Application of Deep Neural Network in TBM Parameters Prediction and Tunnel Inspection

Zhihang Li

Master of Engineering Science (Research)

A thesis submitted for the degree of
Master of Engineering Science (Research)

Department of Civil Engineering
Faculty of Engineering
Monash University

2021

Copyright Notice

Abstract

Tunnel boring machine (TBM) is one of the widely used equipment in tunnel construction, although TBM has many advantages over drilling and blasting method, its poor adaptability to geological conditions has always been the main problem it faces. Moreover, with the increase of the tunnel service life, the tunnel will face a series of safety problems, such as water leakage and cracks of tunnel lining, which will seriously affect the safety and life of the tunnel and cause irreversible damage.

Therefore, this thesis focuses on the application of deep neural networks in underground projects, referring to some classic performance prediction frameworks, convolutional neural network (CNN) network and characteristics, proposes using the deep learning-based algorithm on recurrent neural network (RNN) to predicted the performance of tunnel boring machine and convolutional neural network to detect defects of the underground tunnel.

The Long Short-Term Memory (LSTM) algorithm is applied to the prediction of TBM performance. LSTM is a prediction method based on RNN which uses the gate system to control the hidden layer and improve the prediction accuracy. The hyperparameters in the LSTM model are optimized to achieve better prediction results for TBM operating parameters, the average prediction accuracy is increased by 7%. Based on the existing TBM operation data, the improved LSTM machine learning method can effectively predict the TBM tunnelling parameters. The mean absolute percentage errors (MAPE) are less than 12% and the fitting degree R^2 is greater than 0.89, which has relatively high prediction accuracy and can provide guidance for the parameter selection of TBM during the stability excavation section.

The Faster Regional convolutional neural network (Faster R-CNN) and the YOLOv3 (You Only Look Once version 3) are applied to the detection of TBM lining cracks. Faster R-CNN is a two-stage method of object detection that uses regional proposals to achieve higher detection accuracy. However, the detection speed of the algorithm cannot satisfy the real-time monitoring requirement. Therefore, the YOLOv3 is applied to the detection of TBM lining cracks. The algorithm of the YOLO series is a typical representative algorithm based on the one-stage method, and the characteristics of the one-stage method are it has a relatively faster detection speed compared with the two-stage. Both of the algorithms are optimized according to the characteristic of cracks and the detection accuracy of R-CNN and YOLO are 0.88 and 0.84, respectively. In addition, to test the robustness of the algorithm, noise is added to the test image set. The results of detection show that compared with YOLO, the R-CNN algorithm has better robustness when facing images of tunnel cracks containing noise, and expect to have better working performance in the underground tunnel.

Keywords

Deep learning, TBM, performance prediction, lining defect inspection, long short-term memory, Faster R-CNN, two stage algorithm, YOLO, one stage algorithm, robustness test.

Declaration

This thesis is an original work of my research and contains no material which has been accepted for the award of any other degree or diploma at any university or equivalent institution and that, to the best of my knowledge and belief, this thesis contains no material previously published or written by another person, except where due reference is made in the text of the thesis.

Signature: ……………………

Print Name: Zhihang Li

Date: 05 March 2021

Acknowledgements

The author would like to express his sincere thanks to the following who assisted in the completion of this works:

Professor Jian Zhao, the author’s supervisor, for guidance and encouragement throughout the research and for his carefully instruct;

Professor Yu Bai, the author’s Co-supervisor, for inspiration and technical support throughout the research;

Dr. Qianbing Zhang, for immense and meticulous help throughout the research and for his remarks and criticism, particularly with the structures and English expression, during the writing of this thesis;

Professor Zuyu Chen and China Railway Engineering Equipment Group Co., Ltd, for kindly sharing the TBM operating data.

The authors also would like to thank the colleges in the research group, whose masterful programming skills on the model establishment and incisive and constructive discussion have improved this thesis greatly.

Finally, the author would like to express his heartfelt thanks to his parents for their enormous love and courage during the period of study.

Table of Contents

List of Figures

List of Tables

List of Equations

Chapter 1: Introduction

1.1 Research Background and Significance

With the development and population growth of the metropolitan area, ground transportation is becoming more and more congested. As a part of urban traffic, the underground tunnel greatly relieves the pressure of surface traffic and plays a significant role in the modern transportation system.

Tunnel boring machine (TBM) is one of the widely used equipment in tunnel construction, it has many advantages over conventional tunnelling methods (e.g., drilling and blasting method), such as fast excavation speed, environment friendly construction and continuous (non-cyclic) operation. However, its poor adaptability to geological conditions and high dependence on rock mass parameters has always been the main problem it faces. If the excavation plan hasn't adjusted effectively and on time once the geological condition change, the TBM failure may occur, cause unpredictable loss. Therefore, the adaptable adjustment of TBM to optimized the operating status for preventing potential damage to TBM has become a research focus. Since the prediction of tunnel geological conditions is still challenging before excavation, the prediction of important TBM operating parameters plays an important role in the research on TBM adaptable adjustment. Accurate prediction of TBM performances is required for time planning, cost control and choice of excavation method in order to make tunnelling economical. The TBM performance prediction is certainly a broad topic, both of the TBM operating parameters and the rock mass parameters are going to significantly affect the prediction results. The successful prediction could make underground tunnelling excavation economical and prevent potential hazards. However, the various parameters of TBM have a complex relationship, which is difficult to create a simple mathematical model. Fortunately, neural networks offer an alternative tool for establishing the relationship between different parameters.

