecting data with a clear dimensional frame necessitates this dimensional approach is unfeasible. This fundamentally demands that the directional approach be implemented with it representing the only useful feasible methodology for crack classification.

The implementation of this directional approach unavoidable removes the dimensional dependent classes of shrinkage and crescent simply leaving the objective direction classes of horizontal, vertical and diagonal. The classes now assured the execution type for the crack classification is still unidentified with a multitude of options across the computer vision spectrum potentially reasonable for successful implementation. Due to the success of the CNN in the crack detection task, the preferred solution was to train a secondary classification CNN. This resolution for classification also permits seamless integration between already proficient crack detection CNN and this secondary CNN. As the experimentation to discover the optimum model was already accomplished in Section 4.2 the structure of the novel classification model will reflect that of the optimum model Transfer model 4. The characteristic separating the two CNN's is engrained in the operational task of the respective models the classification model is executing a multi-class classification task, whereas the crack detection CNN is performing a binary task of crack/non-crack detection. Considering this categorical difference of operation task a secondary dataset is required to encapsulate each of the crack classes to assist in quality training. As this secondary task is still constructed in the domain crack analysis the optimized database of 6'600 crack images is used to create the needed secondary class dataset.

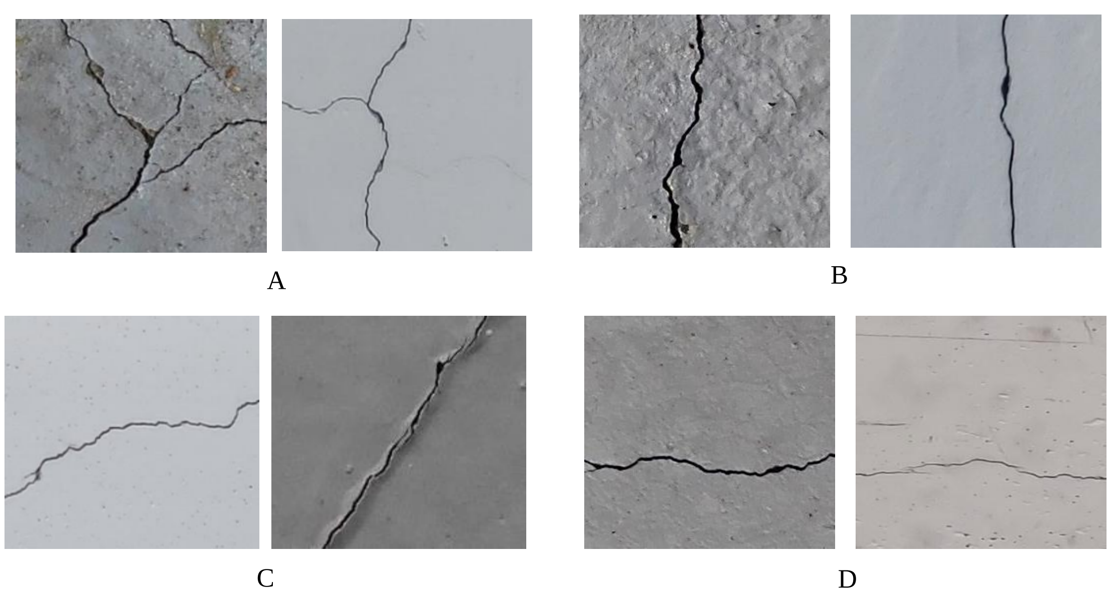


Figure 32 - Samples of the Crack classes. (A) Complex, (B) Vertical, (C) Diagonal and (D) Horizontal (Created by Author)

This database is created to summarize a directional classification approach using the three directional classes of horizontal, vertical and diagonal. However, these classes alone cannot truly represent all cracks with an additional class needed to embody complex cracks areas of the tunnel lining with examples of each class shown in Figure 32. The justification for the inclusion of complex cracks as a class is to prevent miss classification in problematic areas of images where the three directional classes cannot be definitively applied such as areas that contain multiply cracks, conjoining cracks and even cracks areas with unorthodox shape patterns.

As the database consists of images 256 x 256 pixels in resolution, only small differences separate a diagonal crack from both horizontal and vertical cracks with the small variation of the angle being the deciphering feature from the three directional classes as illustrated in Figure 32. This small variation in interpreting a cracks class inherently creates a level of subjectivity when choosing data for the classification dataset. Therefore, to diminish the potential influence of this subjectivity when creating the database if a crack is not obviously distinguishable to what class it belonged to it is not selected. This certainly reduced the number of images that we're qualified to be selected for use from the original database thus leaving only approximately 750 images for each class which were subsequently split up with 500 images used for training and 150 for validation.

By virtue of the directional nature of three of the classes, the volume of data augmentation that could be used was reduced slightly with no direct rotations able to be implemented without changing the class of the image. Therefore, only the utilization of horizontal and vertical flips could be implemented without affecting the directional classed images hence increasing the training data from 500 images to 1500 images per class.

The dataset now defined the training of the model using an unchanged training strategy from that outlined in Section 3.2 is performed achieving 92.6% training accuracy and a validation accuracy of 90.2%. The consequent testing of this secondary classification model is accomplished via incorporation of this secondary model into the scanning window crack detection CNN technique with the classification only adding one additional step to the process as depicted in Figure 33.

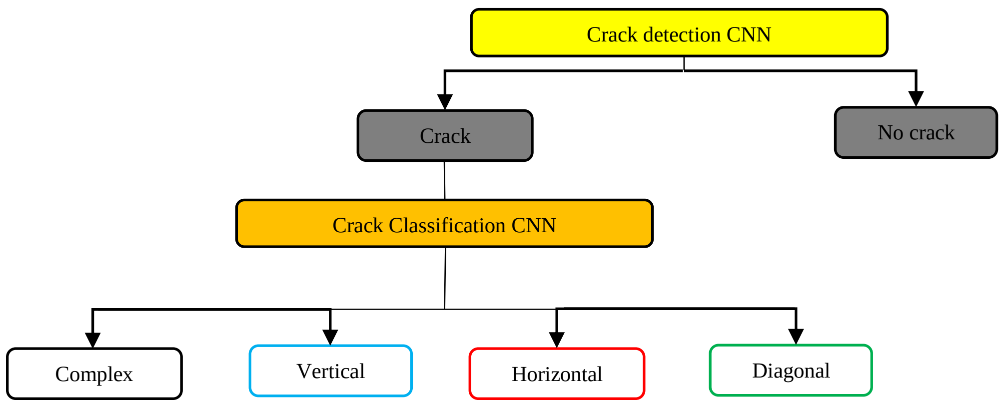


Figure 33- classification flow diagram

The amalgamation of the two models enables seamless testing of the classification model as it practices fundamentally an identical testing procedure to that of the crack detection the sliding window testing method described in 4.3. The dissimilarity separating the two methods is the use of classification borders which are illustrated via an output image comparison between Figure 29 & Figure 35. During the classification step, colour borders are utilized to decipher between the crack classes with the black border symbolizing complex cracks with red, green and blue representing the horizontal diagonal and vertical classes respectfully as illustrated in Figure 35.

This combination of two models demonstrates itself as a highly effective system nevertheless does have shortcomings rooted in its arrangement. The classification model owed to its location succeeding the detection model is profoundly reliant on the precision of the said detection algorithm through precision demonstrating how much of the detected cracks were in fact cracks. Regardless of the crack detection model achieving an extremely high precision of 93.9% with this not attaining 100%, it unavoidably influences the classification model by way of exposing the model to Non-crack images labelled as crack images. This is essentially compounding the error of the crack detection model into the crack classification process.

An option to counteract this problem is the inclusion of a Non-crack class into the crack classification model. This alternative is experimented with via training a five-class model (horizontal, vertical, diagonal, complex and non-crack). However, the inclusion of this additional class has a detrimental effect on the results reducing the training and validation accuracy by roughly 4% each. This reduction in effectiveness is hypothesized as the classification CNN not having a sufficient volume of non-crack data to gain a high enough precision without biasing the model towards this class.

As a result of the reduced performance after the inclusion of the no crack class the existing 4-class model was the best choice to move forward with thus allowing the testing of the classification model to commence. The identical sliding window testing method is implemented as before allowing the standard testing metrics to be calculated with the overall model achieving an Accuracy of 92.3% a Precision of 83.9% a Recall value of 82.3 % and an F1 score of 81.5% with a per class breakdown of the models test metric illustrated in Figure 34