Moreover, after the tunnel construction is completed and opened to traffic, with the increase of service life, the quality of the tunnel will inevitably suffer from the erosion and damage of external forces and natural factors, resulting in the hidden dangers such as cracks, seepage, peeling and other defects. Defects arising in the tunnel are inevitable, therefore it requires inspection of tunnels and discovers defects in time to repair them. In the early period, the routine inspection of tunnels mainly relies on manpower, however, it has servals disadvantages such as low detection efficiency, strong subjectivity, low accuracy, high labour cost, the unfavourable working environment of labour, etc. With the rapid development of computer technology, many technologies based on computer vision and image processing emerge, the research of tunnel defects detection is gradually changing from the traditional non-destructive detection technology to the detection and recognition based on image processing. The existing machine learning based algorithms have good recognition performance on the simple crack image while it requires intervention and empirical judgment by experts. Moreover, the underground tunnel has inadequate working conditions for electronic devices such as high temperature, cold weather, high humidity and the image collect from the site may experience uneven illumination cause lots of noise exist on graph while the lining has complex texture features. Those factors lead it is difficult to achieve higher accuracy of defects recognition and to meet the industrial requirement for field application. Therefore, it is necessary to have an intelligent and efficient way to detect the defects of tunnels.

Recently, deep learning represents the state of the art of artificial intelligence, which has successfully been applied to different aspects such as image recognition, text translation, natural language processing, etc., and achieves huge success. The conventional machine learning based image detection and recognition methods have been gradually replaced by more intelligent and effective deep learning algorithms.

Deep learning is the further development of the artificial neural network. By pre-training the neural network layer by layer, the feature expression of different levels can be learned, and the feature expression of each layer is obtained through the previous expression propagation, then all the layers are composite to form a deep convolutional neural network. Compared with the conventional machine-learning and image classification algorithm, the deep convolutional neural network has superior performance in parameter prediction and image classification. It is suitable for processing of a large amount of data, which is not only can be used TBM operating parameters prediction but also could be used in the detection of tunnel complex defects for underground projects.

1.2 Research Aim and Objects

Therefore, this thesis focuses on the application of deep neural networks in underground projects, referring to some classic performance prediction frameworks, convolutional neural network (CNN) network and characteristics, proposes using the deep learning-based algorithm on recurrent neural network (RNN) to predicted the performance of tunnel boring machine and CNN to detect defects of the underground tunnel. The main contents of this thesis are as follows:

1. The Long Short-Term Memory (LSTM) algorithm is applied to the prediction of TBM performance. LSTM is a prediction method based on RNN which uses the gate system to control the hidden layer and improve the prediction accuracy. The results show that LSTM accurately predicted the performance of TBM.
2. The Faster Regional convolutional neural network (Faster R-CNN) is applied to the detection of TBM lining cracks. Faster R-CNN is a two-stage method of object detection that uses regional proposals to achieve higher detection accuracy. However, the detection speed of the algorithm cannot satisfy the real time monitoring requirement. In order to test the robustness of the algorithm, noise is added to the test image set. The result shows that the Faster R-CNN algorithm has better robustness when facing images of tunnel cracks containing noise, and expect to have better working performance in the underground tunnel.
3. The YOLOv3 (You Only Look Once version 3) algorithm is applied to the detection of TBM lining cracks. The algorithm of the YOLO series is a typical representative algorithm based on the one-stage method, and the characteristics of the one-stage method are it has a relatively faster detection speed compared with the two-stage. the parameters in the algorithm are adjusted to suit of detect small objects such as cracks. The test results show that the improved algorithm has better detection accuracy than before, and the detection speed could satisfy the real-time detection requirements. In addition, the noise is added to the test image set for testing the robustness of the algorithm. The results show that the YOLO algorithm has poor performance when processing images of tunnel cracks with noise.

1.3 Thesis Outline

In this thesis, the application of deep learning in TBM operating parameters prediction and defects detection during tunnel maintenance is proposed. The remainder of this thesis is organized as follows. The next four chapters have a logical progression, but they may be read independently.

Chapter 2 introduces the basic knowledge about TBM, the development of deep learning and its application in underground engineering.

Chapter 3 is the establishment of the TBM operating parameters prediction model and its application on operating parameters prediction. The LSTM model based on RNN is established for TBM operating parameters prediction.

Chapter 4 is the application of tunnel defects detection based on two stage object detection algorithm and an extra robustness test of the model. The two-stage algorithm-based R-CNN model is established to detect the cracks in concrete walls. Moreover, the artificial noises are added to the test sets to verify the robustness of the model.

Chapter 5 is the application of tunnel defects detection based on one stage object detection algorithm and an extra robustness test of the model. The two-stage algorithm-based YOLO model is established to detect the cracks in concrete walls. Moreover, the artificial noises are also added to the test sets to verify the robustness of the model.

Chapter 6 is the conclusion and future work.

Figure 1.1 demonstrates the thesis structure.

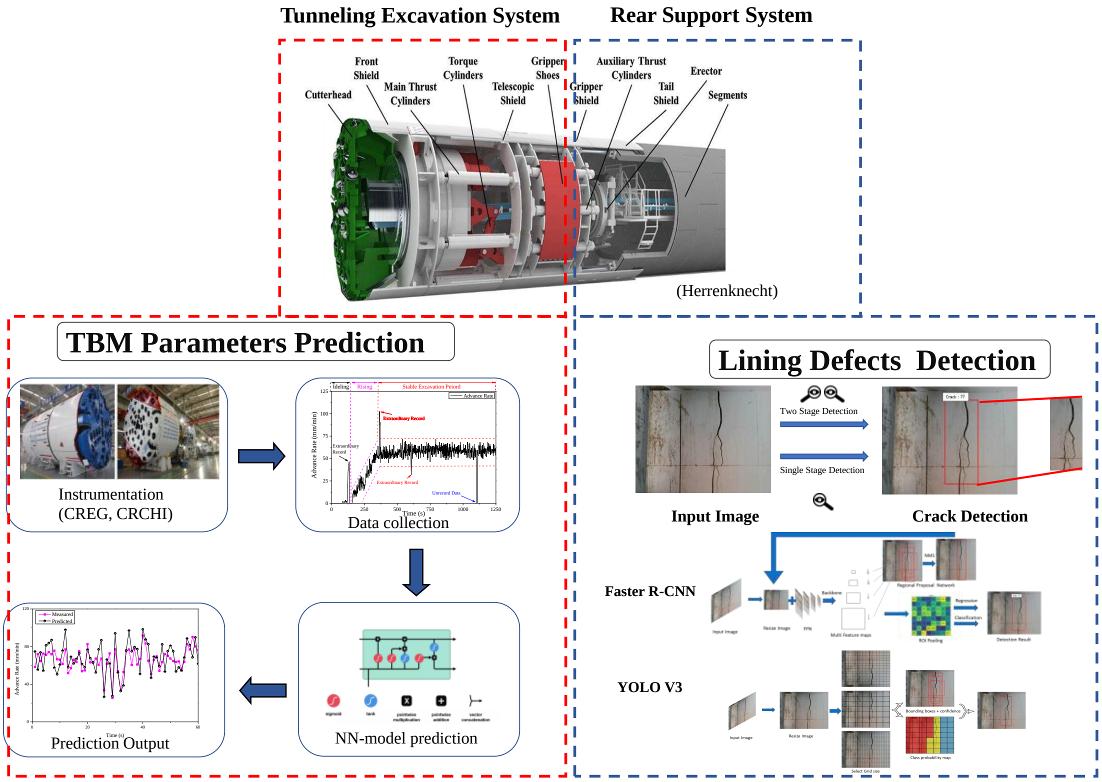


Figure 1.1 Thesis structure

Chapter 2: Literature Review

2.1 Empirical TBM Performance Prediction Model

Tunnel boring machine (TBM) is underground engineering equipment (Nelson, 1983, Bruland, 2000) that integrates the functions of tunnelling, guidance, support, mucking, ventilation, transportation, etc (Zhao et al., 2007a). TBM has fast excavation speed and safe and reliable construction quality compared with traditional drilling and blasting methods (Rostami and Ozdemir, 1993), which can greatly shorten the project period. Figure 2.1 demonstrates the structure and components of TBM.

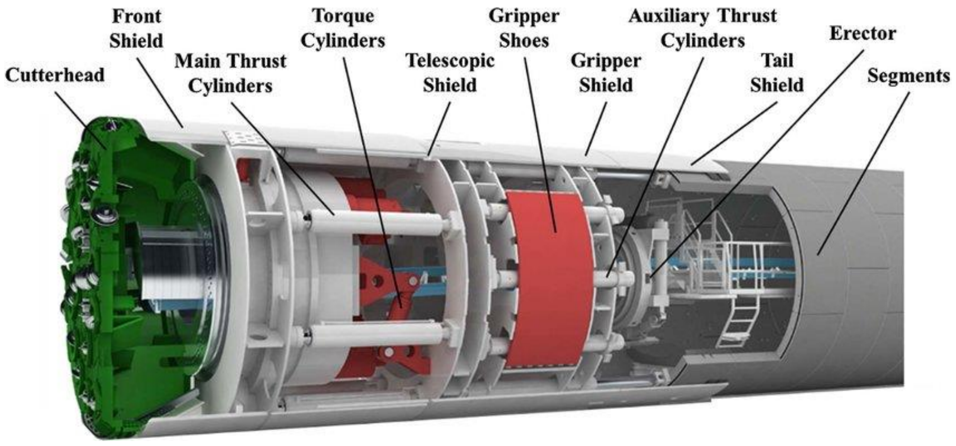


Figure 2.1 Structure and components of TBM (Herrenknecht)

TBM can be divided into two main types according to the different structure of the machine. According to its different application scenarios, it can be divided into two major types: One is shield machine which is using segment lining to support the tunnel, and the other is open type tunnel boring machine that can excavate rock geology that without segment lining support (ITA, 2000). The specific classification of TBM is shown in Figure 2.2. and Table 2.1 summarize the specific classification of TBMs and its application scenarios.