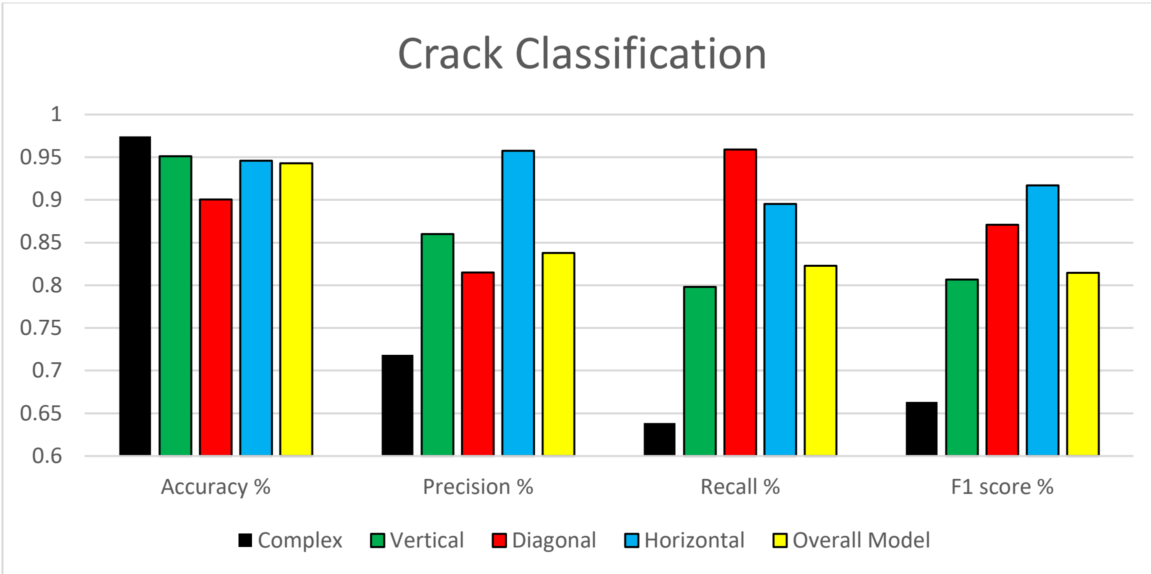


Figure 34-Crack classification testing metrics

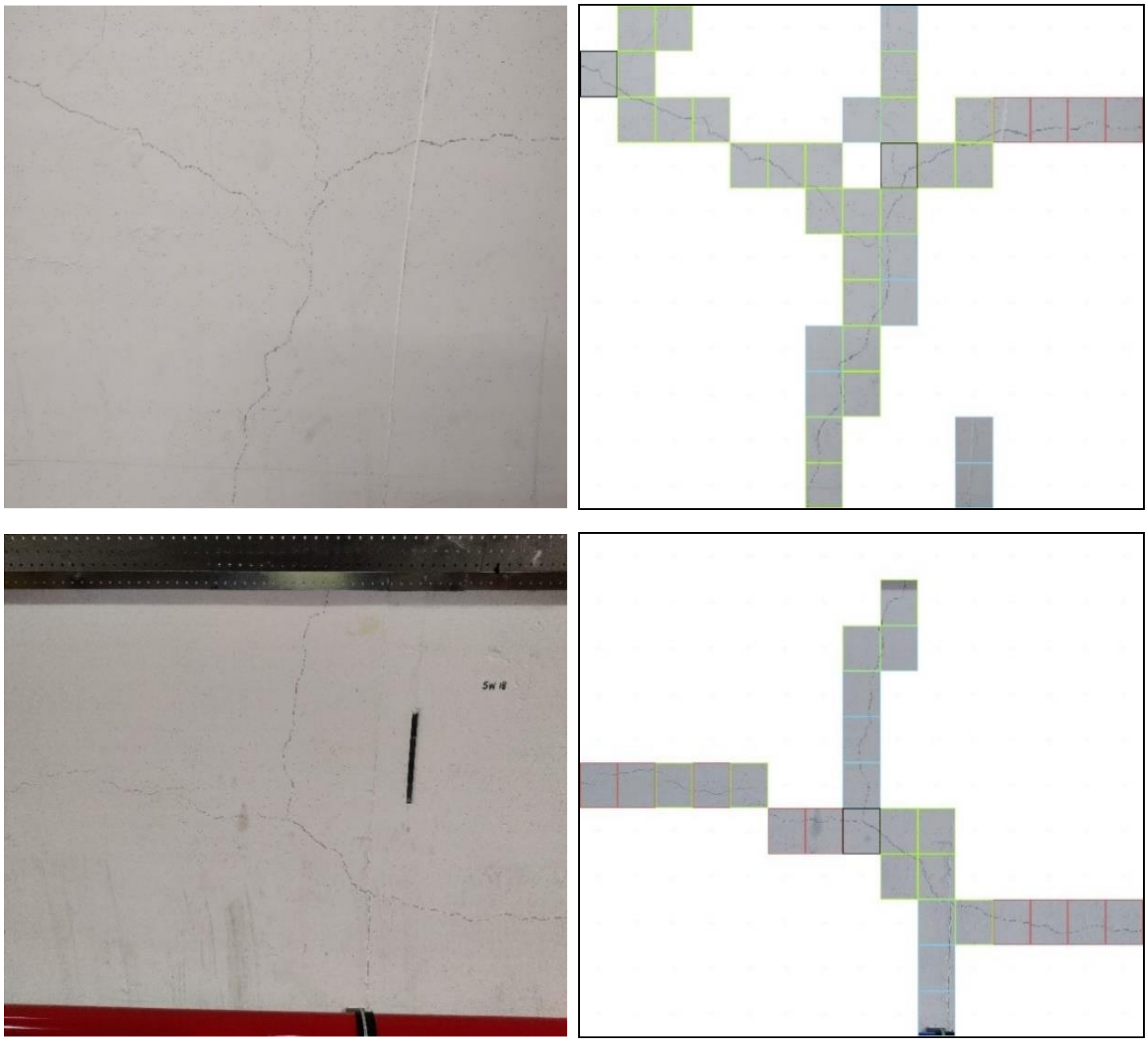


Figure 35- The images on the left are examples of test images that were input into the crack detection/classification CNNs, the images on the right represent the output from the classification of the image where black borders demonstrate complex to crack, red representing horizontal, blue representing vertical and finally greens signifying diagonal cracks.

CHAPTER 6

Field test

To maintain a safe and serviceable tunnel infrastructure as economically possible is a fundamental goal of the tunnel asset stakeholder. The objectives of preservation are conventionally divided into two separate categories of surveying and maintenance measures. The surveying aspect of preservation is the preliminary phase consisting of numerous different surveying techniques dependent on what structural feature is being sought after. In tunnel structures, the prominent surveying technique is visual inspections where the tunnel lining is assessed and categorized dependent on the surface structural health condition of the tunnel. To assist in continuous surveying, measurement control systems can be installed such as piezometers, manometers and even fibre optics to quantify a specific parameter. This monitoring is of huge importance as this data can serve as a hazard warning to facilitate the subsequent tunnel maintenance. Another benefit of the continual monitoring of metrics surrounding a tunnel structure is the collection of data that helps inspectors rationalize/contextualize the occurrence or development of the defects. These visual survey measures serve both as a preliminary archive of the condition and as continuous or intermittent documentation of the condition development. Due to highly frequent tunnel usage, the goal is to perform the inspections safely with as little possible inconvenience to the people or operational tasks. This traditionally means that the inspections are required to be carried out during non-peak / closure hours which is generally during the night which inevitably increases the cost of the inspections.

After a site inspection, there are three basic questions to be addressed:

• What is the condition of the structure?

• What types of damages are present?

• Is there a threat to structural and operational safety & serviceability?

In what manner these questions are answered essentially qualifies the inspector to decide on the tunnel's surface structural health. This classification generally indicates the type of maintenance that is required for a specific class. The maintenance management of the tunnel is essential with preventive, corrective or renewing actions are to be taken to prolong the lifespan of the tunnel structure. Such measures can start as soon as the tunnel construction as-built inspection is completed. The Periodical and systematic inspections assist in keeping the overall maintenance costs low. Usually, early identification of critical areas suggests the refurbishment work of said areas to be executed, while the damages are still small. Otherwise, if the interval between inspections is too large, the risk of a major incident increases thus leading to more extensive and expensive refurbishment works.

These safeguarding procedures of surveying and maintenance are incredibly important in each tunnel environment however these measures are heightened further for the LHC tunnel infrastructure at CERN due to the minimal movement allowed to ensure sustain alignment of magnets. This high degree of accuracy over tens of kilometres in a radioactive underground environment renders it an unprecedented challenging task to complete full inspections. Currently, to accomplish a full inspection of all the underground infrastructure, a Long Shutdown (LS) of two years such as LS1 and LS2 is required when the nuclear research operational tasks are limited or not permitted. The automation of these inspections as researched in this project aims at providing an automated system that is safer, faster and cheaper than the current method of inspection consequently permitting the periodic inspection to be performed more frequently. This atomization aids in the early detection of defects / critical areas which support the reduction in the maintenance cost and crucially minimize the time spent in these hazardous environments. This chapter will describe the collection and consequent creation of continuous tunnel section images as a ‘field test’. The results from this field test are consequently described and contextualized as regards their use and potential for application in forthcoming inspections.

6.1 Image acquisition

*6.1.1 **Existing CERNbot data acquisition system***

Previously, the Beams (BE) department at CERN has developed a system applying robotics to gather data from the tunnel infrastructure. This aforementioned research investigated data acquisition to aid in the inspection process with the techniques of ultrasonic, electrical, remote image capturing and strength-based exploration. The design of the data acquisition system is largely dependent on the constraints present in each specific scenario. The constraints of space, time and environment in combination with the required quality and accuracy ultimately shape the approach of data acquisition that is best equipped for the particular task. The methods of ultrasonic, electrical and strength-based techniques have great potentials in providing valuable and specific information on the tunnel structure. Although these methods can be utilised in the collection of tunnel data the acquisition techniques require close proximity to the surface for collection. This proximity prerequisite fundamentally causes such approaches to be unfeasible for implementation in a large-scale scenario such as the 27 km long LHC tunnel.

Figure 36 - illustration of the CERNbot mobile platform in addition to the depiction of the three-tier camera setup on the platform ((CERN, n.d.)

To be considered a feasible alternative for conventional inspections in such an environment as the one presented at CERN, the data acquisition methodology ought to be robust and adaptable enough to overcome the obstacles that are inherent to the tunnel environment. The aforementioned research implements a Remotely Operated Vehicle (ROV) resembling (Menendez, et al., 2018) developed in the BE department of CERN and is referred to as the CERNbot. This is a mobile platform device on which different devices, such as sensors, cameras and robotic arms can be placed to conduct different interventions. As this research is concentrated on conducting tunnel inspections, the platform is equipped with a metal structure on which multiple cameras are be placed to conduct image capturing the setup illustrated in

Figure 36. The construction of the three-tier collection system is to enable a larger higher quality image of the tunnel lining to be created in comparison to a one camera operational method.

These three cameras work in tandem to collect images with a sample of the collected images shown in

Figure 37. These separate images are subsequently stitched together to create one larger better representative image of the tunnel lining with the results of this process depicted in Figure 38.

Figure 37- A example of the collection of the image data from the CERNbot with the image on the left the image captured from the highest of the 3-camera setup and the image on the right indicating the image captured for the lowest camera. (CERN, n.d.)

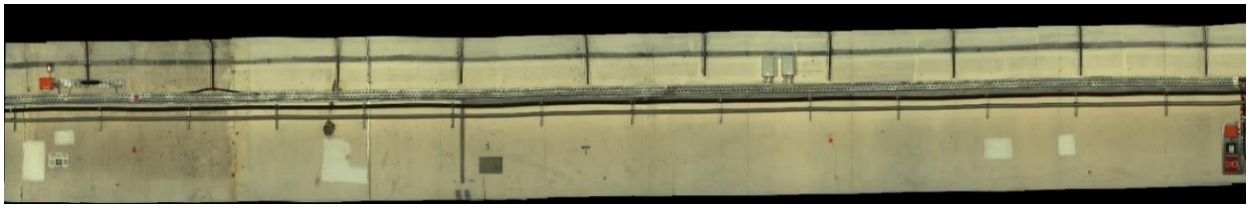
A functioning collection system now created the execution of the methodology of data acquisition was performed in controlled tunnel scenarios. The data acquisition was achieved by positioning the CERNbot approximately 1.5m from the wall during operation while driving the mobile platform at constant 0.2m/s along the tunnel section in one direction as the coordinated camera set captured images to create stitched images like that depicted in Figure 38. This effective collection system was implemented in three separate sections of the tunnel infrastructure at CERN approximately an 80-meter section of the LHC, roughly 50-meter and 20-meter sections of the TT1 tunnel infrastructure.