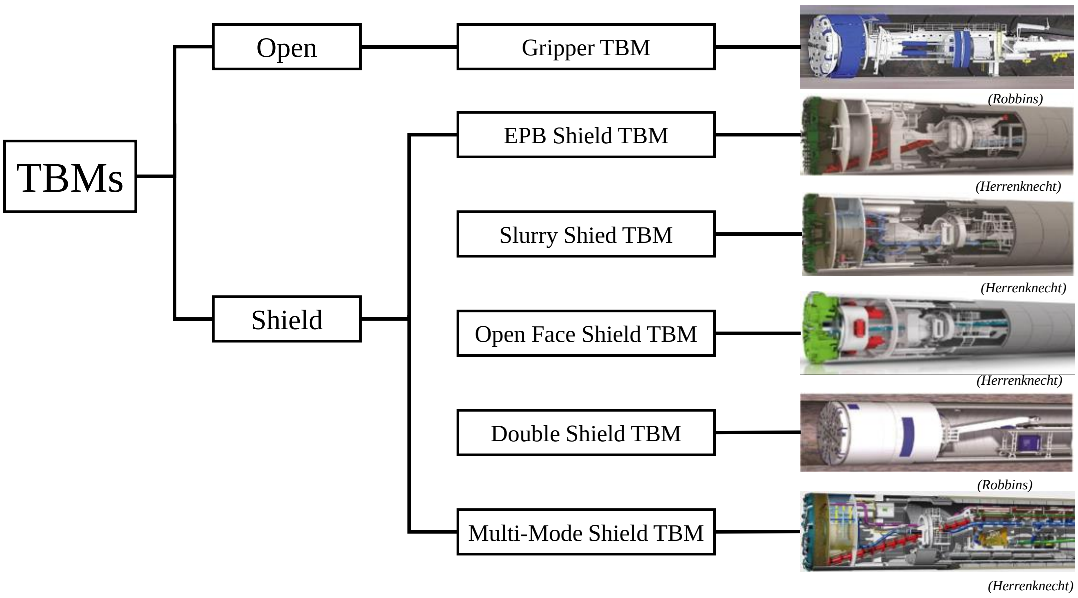


Figure 2.2 Classification of TBM

Table 2.1 Classification of TBM, modified type, suitable grounds and support structures

TBM Type	Common Name	Modified Type (Note)	Suitable Ground	Tunnel Support
Non-Shield	Gripper		Competent rocks	Bolt, shotcrete, wiremesh, steel set
Shield	EPB Shield		Clayey-silty soils	Segmental lining
		EPB Mix-Shield (with roller cutters)	Clayey soil-rock mixed	
	Slurry Shield		Sandy-gravelly soils	
		Slurry Mix-Shield (with roller cutters)	Sandy soil-rock mixed	
	Open Face Shield	(Often with mechanical face support)	Firm soils	
		Open Mix-Shield (with roller cutters)	Firm soil-rock mix	
	Double Shield	(Mix of gripper with any shield type)	Soil-rock mix	
	Multi-Mode Shield	(Mix of EPB, slurry and open modes)	Clayey-sandy-firm soils	

Note. EPB=Earth Pressure Balance.

Generally, the shield machine uses scrapers to excavate soft soil, and the focus point is the ground subsidence and the instability of the working face. While the hard rock tunnel boring machine uses the disc cutters to excavate the rock, and the key problem to be addressed is the excavation efficiency (Rostami and Ozdemir, 1993, Barton, 1999, Benardos and Kaliampakos, 2004).

The TBM is generally composed of two parts, the rear supporting system and the main engine system (Bruland, 2000, Yagiz and Karahan, 2011). The principle of hard rock TBM excavating tunnel is (Gong et al., 2006, Ghasemi et al., 2014, Zhao, 2006): when the TBM excavates the tunnel, the cutter head continuously rotates, and the disc cutters on the cutter head continuously cut the rock under the action of the propulsion force provided by the propulsion system. Each disc cutter creates a circular cutting track of different sizes on the face of the tunnel, and the rock will continuously broke and fallen off by the disc cutters and turns into slag (Bruland, 2000). The stone slag will be automatically collected, sent to the main belt conveyor by the slag shovel device on the cutter head, then the stone slag transported to the auxiliary belt conveyor of the supporting system through the main belt conveyor, and transported to the slag truck (Herrenknecht, CRCHI, CREG, Robbins). When the driving rod of the propulsion system reaches a certain length, the TBM will stop excavation, and the rear supporting system will move forward. The driving force of the propulsion system is provided by the friction between the boot and the wall of the tunnel. At present, TBM has been widely used in tunnel excavation, mining, water conservancy and hydropower projects (Li et al., 2017).

Tunnel boring machine (TBM) is one of the widely used equipment in tunnel construction, it has many advantages over conventional tunnelling method (e.g., drilling and blasting method), such as fast excavation speed, environment friendly construction and continuous (non-cyclic) operation (Rostami and Ozdemir, 1993, Barton, 1999, Bruland, 2000, Gong et al., 2006, Zhao et al., 2007b, Li et al., 2017). However, its poor adaptability to geological conditions and high dependence on rock mass parameters has always been the main problem it faces. If the excavation plan hasn't adjusted effectively and on time once the geological condition change, the TBM failure may occur, cause unpredictable loss. Therefore, the adaptable adjustment of TBM to optimized the operating status for preventing potential damage to TBM has become a research focus. Since the prediction of tunnel geological conditions is still challenging before excavation, the prediction of important TBM operating parameters plays an important role in the research on TBM adaptable adjustment. Accurate prediction of tunnel boring machine (TBM) performances are required for time planning, cost control and choice of excavation method in order to make tunnelling economical.

However, TBM has poor adaptability to geological conditions and high dependence on rock mass parameters has always been the main problem it faces (Li et al., 2017, Zhang et al., 2019). If the excavation plan hasn't adjusted effectively and on time once the geological condition change, the TBM failure may occur, cause unpredictable loss. Therefore, the adaptable adjustment of TBM to optimized the operating status for preventing potential damage to TBM has become a research focus. Since the prediction of tunnel geological conditions is still challenging before excavation, the prediction of important TBM operating parameters plays an important role in the research on TBM adaptable adjustment. Accurate prediction of tunnel boring machine (TBM) performances is required for time planning, cost control and choice of excavation method in order to make tunneling economical.

The prediction of TBM operating parameter has been studied for many years and various predictive model has been developed. The existing prediction model could be divided into theoretical and empirical models (Sapigni et al., 2002). Among the Colorado School of Mines (CSM) model proposed by Rostami and Ozdemir (1993) is the most typical theoretical model (Rostami, 1997, Hassanpour et al., 2011). The model is based on the rock strength index, the cutter spacing and the corresponding grooving depth to estimate the TBM penetration rate. Table 2.2 summarizes the advantages and disadvantages of different types of models for performance prediction of hard rock TBMs.

Table 2.2 Advantages and disadvantages of different types of models for performance prediction of rock TBMs (Rostami, 2016)

Model types	Advantages	Disadvantages
Theoretical	Flexible with cutter geometry and machine specifications	Unable to easily account for rock mass parameters
	Can be used in trade-off between thrust and torque and optimization	Lack of accounting for joints
	Can be used for cutter head design and improvements	Can be off by a good margin in jointed rock
	Can explain the actual working condition of the discs and related forces	Inability to account for required field adjustments
Empirical	Proven based on observed field performance of the TBMs in the field	Lower accuracy when used in cases when input parameters are beyond what was in the original field performance database
	Accounts for TBM as the whole system Many of field adjustments are (i.e., average cutter conditions) implied	Unable to account for variations in cutter and cutter head geometry, i.e. cutter tip width, diameter, spacing, gage arrangement
	Ability to account for rock joints and rock mass properties	Extremely sensitive to rock joint properties

However, the laboratory test is difficult to simulate the geological conditions of the rock mass comprehensively and accurately at the excavation site, while the prediction error usually exceeds the requirement (Gong et al., 2006, Zhao et al., 2007b). Compared with the theoretical model, the establishment of the empirical model is generally based on a large volume number of field measured data, the factors considered are more comprehensive, and the prediction accuracy is higher under similar stratum conditions, thus occupying the mainstream of performance prediction research (Yagiz and Karahan, 2015). Early empirical models were based solely on a single rock mass strength index and presented empirical expressions of TBM performance indicators (Graham, 1976, Farmer and Glossop, 1980, Hughes, 1986).

Table 2.3 summarizes the different model of TBM operation parameters predictions by the empirical method.

Table 2.3 Different models of TBM operation parameters predictions by the empirical method

Method	Applications	Parameters	Model	Reference
Empirical method	Penetration rate	Mechanical data and Geological Data	CSM	(Rostami and Ozdemir, 1993)
	Penetration rate	Mechanical data and Geological Data	NTNU	(Bruland, 2000)
	Penetration rate	Mechanical data and Geological Data	SRMBI	(Gong and Zhao, 2009)
	Penetration rate	Mechanical data and Geological Data	NTNU	(Benato and Oreste, 2015)
	Penetration rate	Mechanical data and Geological Data	HC	(Liu et al., 2017)
	Penetration rate Advance rate	Mechanical data and Geological Data	Q_{tbm}	(Barton, 1999)
	Field Penetration Index (FPI)	Mechanical data and Geological Data	RMR	(Khademi Hamidi et al., 2010)
	Field Penetration Index (FPI)	Mechanical data and Geological Data	Regression Analysis	(Hassanpour et al., 2011)

Note. CSM= Colorado School of Mines, SRMBI= Specific Rock Mass Boreability Index, NTNU= Norwegian University of Science and Technology, HC= Hydropower Classification, RMR= Rock Mass Rating.

Researchers have successively proposed complex models with multiple factors. a series of empirical relationships on TBM excavation performance based on a large number of on-site measured data are proposed by Bruland of the Norwegian University of Science and Technology (NTNU), through regression analysis of rock mass parameter and TBM data (Bruland, 2000). P.P.Nelson divided the field collected data into four databases according to the detailed data, and used the probabilistic method to simulate the operating parameters of the TBM, and then established a probability model to predict the construction performance of the TBM (Nelson, 1983). Gong and Zhao (2009) combining the rock mass conceptual model for evaluating rock mass boreability with the established database, a statistical prediction model of TBM penetration rate is set up by performing a nonlinear regression analysis. In addition, empirical models with rock mass grades as input parameters have also been widely used, such as Q_{tbm}, Rock Mass Rating (RMR) (Barton, 1999, Khademi Hamidi et al., 2010). These models are generally created on the grounds of knowledge acquired and information collected from previous tunnelling projects to derive the complicated and non-linear connection between the TBM operating parameters and the parameters of rock mass affecting. Table 2.4 summarizes the different models of TBM performance prediction.