Figure 38- the subsequent stitched image created from the 3 images in Figure 37

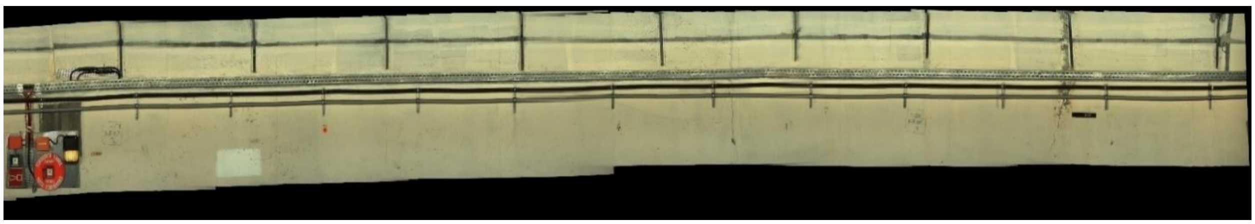
6.1.2 *Stitching long tunnel panorama*

This database of uninterrupted sections of tunnel linings formed the basis of the field test experiment. As the methodology of collection of the images demands the mobile platform move quite slow in the collection of the images to ensure the images are of sufficient quality i.e. adequate focus. The slow continuous movement along the tunnel fundamentally results in a proportional overlap from image to image. This commonality in images allows for the section of images collected to be stitched together to form a panorama type image of the overall section of the tunnel with the created profiles of CERNbot collected images illustrated in

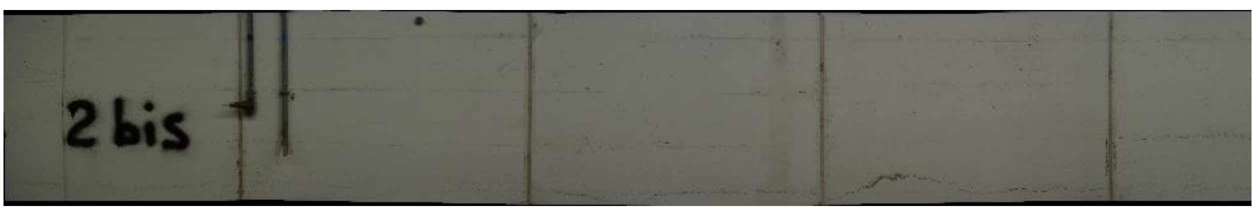
Figure 39 and Figure 40. The creation of these sectional panoramas is central in permitting the execution of a field test with the purpose of said test being twofold. Firstly, these sectional image profiles were created to enable the assessment of the developed algorithm speed, robustness and accuracy on a continuous section which indicates the feasibility prospect for future inspections. Secondly, these images facilitate in moving the algorithm from a controlled virtual environment of being subjected to specific crack area images to a more realistic depiction of a common section of a tunnel. This exposure to more realistic images helps the algorithm adapt to images where the crack percentage is quite low in comparison to the high crack percentage images on which the algorithm was developed and examined upon.



(A)



(B)



(A)

(B)

Figure 39- The stitched 50-meter section of the TT1 tunnel with (A) demonstrating roughly the first 25 meters and (B) indicating the remaining 25 meters (Created by Author)

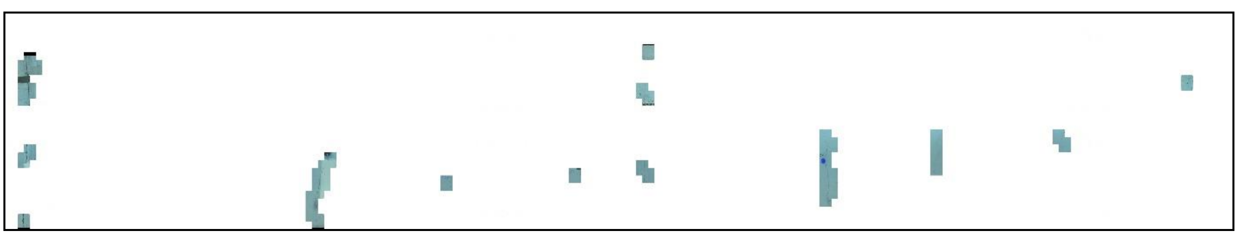
Figure 40 - the 20m stitched image created of the TT 1tunnel Created by Author)

6.2 Crack analysis of stitched long tunnel panoramas

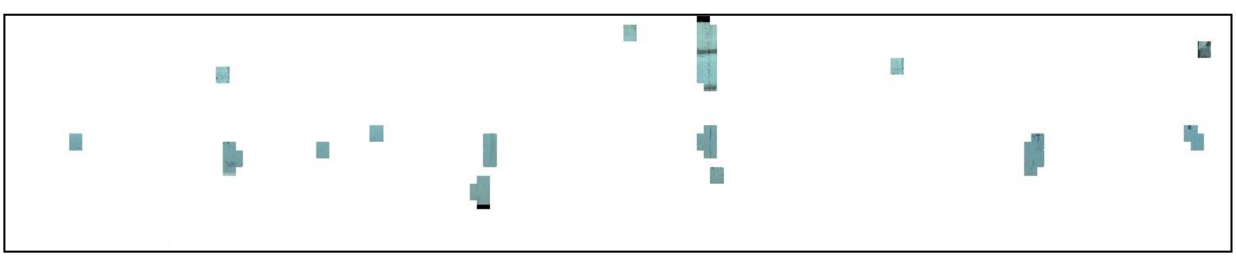
Based upon the stitched sectional panoramas, the field test is performed to present the tunnel profiles to the algorithm. The combined total length of the 3 images roughly equates to 150 meters. The profiles created only encompasses half of the tunnel lining due to the collection method not accumulating the full 360 degrees of the tunnel profile. The algorithm takes just 9 minutes and 45 seconds to execute the detection, classification and creation of crack profile visual data output. As the processing speed does not slow down with the content of the image rather the size it is fair to assume that if the full tunnel 360 degrees lining sectional panorama was collected the processing time would double. This assumption results in the processing time being computed as 6.8 seconds per meter of the tunnel. The extrapolation of this processing time to the 27km of the LHC equates to a full inspection time of just under 59 hours, which saves considerable time compared to conventional manual visual inspection.

6.2.1 ***Global crack percentage***

The resulting crack profiles created from performing the field test are portrayed in Figure 41, Figure 44, Figure 45 where a high degree of accuracy is displayed by the algorithm in detecting and tracing cracks along a continuous tunnel section. To complement the visual results created from the algorithm, numerical results relating to the visual data are likewise produced with an example of such results illustrated in the results corresponding to Figure 41. The numerical result in combination with the corresponding visual output can be utilised as a useful tool by the inspector to gain a comprehensive understanding of large-scale tunnel behaviours and classify them accordingly. The results produced can be processed under specific criteria to attain the information the inspector is seeking extract whether it be an overall crack distribution pattern, prominent crack type or an additional refinement of the data into sectional information. Each of these individual numerical values concerning the tunnel cracking can be utilised differently to either classify the tunnel or to gain a deeper understanding of tunnel performance embedded in the surrounding varying rock layers. The overall crack percentage output as illustrated in Table 2 delivers a general view of the surface structural health of the tunnel structure which could be subsequently used to classify the severity of the damage. Once the algorithm is exposed to a more continuous tunnel section data, thresholds following CERN's five classes of severity can be created.



(A)



(B)

Figure 41- The resultant output image once

Figure 39 has passed through the crack detection algorithm.

The secondary output of crack classification can also be applied to gain a higher understanding of the behaviours/ movements that are occurring in the structure. This interpretation of the mannerisms of the tunnel is gained via the direction of the crack class indicating the potential formation rational. Each of the three directional cracks classed signifies movements in the tunnel with vertical cracking being equated to a compression strain on the sides of the tunnel, horizontal relating to compression on the crown of the structure and a diagonal crack signifying the structure is experiencing some torsion force. The creating and inclusion of the complex crack does not directly indicate any prominent behaviour however still does give useful information. The inclusion of such a class denotes unorthodox behaviour occurring or a combination of multiple movements causing multiple cracks to form or the joining of two separate cracks.

Table 2 - The numerical breakdown of the results output from

	First Pass		Second Pass	
	Crack windows detected in the first pass	Class percentage %	Crack windows detected in the second pass	Class percentage %
Complex	0	0	1	0.04
Diagonal	22	0.82	23	0.93
Horizontal	0	0	0	0
Vertical	12	0.45	13	0.52
non crack	2654	98.73	2446	98.51
Total % of crack	1.264880952		1.490132904	
Overall % of crack	1.377506928			

If we take Figure 41 as an example, the image encompasses roughly 80m of a tunnel with extraordinarily little and very dispersed cracking with an overall crack percentage of 1.38%. This value of overall crack percentage equates to the equivalent crack window area in relation to the overall area of the image essentially 1.38% of the image area contains a crack. This value can fundamentally be used as a simple metric by an inspector to make a judgement on the surface structural health of the tunnel. The value whether it be large or small still provides useful information the lack of cracks enables an inspector to assume that no prominent external forces cause cracking in this section, while the converse applies if a large crack percentage is determined. In comparison to Figure 41 the 20m section of TT1 depicted by Figure 45, the crack detected area is higher with a 3.2% crack percentage with diagonal cracks being the prominent kind classified, hence implying that this section of the tunnel is undergoing a torsion force.

6.2.2 ***Sectional crack percentage***

The overall results output from the tunnels concerning the total crack percentage and prominent crack type are particularly useful, although can be somewhat misleading or misrepresentative. As the aforementioned outputs are relative to the overall tunnel section the global result can be disingenuous for example if this method was applied over a large tunnel distance the resultant crack percentage will generally be quite low. The use of just global value as the diagnostic metric when utilized over larger distances is flawed as a substantial risk extremely cracked section have the potential to be overlooked. This outlier section will not impact the overall crack percentage value enough to gain the attention of an inspector if the remainder of the tunnel has a relatively low crack percentage. To prevent this oversight while enriching the potential for understanding the tunnel movements, the numerical information is distilled down further into desired chainages. This supplementary sectional perspective on how the tunnel is performing over the set chainages is gained via crack percentage and crack formation type.

The sectional summary of the crack percentage serves to highlight problematic chainages of the tunnel that require a second opinion by a trained inspector to quantify/contextualize the highlighted area. As the tunnel profile panoramas are relatively small to best demonstrate the benefit of sectional segmentation of the results the chainages are set at 1-meter intervals. In practice, these 1-meter chainages are too small for implementation in longer tunnel infrastructure in which more suitable values would be implemented to segment the relevant tunnel information.

The numerical sectional breakdown is demonstrated in Figure 42 where the crack percentage per meter of the Figure 44 tunnel section is plotted. These results are to illustrate the benefit of sectional breakdown by highlighting a problematic section in 32nd-meter chainage which contains 52% cracks. This value is extremely high and may seem worrying. However, when a more common chainage of 5 meters is implemented for this tunnel section, this exceptionally high value is less alarming as it is reduced to 13.0% as shown in

Figure 43. This in-depth data analysis creates ease of work for the inspector as the tunnel is inspected via the algorithm and the problematic areas are flagged/ highlighted by the automated inspection. These areas can be seamlessly observed on the visual data to permit the inspector to concentrate on these areas of the data for a more nuanced opinion on whether a personal inspection or maintenance works are required.

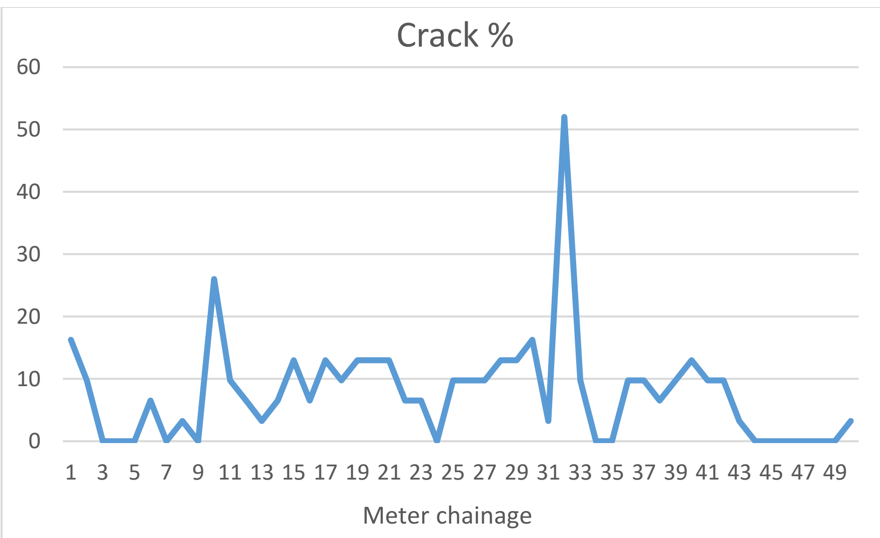


Figure 42- Figure 44 crack percentage broken down into 1m chainages.

Further to the global crack percentage, the creation of the sectional data of crack classification has greater potential for providing more insightful information in demonstrating the change in movement along a full tunnel section. These results will be especially beneficial for tunnels that are subjected to sectional change regarding the drainage/ geological conditions such as the injection tunnels used at CERN. The deeper understanding provided by refining the data into chainages permits a greater understanding of the CERN tunnels thus allowing for more informed decisions as to what can be done to reduce further and future damage to the structure to incense the serviceability lifespan.

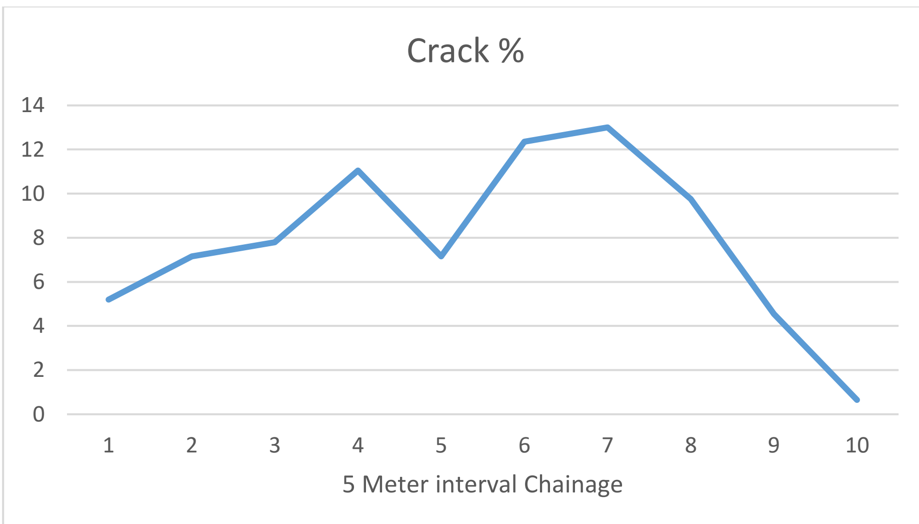
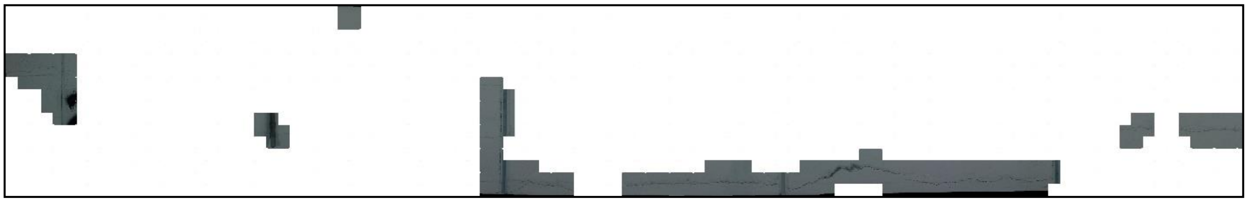
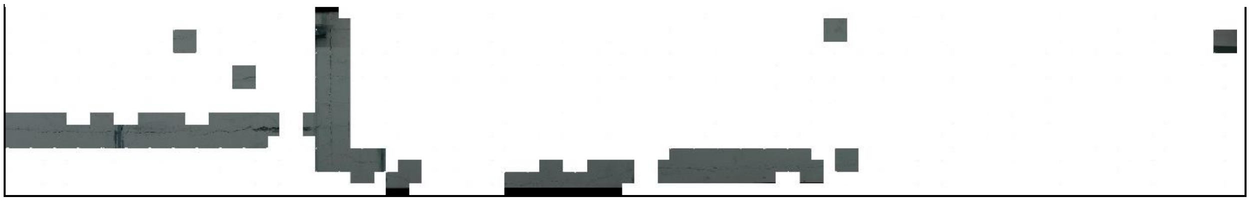


Figure 43-Figure 44 crack percentage broken down into 5m chainages



(A)



(B)

Figure 44-The consequent image created after Figure 39 has experienced the crack detection algorithm.

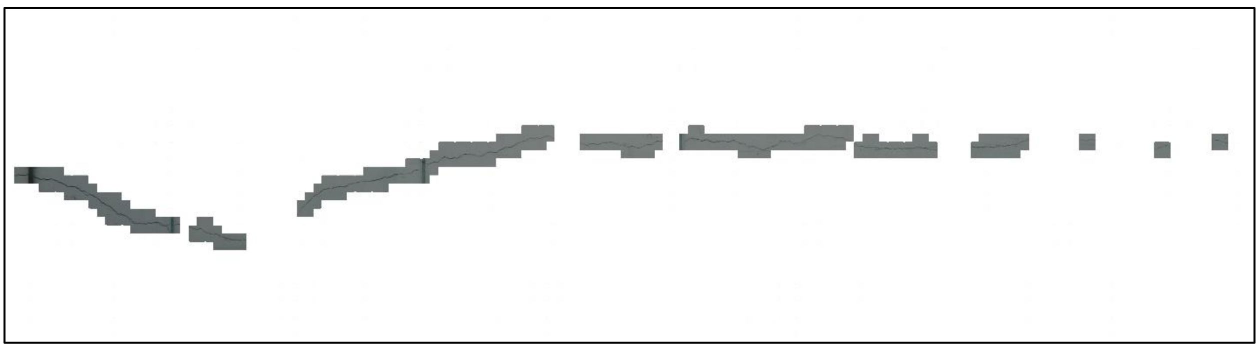


Figure 45- the output image after Figure 40 has been passed through the developed crack detection algorithm.

6.3 Drone flight

As outlined in Section 6.1 the image acquisition of the tunnel lining in the sections of the SPS, LHC and TT tunnels is quite technologically advanced and specialized for particular sections. The development of the technology to prevail over any tunnel portion circumstances is quite labour intensive due to the space limitations, available time and other environmental conditions that CERN tunnels provide. Coinciding with these criteria of developing the standards of inspection in regard to the choice of the sensors, the data that is required; its quality, resolution and accuracy all compile in the difficulty of deployment of the CERNbot across the tunnel infrastructure.

The subsurface infrastructure at CERN does not strictly comprise just of tunnels with access & service shafts making up an integral role in the continued operation of the infrastructure. These shafts provide key access to supplies, equipment and crucial personnel that are integral in the operation and elongation of the functionality of experiments and especially the tunnel structure themselves. Currently, these essential structures are inspected via visual in-person inspections. However, these inspections are incredibly difficult due to the infrastructure nature of the shafts being quite large and, in some instances, going to depths of over 100 meters and 20m in diameter. Alongside their size, the access to some of the shafts is quite limited due to the operational works going on within them such as personal access lifts and material access crane lifts. In these shaft cases, the limited access combined with the shaft lining often obstructed ultimately results in limited inspections area and therefore more assumptions being made on the structural capabilities. Corresponding to these operational shafts are dormant shaft which pose different but equally as problematic inspection problems . Given that these shafts were predominantly used during the construction phase only the access from the surface to these tunnels were merely temporary and upon construction the tunnels capped, and access removed. This has led to this share of the shafts inspections to be carried out from the ground to as much of internal lining as can be seen. These procedures currently are the only feasibly way to carry out such inspections and needless to say is unsatisfactory.