Table 2.4 Different models of TBM performance prediction

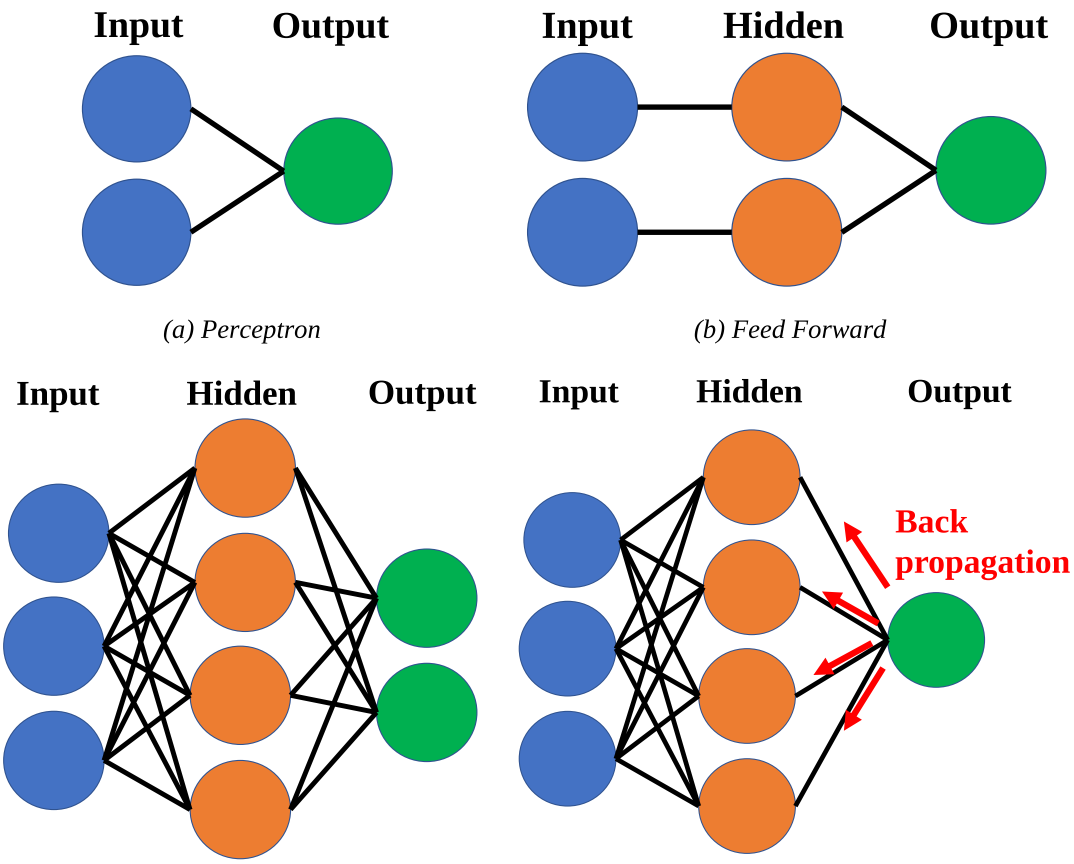
	Example	**Advantages**	**Disadvantages**
Simple models	(Graham, 1976)	•Easy to apply	•Might underestimate due to lack of joint parameters •Limited range of application
Multiple parameters models	CSM (Rostami and Ozdemir, 1993, Rostami, 1997), NTNU (Bruland, 2000), Q_{TBM} (Barton, 1999)	•Accounting for both rock mass and TBM parameters •Relying on good database	•Several parameters •Complex relationships •Using uncommon tests
Probabilistic models	(Laughton, 1998)	•Accounting for randomness and approximation	•Lack of detailed information from a like-case tunnel
Computer-aided models	Neural network models (Alvarez Grima, 2000, Zhao et al., 2007b, Gong and Zhao, 2009)	•Relying on good database	•Complex underlying structure •Over fitting •Usually not available in public domain

With the development of TBM, there are dozens of kinds of data directly collected by TBM through sensors, including a variety of parameter types that have a great influence on the tunnelling speed, such as total thrust, penetration, cutterhead speed and cutterhead torque. The reasonable use of these data has become a key issue.

2.2 Machine Learning Based TBM Performance Prediction Model

Correlation analysis of TBM operating parameters and prediction of advance speed is an important research topic of TBM excavation, among them, the selection of geological parameters, mechanical parameters and operating parameters of TBM could significantly influence the excavation efficiency of TBM (Yagiz and Karahan, 2011, Wang et al., 2012, Li et al., 2017, Liu et al., 2017, Gong and Zhao, 2007). During the excavation period, the sensors installed on TBM record massive operating data of TBM, however, due to the existing model could not utilize such a huge data set, thus a large number of field excavation data are sealed up, resulting in waste. Compared with the conventional methods, machine learning can effectively use a large amount of data as training sets, so that the field collected data can be utilized without being wasted (Zhao et al., 2007b, Yagiz and Karahan, 2011, Zhang et al., 2019, Gao et al., 2020, Wei et al., 2020). Moreover, compared with the conventional methods, machine learning has higher training efficiency, which means it has potential higher prediction accuracy.

Artificial intelligence originally proposed in the 1950s (Rosenblatt, 1958), refers to the idea of the neural network from the biological model of the human brain, the brain is made up of an infinite number of neurons, each of which plays its own role. a series of neurons are used to simulate the neural network structure of the human brain which is the artificial neural network. It was constructed by heuristic simulation of the human brain mechanism, and researchers used computer code to realize this network structure. The artificial neural network is based on the mechanism of simulating the information transmission of neurons in the human brain. Biological studies show that the synaptic part of neurons is the key to information transmission, and the activity of information transmission between two neurons is positively correlated with the connection intensity of the synapses between them. The smallest unit of an artificial neural network is a single neuron. Figure 2.3 shows the structure of different neural networks.



(a) Perceptron

(b) Feed Forward

(c) Deep Feed Forward

(d) Back propagation

Figure 2.3 Structure of different neural networks (Lipton, 2015)

The artificial neural network is divided into three parts which are the input layer, the hidden layer and the output layer (Rosenblatt, 1958, LeCun et al., 2015, Lipton, 2015). Every single layer has a series of neurons that are not connected to each other in the same layer while the hidden layer can have multiple layers. The network extracts the feature of expression from a large amount of input data through the hidden layer to cope with different tasks for the data processing required in the application.

Machine learning uses algorithms to analyse data, train from it, then make decisions and predictions based on previous data (Hinton et al., 2012). Different from the traditional software programs to solve specific tasks, machine learning uses a lot of data to "train" and uses various algorithms to learn how to complete tasks from the data. In terms of learning methods, machine learning algorithms can be divided into supervised learning (such as classification problems), unsupervised learning (such as clustering problems), semi supervised learning, integrated learning, deep learning and reinforcement learning.

At present, with the development of artificial intelligence technology, a large number of machine learning methods are widely used to deal with the regression relationship between rock mass conditions and operating parameters to establish TBM performance prediction models, such as the artificial neural network (ANN) (Eftekhari et al., 2010, Mohammadi et al., 2015, Afradi et al., 2019), particle swarm optimization (PSO) (Yagiz and Karahan, 2011), Support Vector Machine (SVM) (Mahdevari et al., 2014, Fattahi and Babanouri, 2017, Afradi et al., 2019) and Fuzzy Logic (FL) (Ghasemi et al., 2014, Yagiz and Karahan, 2015). The above machine learning models have been verified by engineering and have practical value in engineering. Table 2.5 summarizes the method of the TBM performance prediction model.

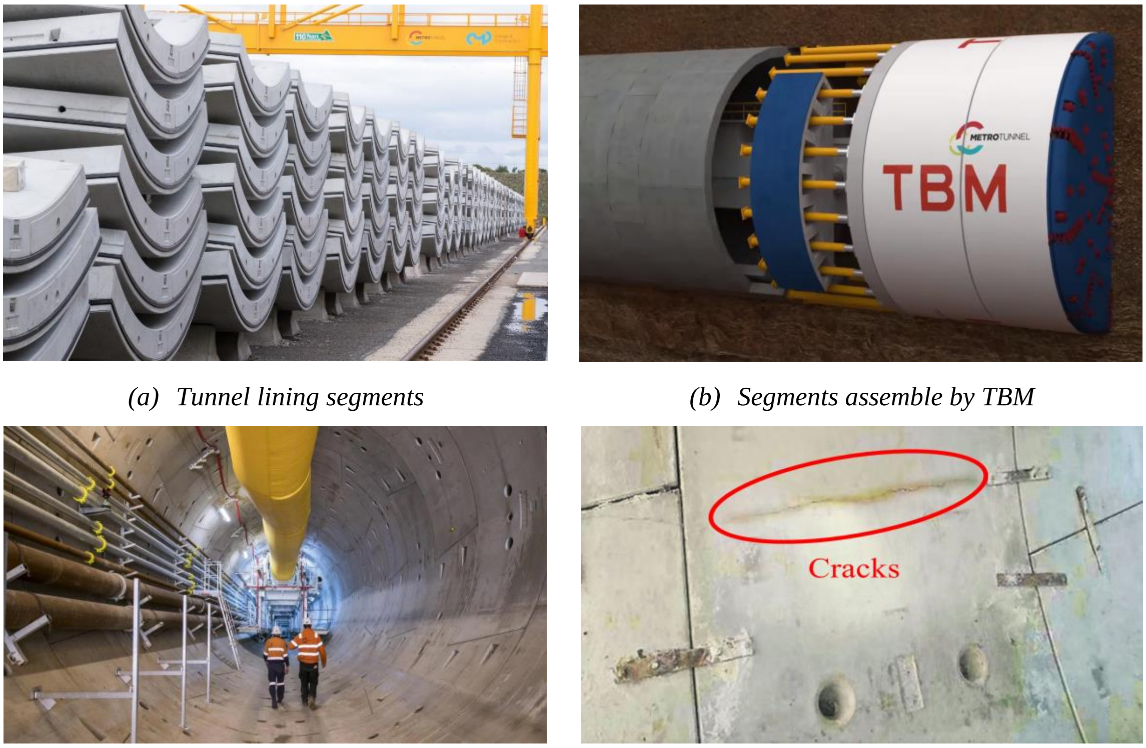
Table 2.5 Different model of TBM operation parameters predictions by conventional machine learning