The inspection of the shafts at CERN are incredibly important as their vertical configuration of the structure is exposed to numerous varying loading scenarios across its entirety. These circumstances are a resultant of the variation of strata the shaft has been constructed through blended with overlapping and varying water table ultimately altering the structural loading boundary conditions. If inspected correctly the collected data could be incredibly insightful to the structural capabilities of the tunnel structure under the varying boundary conditions as a correlation between degradation and strata/water table could be found. This insight will provide incredibly useful information that can be used by CERN for the future developments/ expansion of the LHC to make structural alterations to reduce the impact such factors could have on the construct.

The development of technology comparable to that of the CERNbot to operate these shaft inspections is beyond the bounds of this research. However, a feasibility inspection was undertaken to examine the capabilities of the developed algorithm on shaft sections. The shaft lining in CERN can be separated into two distinct groups coarse finish of shotcrete and fine paint finish tunnel lining. The coarseness of the shotcrete is an entirely different finish to that of the images that this algorithm has been trained upon. Shafts that utilized shotcrete as the internal tunnel lining are generally inactive/ capped shafts resulting in a reduction in the inspection importance. Due to the juxtaposition of the finish of the shotcrete to that of the finish of the images used in the training data, it was decided to perform the viability flight of the drone in a finely finished shaft to give the best representation for implementation for future inspection

This inspection was carried out in Point 8 a shaft located just off the LHC as shown in Figure 46. This shaft was chosen for the feasibility flight due to the resemblance of the shaft lining to that of the tunnel lining in combination with the ease of availability to halt shaft operations during the flight.

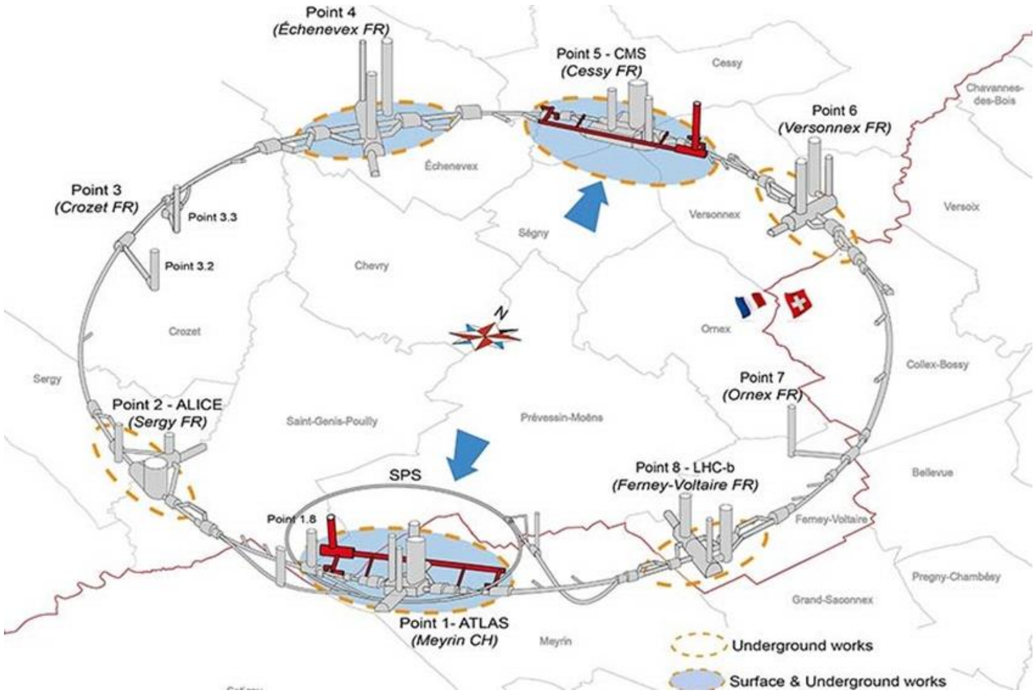


Figure 46 - LHC infrastructure schematic (CERN, n.d.)

Figure 47 - Images of the Shaft at Point 8 (Images Captured By Author)

6.3.1 Flight Procedure

The operational procedure of the inspection of the shaft can be basically separated into two categories flight preparations and flight operation. The preparation for the flight is straightforward and revolves around the calibration of the internal compass of the drone. This calibration is especially important while operation in the shaft environment as the done is fully reliant on this calibration due to the lack of any GPS Signal to help. The remainder of the preparation operates on setting up a safe environment to perform the flight. The secondary category of the operational procedure relating to the flight procedure is broken down into the following steps:

1. Initial position of drone is 2.5 m from the wall, on the ground.
2. Turn on the spotlight on the controller and adjust the luminosity
3. Manually adjust the distance to wall and luminosity of spotlight to get satisfying images quality.
4. The operator stands close to the tunnel wall, facing the tail of the drone. Fly up and down using T mode. While going up, take images every 1/3 of the screen height (estimated as every 1m).
5. Fly the drone vertically up to a height of up to 7.5 m (which is equal to the maximum horizontal distance between operator and drone) and fly back down to the original position.
6. After landing, move drone to next point (estimated as 0.5 m along the perimeter) on the internal black circle, making the 2/3 of screen width is overlapped horizontally with previous images. The operators move to the new position to see Figure 48.
7. Repeat steps 1-6 until the tunnel shaft perimeter is inspected completely.

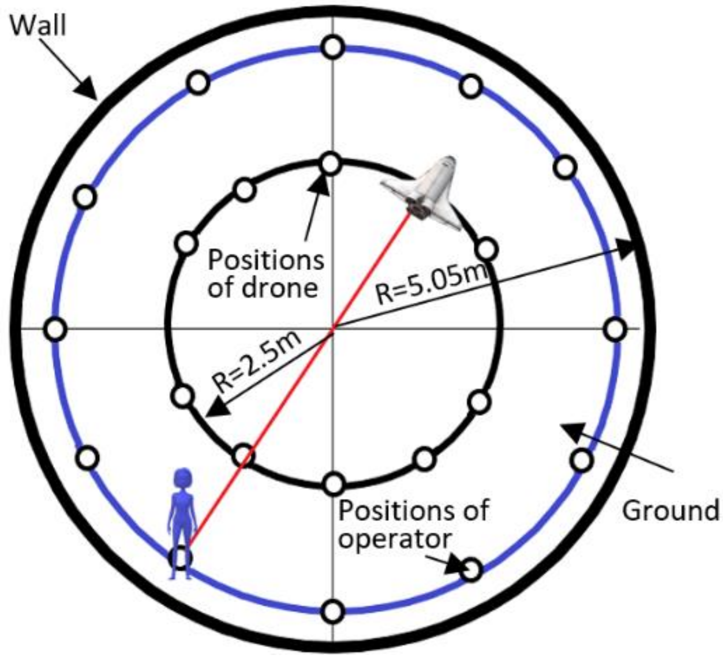


Figure 48-Flight operational positions Created by Author)

6.3.2 Flight Results

Following the operational flight, the data was downloaded onto a hard drive of the computer. This flight data was subsequently divided into each flight path to enable sectional panoramas to be produced utilising the same methodology to that used in Section 6.1.2. Once all the sectonal flight paths had been seperated and subsequintly stiched togeteher there were some repetitive inconsitencys noticed in the created pannoromas. As highlighted in Section 6.1.2 the slow continuous movement essentially creates overlap from image to image to enable the stitching to occur. After analysing the results from the drone flight an inconsistent sectional width/path of the flight is illustrated in the created panorama’s. These discrepancies are demonstrated in the three flight paths in Figure 49.

As with the nature of these shafts there are verry little object or distinguishing features this lack of features. This lack of features has a negative influence in the creation of the Flight Path sectional Panoramas as the stitching software has little to no feature in the overlapping images to definitively say where the two independent images intersect. Coinciding with the lack of characteristic features in the shafts is the drone flight was performed manually using judgement of the operator. During the experiment the operator had to decipher when to stop the upward trajectory of the aiming for 1/3 overlap with that of the previous image. The consistency of the commonality was deemed adequate during flight operation but unfortunately non comparable to the programmed operational speed CERNbot to facilitate sufficient overlap. The last significant factor that effected the results was the operational wind that was swirling in the shaft. On the ground during the experiment of the shaft the feeling of wind was inconsequential. However during the flight it was apparent the drone was experiencing turbulence and struggling to keep the distance from the wall consistent. As the operator was located directly beneath/behind the drone during the flight duration the vertical flight path was more constant but not flawless.

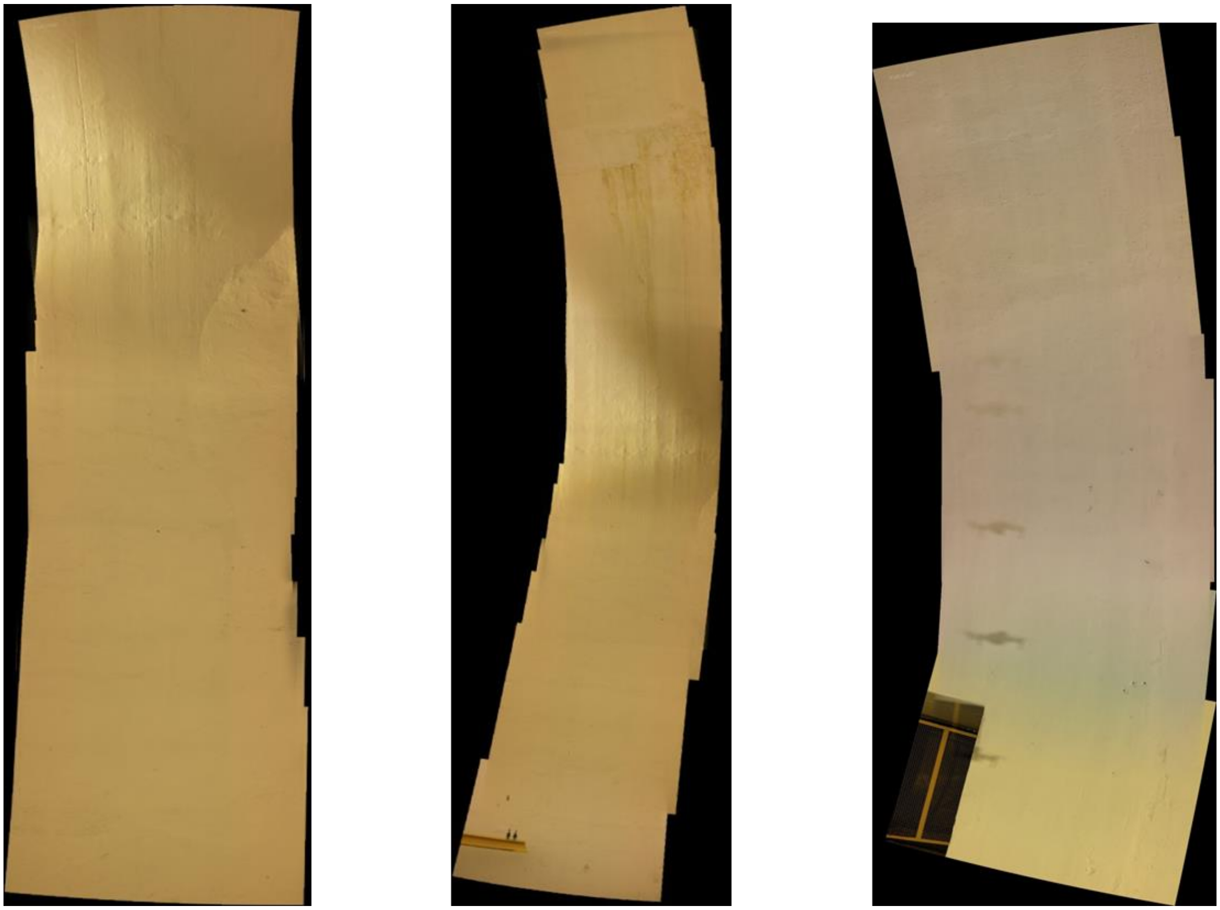


Figure 49 - Three individual Drone Flight paths Panoramas

6.3.2.a Envisaged improvements

The above-described complications combined were fundamentally unfavourable in the formation of fully Coherent sections. However, as the purpose of the flight was to deterimine the feasibility of implementation of such tecnology in inspections. This intial test was extreamly limited due to time and tecnology avilable yet provided some promising result with some specific adjustments implemetaion the above discovered limitations could be limited and even nullified.

The implementaion of a sensors to monitor the vertical flight distance combined with the horizontal movement of the drone during the flight would hugely benefit such inspections. This data fusion with the already incredibly advanced tecnolgy that the drone provides would infinetly enhance the results . The integration of such geo location properites with the image data has offers huge benifitial impacts to results. Firstly the ability of accuratly know the flight path and where alterations to the data is needed to facilitates articulate accurate panorama be created. These impoved vertical sections could be subseqentially stiched togeter and plotted 3 dimisonally provding a more enhaced representaion of the tunnel which over its lifetime can be tracked providing a degredation profile of each shaft. . The geo refernce ability of a image could become a incredibly powerful tool from something as simple as allowing for faster repairs to something as complex as creating and tracking degredation profile of each shaft using these created 3D image models. These degredation profile’s would be constucted through the monitoring of the development of cracks ,water ingress or any other stuctural defect from inspection to inspection. The intergrated location also enables corilations of defect type , defect acceleration etc. to the surronding geological envroment. This could be ehanced yet again with the inclusion of the infared capabilitys of multiply data collective drone such as the one used in this test providing a additional layer of data to gain a better understanging to deteration rational.

Such develoment to the already advanced Drone Data would provide incredibly usefull information to how shafts such as these degredate over time . This information is invaluable to CERN who plan on expanding the LHC and knowing how a structural element degenerated over time and how design against it could hugely enhace the lifespan of the infrustucture .

CHAPTER 7

Conclusion

The computational processes of machine learning and deep learning emerging importance over the last two decades is due to the accumulation of vast amounts of data spanning every facet of life granting the practices to gain traction in the mainstream. The expanding capabilities of technologies in conjunction with the increased speeds of the computational concepts which once were thought to be unfeasible for implementation in preserved niche projects due to their expense are now being researched and executed as viable options. The Civil engineering spectrum is not exempt from the influence of such technologies with one of the standouts developments the advances made in surface structural health monitoring. The use of computer-driven vision-based defect detection is commonly performed using image processing techniques (IPT's) or convolution neural networks (CNN's) which are the leading artificial intelligence (AI) used for surface structural health monitoring. The surface structural health image-based technology monitoring development serge has been predominately developed with implementation on above ground civil structures with the underground infrastructure dwindling behind. One of the justifications behind this deficiency of the underground image-based technology improvement in contrast to its other civil structural counterparts is associated with the difficulties that this underground tunnel environment poses for image acquisition. The lighting or lack thereof lighting, access and the obstructions common to tunnel linings create a challenging environment to collect quality image data. This lack of quality data has a consequential effect of increasing the difficulty in creating a vision-based defect detection in the civil environment. Of the previous systems created the concentration has primarily been on defect detection, the innovation of this research extends the detection a further step by performing additional classification on the detected cracks based on directional criteria.

7.1 General discussion

Traditional machine learning (ML) and Deep Learning (DL) frameworks present considerable differences even though the mathematics applied in the construction of the algorithms are almost identical. The ways of handling problems with these two approaches necessitate different strategies due to the nature of data provided to each framework. Traditional ML algorithms require the explicit definition of the features deemed relevant for the task in contrast the appropriate feature extraction is conducted within the DL algorithm framework. The performance of feature extraction internally by a DL algorithm facilitates straightforward construction and development on readily available and relevant raw data. In contrast, the time-consuming manual feature extraction and precise feature definition are necessary before the commencing of training for the ML equivalent algorithm. The capability to operate directly on raw data without explicating definition of the features that hold significant benefits especially in problems that hold a level of complexity and require subjectivity for interpretation. In chapter 2 a comprehensive literature review is conducted for crack detection on both DL and ML processes. As studies of crack detection in tunnels using this technology are quite limited the literature review is expanded to incorporate other civil infrastructure utilizing this expertise for structural assessment.

As a result of this review, it is observed that the performance and popularity of Convolution Neural Networks (CNN) applications surpass those of the traditional ML algorithms. The increased popularity of these DL CNN algorithms is due to the capabilities of CNNs in processing multidimensional raw data while conserving the spatial relations which is crucial when employing visual data. The direct availability of highly sophisticated and accurate CNNs models e.g. VGG16and ResNet for redeployment in a secondary image classification assignment such as crack detection improves the feasibility of CNNs for distinctive tasks. The accessibility to such powerful algorithms for redeployment is due to a phenomenon known as transfer learning which hugely reduces the traditional barriers of DL algorithm development hence potentially saving much time required for image data acquisition by traditional inspections. The benefit transfer learning provides is the exponential reduction in the volume of data required to sufficiently train a Deep learning CNN model which was conventionally the obstacle for niche tasks such as crack detection.

The task of crack detection is objectively a straightforward binary problem task, which classifies images to whether they contain a crack or not. However, this task when implemented into a DL model is not quite as simple owing to the limitation on discernable features a crack holds to decipher it from crack like non-crack objects common in civil structures such as shadows, irregularity in surface texture etc. This research creates a Deep learning CNN application algorithm of sufficient precision to enable subjective structural heath monitoring inspection of the tunnel infrastructure at CERN. The physical characteristics of the tunnel infrastructure provide a quite distinctive and challenging environment for the collection of excellent quality image data. The lighting conditions and obstructions on the tunnel lining are the more noticeable obstacles hindering the collection of the desired quality image data. To encompass the intricacies of the particular environment posed by tunnels at CERN a specific database is crafted to resemble the task such that a crack detection algorithm can learn complexities and attain the appropriate precision for the task. The desired information sought from this dataset is to define the characteristics that quantify a crack specifically in the CERN infrastructure. For a computer vision application such as the one developed in this research to be effectively applied a well-defined goal, dataset and structure are essential for successful execution. In chapter 3 all the necessary steps taken in the creation of the substantial crack dataset are demonstrated. As the detection of cracks in images is the initial goal of the algorithm the remainder of chapter 3 describes the training procedure and the CNN's structural steps taken to solidify the computer vision goal. The precautions taken within the CNN structure to facilitate improved training accuracy and speed while reducing the possibility of overfitting are described and rationalized.

Chapter 4 exemplifies the Data and the Architecture optimization owing to the influence they hold concerning improving algorithm applicability. The optimization possibilities for CNN's are limitless however the features of complexity and volume are the parameters with the utmost significance and hence hold the most possible influence to increase efficiency. The objective of Database optimization is to increase the accuracy, efficiency and applicability of the model. These aims can be achieved through applying a specific or combination of multiple methodologies to ultimately reduce the complexity of the process within the model. The influence the data holds over model development should not be underestimated, as the data is the singular most dominant factor in determining the performance of a model in any task. The creation of a dataset is fundamentally the easiest part as all that is required is the collection of the data. The creation of a quality dataset however is a more challenging prospect in essence the intention is to strike to correct balance between the quantity and quality of the data. This optimization of a database is a case-specific task as the database must incorporate the necessary complexities of the specific tasks without preserving unnecessary lower quality data. The evaluation of the database quality is simply performed by preliminary training models on the data and cataloguing the misclassified data. This documenting of misclassifications allows for database refinement by illustrating the recurring themes triggering misclassification. In this study, cracks on the extremities and the original database insufficient incorporation of common CERN tunnel obstructions were the underlying topics of misclassification.

The operation of these refinement practices results in the database becoming optimized to the necessary volume while containing the necessary complexity of the CERN tunnel scenario. Once the data was optimally refined the model architecture optimization could commence. Similar to the data optimization the complexity and volume were the parameters emphasized most in architectural enhancement. However, in the case of the model enhancement, it was the volume of trainable parameters and complexity of training techniques that were altered during the experimental procedures. In chapter 4 the various experimentation models results are outlined where it is observed that the pretraining network leveraging the transfer learning fine-tuning method converge rapidly and obtain the highest results of 98.7%, 98.3 % for training and validation accuracy, respectively.

These experimentation models were tested further via exposure to 30 larger specific cracked areas of the CERN tunnel infrastructure to genuinely assess the robustness and applicability of the algorithms for deployment. The fine-tuning transfer learning model once again demonstrates to be the optimal model by achieving the highest F1 score (93.5%) as well as showing no degradation of accuracy when faced with new images by attaining a testing accuracy of 98.4% only slightly below that of the training accuracy. This shows the model is accustomed to the intricacy that is inherent to a tunnel environment especially the intricacies of the CERN tunnels.

Following the detection of a defect, the subsequent classification of the degradation is crucial in the analysis of the surface structural health condition of the structure. The style of classification utilized in the assessment of a structure is conditional on the preferred outcome of the inspection with the classification possibilities of severity, orientation or dimension with each option giving a distinct evaluation perception of the tunnel surface structural health/behaviors Historically crack detected via visual inspection are classified by applying the inspector's vast experience which is an inherently flawed and subjective strategy of assessment.

The originality of this research is attributed to the introduction of an objective directional crack classification method in conjunction with a crack detection algorithm both of which utilize transfer learning. The proposed novel approach of connecting two separate algorithms to generate a clear, objective, precise, and robust crack detection and classification strategy. Chapter 5 demonstrates the rationale for the data design process needed for the creation of the second directional crack classification model. In addition to the data and training alterations, the testing of the integrated detection/classification model is conducted under identical conditions to the singular detection model. The integrated approach accomplishes good testing metrics of 92.3% for Accuracy, a Precision of 83.9%, a Recall value of 82.3 % and an F1 score of 81.5%. The degradation of the results can be attributed to the extension of the performance task from a binary to a multiclassification task which inevitably increases the opportunity for misclassification. The use of the distinctive integrated two model crack detection and classification permits an unbiased evaluation of a tunnel's surface structural health as well as allowing further insight into the structural actions of the tunnel lining. This further insight is gained by using an idealistic theoretical approach that the formation of each crack mainly develops due to geomechanical deformation subject to external forces on the tunnel structure.

In reality, the formation of cracks is not as black and white as this with the formation of the cracks being a lot more nuanced. Thanks to the functionality nature of tunnel structures cracks can also be formed by humans interacting with the tunnels via the installation of objects into the lining, collision with tunnel wall in transport tunnel and geochemical process in the long term among a whole host of outer possibilities. This nuanced understanding of the tunnel can only be provided by informed personnel who understands the structure. This underlines the current limitations of a computational algorithm approach as it lacks this deeper context of the structure that cannot be provided by just visual data but can easily be provided by a knowledgeable inspector. The benefits of automated inspection systems such as this provides are enormous by reducing inspection time, subjectivity, human exposure time in the hazardous environment while increasing crack detection accuracy the limitations currently all relate to context. The development aims of tools such as this are not to eradicate the requirement for inspectors but to increase the speed of inspections while reducing the workload of inspections. Tools such as this move the inspections from the physical in-person examination of the tunnels toward the contextualization of the data provided by the atomisation crack detection inspection.

The development and testing of the detection and classification interrogated application is performed in an idealized controlled laboratory-like environment with the exposure of the algorithm exclusive to problematic densely cracked visual data. To transition the research, from the experimental setting into a realistic tunnel environment a field test is performed on three separate tunnel sections with an accumulated length of 150 meters. These continuous tunnel sections resembling panoramas of the tunnel lining exposed the algorithm to a realistic tunnel environment where the crack density is considerably lower than the images used privately to test the crack detection and classification model. The execution of the field tests over a set length allows a per meter processing speed to be calculated and extrapolated over the entirety of the LHC producing an inspection time of 59 hours. These calculations were performed based on the processing speed of the google collaboration GPU's which if desired a more powerful system could be utilized to further reduce the inspection time. This computerization of inspections such as this has the additional benefits of processing speeds continuously increasing that inspection time will fall further and further in the future while the human equivalent does not have this possibility. The time-consuming nature of the data collection for processing is the limiting factor of automating these visual inspections.

In summary, the main conclusions of this two-year research master study are given as below:

1. A Crack detection algorithm utilizing transfer learning leveraging the fine-tuning approach provided a CNN algorithm with accuracy results of 98.7%, 98.3 % for training and validation respectively while attaining an F1 score of 93.5% during training.

2. The prior defined optimal fine-tuning approach was altered to facilitate the development of a Novel crack directional classification algorithm comprising of four classes of horizontal, vertical, diagonal and complex cracks. This produced CNN attains testing scores of 92.3% for Accuracy a Precision of 83.9% a Recall value of 82.3 % and an F1 score of 81.5%.

3. The directional information provided by the designed approach when employed on a continuous tunnel section enables underlying mechanical movements to be established. These insights are gained by correlating the directional formation patterns of the determined cracks to external forces distorting the tunnel lings.

4. The developed integrated detection and classification algorithm when applied on a real-life tunnel continuous section performed consistently and compares well to the traditional visual inspection method in regard to inspection accuracy and speed. The established technique

detection/classification provides an objective and numerical evaluation of the tunnel's surface structural health.

7.2 Recommendations for Future Works

In comparison to conventional manual inspection, automated visual inspection has been increasingly developed and popular in academia and industry. This shift away from the conventional methods should not only be attributed to the diminished financial burden or reduced health risk but primarily motivated by the increased accuracy and decrease in subjectivity. Even with this research making huge strides toward the computerization of inspections at CERN, the research is still merely in its infancy in comparison to the desired automation. There remain significant further works before the algorithm is to an adequate point of accuracy and is reliable enough to complete an autonomous inspection of the CERN infrastructure. The three areas that demonstrate the maximum potential for progress is based on the enhancement of collection, classification and contextualization of the data. These three components are integral to the development of the automation algorithm, but the Collection is the area of most significance as the latter two of these features are dependent on the collection of additional and improved data. This in essence is due to the fact Deep learning algorithms are reliant on data for development but can only develop to the level of the data to which it is subjected. As detailed in chapter 6 there is a more than adequate data acquisition system at CERN to collect more data. The collection of more data is desired however the acquisition of more applicable data in which a dimensional reference remains the aspect of collection in which most improvement can be made.

There are many ways in which this dimensional reference can correlate to the image the two possibilities that show the most promise is the superimposition or integration of the dimensional into the data. The superimposition method theorized would use a set of lasers set up in a small square to project a designed established area onto the surface that is being captured. This would provide a reference area in the image on which a CNN could be trained using to dimeson any defects found in the image. As this is a 3-dimensional problem one of the lasers utilized would be required to have the capabilities to measure and store distance to the tunnel surface which will support an even more accurate defect classification. The second alternative is the integration of the dimension into the capturing process itself by using an RGB-D Sensors. This is a specific type of depth-sensing device that works in association with a conventional RGB sensor camera. This sensor can augment the conventional image with depth information on a per-pixel basis. This augmented information can be used in combination with the focal length of the camera to calculate the relative field of view on a per-pixel basis.

The use of either of these methods of data acquisition would provide the necessary information to develop the second of the aforementioned 3 aspects for improving algorithm Classification. At present, the algorithm is implementing a four-class directional classification approach. The availability of data with a dimensional frame of reference would enable the research to move towards a more conveniently dimensional crack classification methodology. The progress of the classification in this manner will not alone move it closer to the current classification procedure but also create a more concrete and accurate classification algorithm. The integration of dimension into the classification procedure empowers the algorithm to make more objective decisions based on crack width and length in comparison to the more subjective directional approach. The development of the classification procedure has the additional benefit of making it a more accurate and more realistic depiction than the current crack detection/ classification procedure. The benefit gained from such a development is mainly attributed to the fact that the smaller shrinkage crack or artificial cracks formed from human interaction with the tunnel lining will still be detected by the initial crack detection CNN but will now be classified as a monitor structural defect by the secondary CNN. This will be a huge advantage of the dimensional approach as the severity of the crack is better represented in comparison to the directional approach which weights the value of each crack the same which in truth is not the correct procedure. This enhancement of classification will be hugely beneficial to the system as the crack percentage distribution currently in place will be developed into crack severity distribution. This information will be more useful to the inspectors as sectional information output will support the categorization of each section by severity thus highlighting areas of high crack distribution like before but areas of high severity. This severity classification is more important in tunnel asset management (TAM) than a crack percentage as in a tunnel. The preference of severity is attributed to the more informative output information as currently an area could be highly cracked with minor cracks and therefore classified as a concerning section when in reality it is of little concern. While if a severity approach is developed areas in which have a low crack percentage but contain a severe crack that requires immediate assessment to sustain serviceability would not be overlooked.

The incorporation of a directional methodology does not nullify the valuable information provided by the directional classification essentially improving the output information. This increase of suitability is mainly due to the directional classification being performed following severity categorization therefore only cracks of significance need to be classified. This in essence tackles the problems that currently are ingrained into the system of treating every crack detected equally which is not how visual investigations are approached. This will allow for a more nuanced directional classification system to be created which demonstrates better the behavioral circumstance of the tunnel structure. The system will not only be improved by the nuance but having the availability of the dimensions of the crack will facilitate the calculation of the orientation angle of a crack. This will significantly improve the directional classification system by reducing the remainder of the subjectivity that is engrained in developed CNN as with the orientation angle calculated angular bounds can be created to encompass each crack class and thus making it a more transparent classification system. The dimeson availability also unlocks the ability to expand the 4-class directional system currently in place to a more robust 5 class system with crescent-shaped cracks defining attribute of curvature which separates it from a diagonal crack has the potential to be calculated thus allowing for a more detailed analysis to be conducted.

Once the improvement in the collection and classification of the data is complete the contextualization becomes a much easier and less formidable prospect. The contextualization improvements are primarily centered on exposure to more data of tunnel sections of varying lengths and most important severity. This will allow the system to be adapted to what crack thresholds are necessary to separate the tunnels into the appropriate severity. The current problem concerning contextualization is based on the fact the CNN's used are trained on smaller images to scan larger images. The two-scan method is incredibly useful in enabling quality detecting and tracing of cracks in larger images however due to the fact of the operation method utilizing these smaller windows the system lacks context concerning the larger image. The development of the system to not only understand the window in question but to gain an understanding of what is happening in the conjoining windows and even further would help the system gain a better more global perspective on each image.

For brevity, future research may focus on the following 3 aspects:

1. Collection - The assortment of more data is always desired to enhance a deep Learning approach. However, the acquisition of more relevant data which contains a dimensional reference within every single data point is the aspect of the collection that will provide the most potential for improvement towards a fully automated system.

2. Classification - The availability of data with a dimensional frame of reference will enable the research to move towards a more conveniently dimensional crack classification methodology. The incorporation of dimension into the classification procedure permits more objective decisions to be made based on computed crack width and length in comparison to the current more subjective directional approach.

3. Contextualization - The improvements of collection and classification techniques provide a more sophisticated detection/ classification system. The exposure of this enhanced system to a wider variety of tunnel sections will enable the system to gain the necessary information to contextualize tunnels into the appropriate severity. The movement of the system from the singularity of a window-by-window classification towards a global image classification will significantly increase the contextual applicability of the classification system.

REFERENCES

Anon., 2021. *Dlpng.com.* [Online] Available at: https://dlpng.com/png/1131961 [Accessed 22 06 2021].

Anon., 2021. *K-Nearest Neighbors.* [Online] Available at: https://medium.com/@srishtisawla/k-nearest-neighbors-f77f6ee6b7f5 [Accessed 3 06 2021].

Artificial Intelligence Market Size & Share Report, 2.-2., n.d. *https://www.grandviewresearch.com/industry-analysis/artificial-intelligence-ai-market.* [Online].

Attard, L., 2018. Tunnel inspection using photogrammetric techniques and image processing: A review. *ISPRS Journal of Photogrammetry and Remote Sensing,* Volume 144, pp. 180-188.

Attard, L., Valentino & Gianluca, C. J. D., 2019. *Automatic Crack Detection Using Mask R-CNN.* Dubrovnik, s.n.

Ballard, D. H. & Brown, C. M., 1982. *Computer vision.* Englewood Cliffs, N.J.: Prentice-Hall.

Bapat, G., 2021. *Let's Understand the Difference between Machine Learning Vs. Deep Learning.* [Online] Available at: https://www.linkedin.com/pulse/lets-understand-difference-between-machine-learning-vs-gauri-bapat/ [Accessed 28 may 2021].

Boser, B., Guyon, I. & Vapnik, V., 1992. Training Algorithm for. *Proceedings of the fifth annual workshop on,* pp. 144-152.

CERN, n.d. [Online] Available at: https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.geplus.co.uk%2Fevents%2F20-may-cerns-hl-lhc-project-05-05-2021%2F&psig=AOvVaw201wGAnrV_jp1JCtp69loF&ust=1653222567353000&source=images&cd=vfe&ved=0CAwQjRxqFwoTCPiYppbM8PcCFQAAAAAdAAAAABAI

Chaiyasarn, K. et al., 2018. *Crack Detection in Masonry Structures using Convolutional Neural Networks and Support Vector Machine.* Berlin, ISARC, pp. 118-125.

Cha, Y.-J., Choi, W. & Büyüköztürk, O., 2017. Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks. *Computer-Aided Civil and Infrastructure Engineering,* 32(5), pp. 361-378.

Chen, F.-C. & Jahanshahi, M. R., 2018. NB-CNN: Deep Learning-Based Crack Detection Using Convolutional Neural Network and Naïve Bayes Data Fusion. *IEEE Transactions on Industrial Electronics,* 65(2), pp. 4392-4400.

Chul Min Yeum, S. J. D., 2015. Vision-Based Automated Crack Detection for Bridge Inspection. *Computer-Aided Civil and Infrastructure Engineering,* 30(10), pp. 759-770.

Cubero-Fernandez, A. et al., 2017. Efficient pavement crack detection and classification. *EURASIP Journal on Image and Video Processing,* Issue 1.

D. Scherer, A. M. a. S. B., 2010. Evaluation of pooling operations in convolutional architectures for object recognition. pp. 92-101.

Dongmei Han, Q. L. ,. F., 2018. A new image classification method using CNN transfer learning and web data augmentation. *Expert Systems with Applications,* Volume 95, pp. 43-56.

Dung, C. V. & Anh, L. D., 2019. Autonomous concrete crack detection using deep fully convolutional neural network. *Automation in Construction,* Volume 99, pp. 52-58.

Feng, C., -Yu Liu, M.-. & Teng-Yok Lee;, 2017. *Deep Active Learning for Civil Infrastructure Defect Detection and Classification.* s.l., s.n.

Gao, Y., 2018. Deep Transfer Learning for Image-Based Structural Damage Recognition. *Computer-Aided Civil and Infrastructure Engineering,* 33(9), pp. 748-768.

G. D. et al., 2019. Defect detection in tunnel images using random forests and deep learning. *10th International Conference on Pattern Recognition Systems.*

Gopalakrishnan, K., Khaitan, S. K. & Choudhary, A., 2017. Deep Convolutional Neural Networks with transfer learning for computer vision-based data-driven pavement distress detection,Ankit Agrawal. *Construction and Building Materials,* Volume 157, pp. 322-330.

Grosz, B. J. & Stone, P., 2018. A century-long commitment to assessing artificial intelligence and its impact on society. *Communications of the ACM,* 61(12), pp. 68-73.

Hadjidemetriou, G. M., Vela, P. A. & Christodoulou, S. E., 2018. Automated Pavement Patch Detection and Quantification Using Support Vector Machines. *Journal of Computing in Civil Engineering,* 32(1), p. 04017073.

Here, G., 2021. *5 Classification Algorithms you should know – introductory guide.* [Online] Available at: https://www.analyticsvidhya.com/blog/2021/05/5-classification-algorithms-you-should-know-

introductory-guide/
[Accessed 1 June 2021].
Hinton, V. N. a. G., 2010. Rectified linear units improve restricted Boltzmann machines. pp. 807-814.
Islam, M. M. M. & Kim, J.-M., 2019. Vision-Based Autonomous Crack Detection of Concrete Structures Using a Fully Convolutional Encoder–Decoder Network. *Sensors.*
K. Makantasis, E. P. A. D. N. D. a. C. L., 2015. Deep Convolutional Neural Networks for efficient vision based tunnel inspection. *IEEE International Conference on Intelligent Computer Communication and Processing (ICCP),* pp. 335-342.
Kim, H., Ahn, E., Shin, M. & Sim, S.-H., 2019. Crack and Noncrack Classification from Concrete Surface Images Using Machine Learning. *Sage,* 18(3), pp. 725-738.
. L. B., 2001. Random Forests. *Machine Learning,* 45(1), pp. 5-32.
Li, G. et al., 2020. Automatic Tunnel Crack Detection Based on U-Net and a Convolutional Neural Network with Alternately Updated Clique. *Sensors,* 20(3), p. 717.
Mei, Q. & Gül, M., 2020. Multi-level feature fusion in densely connected deep-learning architecture and depth-first search for crack segmentation on images collected with smartphones. 3 January.
Menendez, E. et al., 2018. Tunnel structural inspection and assessment using an autonomous robotic system. *Automation in Construction,* Volume 87, pp. 117-126.
Mokhtari, S., Wu, L. H.-B. & Yun, 2016. Comparison of Supervised Classification Techniques for Vision-Based Pavement Crack Detection. *Transportation Research Record: Journal of the Transportation Research Board,* 2595(1), pp. 119-127.
N. Tajbakhsh, J. S. S. G. R. H. C. K. M. G. J., 2016. Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning?. *IEEE Transactions on Medical Imaging*, 35(5), pp. 1299-1312.
N. S. a. T. C., 2016. *PREPARING FOR THE FUTURE,* s.l.: Executive Office of the President.
Protopapadakis, E. et al., 2019. Automatic crack detection for tunnel inspection using deep learning. *Applied Intelligence,* 49(7), pp. 2793-2806.
Quinlan, J., 1986. Induction of decision trees. *Machine Learning, 1,* pp. 81-106.
Ren, Y. et al., 2020. Image-based concrete crack detection in tunnels using deep fully convolutional networks. *Construction and Building Materials,* 234(117367).
Russakovsky, O. et al., 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision,* 115(3), pp. 211-252.
Samuel, A. L., 1959. Some Studies in Machine Learning Using the Game of Checkers. *IBM Journal of Research and Development*, 3(3), pp. 210-229.
Sari, Y., Prakoso, P. B. & Baskara, A. R., 2019. Road Crack Detection using Support Vector Machine (SVM) and OTSU Algorithm. *2019 6th International Conference on Electric Vehicular Technology (ICEVT),* Volume 10, pp. 349-354.
Sattar Dorafshan, R. J. T. ,. M. M., 2018. SDNET2018: An annotated image dataset for non-contact concrete crack detection using deep convolutional neural networks. *Data in Brief,* Volume 21, pp. 1664-1668.
Sharma, N., 2020. CRACK DETECTION IN CONCRETE USING TRANSFER LEARNING. *Advances in Mathematics: Scientific Journa,* 9(6), pp. 3895-3906.
Silva, W. R. L. d. & Lucena, D. S. d., 2018. Concrete Cracks Detection Based on Deep Learning Image Classification. *Proceedings,* 2(8), p. 489.
Sinha, S. K. & Fieguth, P. W., 2006. Neuro-fuzzy network for the classification of buried pipe defects. *Automation in Construction,* 15(1), pp. 73-83.
SPS, C., n.d. [Online]
Available at: https://cerncourier.com/wp-content/uploads/2020/09/eSPS-overview-jpg.jpg
Tunnels, C. d. d., 2015. *Road tunnel civil engineering inspection guide Book 2: Catalogue of deteriorations.* s.l.:Centre d'Études des Tunnels.
Turning, A. M., 1950. I.—COMPUTING MACHINERY AND INTELLIGENCE. LIX(236), pp. 433-460.
Vapnik, V. N., 1963. Pattern Recognition Using Generalized Portrait Method. *Automation and Remote Control,* pp. 774-780.
Younes, G., Hadda, C. & Djelloul, Z., 2009. Supervised learning and automatic recognition of. *IEEE/ACS International Conference on Computer Systems and Applications,* pp. 205-210.
Yuvaraj, N., Kim, B., Sri Preethaa, K. & Arun Pandian, R., 2021. Surface crack detection using deep learning with shallow CNN architecture for enhanced computation. *Neural Computing and Applications.*
Zhang, A. et al., 2017. Automated Pixel-Level Pavement Crack Detection on3D Asphalt Surfaces Using a Deep-Learning Network. *Computer-Aided Civil and Infrastructure Engineering,* 32(10), pp. 805-819.

Zhang, W., Zhang, Z., Qi, D. & Liu, Y., 2014. Automatic Crack Detection and Classification Method for Subway Tunnel Safety Monitoring. *Sensors*, 14(10), pp. 19307-19328.