Method	Applications	Parameters	Model	Reference
Conventional Machine-Learning	Penetration rate	Mechanical data and Geological Data	ANN	(Eftekhari et al., 2010)
	Penetration rate	Mechanical data and Geological Data	ANFIS	(Alvarez Grima et al., 2000)
	Penetration rate	Mechanical data and Geological Data	ANN, SVM	(Afradi et al., 2019)
	Penetration rate	Mechanical data and Geological Data	ICA-ANN, PSO-ANN	(Armaghani et al., 2017)
	Penetration rate	Geological Data	Fuzzy Logic	(Ghasemi et al., 2014)
	Penetration rate	Mechanical data and Geological Data	ANN/fuzzy logic	(Yagiz and Karahan, 2015)
	Penetration rate	Geological Data	PSO	(Yagiz and Karahan, 2011)
	Penetration rate	Geological Data	ELM	(Shao et al., 2013)
	Penetration rate	Mechanical data and Geological Data	SVR	(Mahdevari et al., 2014)
	Penetration rate	Geological Data	Random Forests	(Tao et al., 2015)
	Penetration rate	Geological Data	Bayesian inference	(Adoko et al., 2017)
	Penetration rate	Geological Data	SVR	(Fattahi and Babanouri, 2017)

Note. ANN = Artificial Neural Network, ANFIS= Adaptive Neuro Fuzzy Inference System, SVM= Support Vector Machine, ICA= Imperialist Competitive Algorithm, PSO= Particle Swarm Optimization, ELM= Extreme Learning Machine, SVR= Support Vector Regression.

2.3 Inspection of TBM Segments Defects

During TBM tunnel excavation, precast concrete segment linings are be used for lining and supporting (Herrenknecht). Because the tunnel excavation radius is larger than the segment support radius, there is a gap between the surrounding rock and the lining segment support. After the tunnel is in contact with the surrounding rock, the lining segment and surrounding rock bear the surrounding rock pressure (Zhou et al., 2019, Gall et al., 2018). Therefore, the tunnel lining will appear irreversible cracks with the increased service life of the tunnel. The seemingly insignificant minor cracks may lead to major safety hazards, directly affect the operation of the underground transport, and even lead to life and property safety threats when serious accidents occur. Therefore, it is necessary to detect cracks efficiently and accurately. Figure 2.4 demonstrates the tunnel lining in underground tunnel construction by TBM.



(a) Tunnel lining segments

(b) Segments assemble by TBM

(c) Constructed tunnel by TBM

(d) Cracks on lining

Figure 2.4 Tunnel lining in underground tunnel construction by TBM (RailProjectVictoria)

The current research on detection and recognition of TBM tunnel lining cracks can be divided into two categories. One is the fixed crack detection method based on deformation data (Li et al., 2015), another is a crack detection method based on image processing (Davoudi et al., 2018). In most fixed crack detection methods, the sensor that collects crack deformation data is installed directly on the tunnel, certain algorithms are used to evaluate them to obtain the current situation of crack propagation and the damage level once the corresponding data are detected. This method requires installing the sensor in the tunnel and it is resources consuming, therefore the engineers proposed methods based on computer vision for cracks detection. Cracks detection methods based on computer vision do not need to install any sensors in the tunnel, it usually builds a mobile platform with cameras and light sources to collect the image set of tunnels. Certain algorithms are used to recognize and locate the feature of cracks in the image (Jiang et al., 2019). The advantage of the method is it has higher detection efficiency compare with the fixed crack detection method and less financial cost.

2.4 Conventional Machine Learning Object Detection Method

In the early period, the routine inspection of the tunnel mainly relies on manpower. This rather intrusive way of inspection not only places workers in great danger but can also result in subjective and low-precision observations. Therefore, the vision-based inspection that assists operators with advanced optical devices to record the conditions via images or videos and analyse with computer-based techniques has been greatly promoted (Attard et al., 2018).

The vision-based crack identification methods can be roughly divided into two main categories, the conventional machine learning based algorithm (Gall et al., 2018) and the novel deep learning-based algorithm. The conventional algorithm is generating bounding boxes by selective search method to extract features, then the classifier is used to identify the objects in the image and classify the cattery of the objects. The deep learning-based algorithm has a powerful ability of feature extraction and classification. The disadvantage of it is that it contains a large number of parameters and complex network structures, and the model training is time consuming and requires large computing power. Therefore, it is necessary to use high-performance computing platforms and massive data sets to assist the learning and training of neural networks.

Traditional machine learning can be dividing into three steps (LeCun et al., 2015, Dalal and Triggs, 2005). Firstly, using the selective search method to acquire the object in the image and generate the bounding boxes. Then, a specific algorithm is used to extract the features of the object for analysis and comparison. Finally, the features of the object are feed into the pre-trained classifier to indicate the category of object and mark the position of the object.

2.4.1 Bounding Box Generate.

Figure 2.5 shows the process of conventional machine learning based object detection method

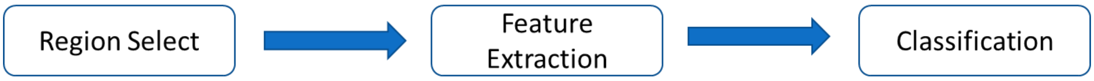


Figure 2.5 Process of conventional machine learning based object detection method

The first step in the machine learning method is to select the object area in the image and generate a bounding box. There are serval methods for generating bounding boxes such as selective search (Uijlings et al., 2013), sliding windows (Zitnick and Dollár, 2014), etc. Considering that the object may appear in any region of the image and its size and aspect ratio are not fixed, it is necessary to set a specific scale proportion to improve the accuracy of different objects. The sliding window is a common method to detect the target object in the image in the machine learning algorithm due to its versatility. Figure 2.6 shows the process of the sliding window method.

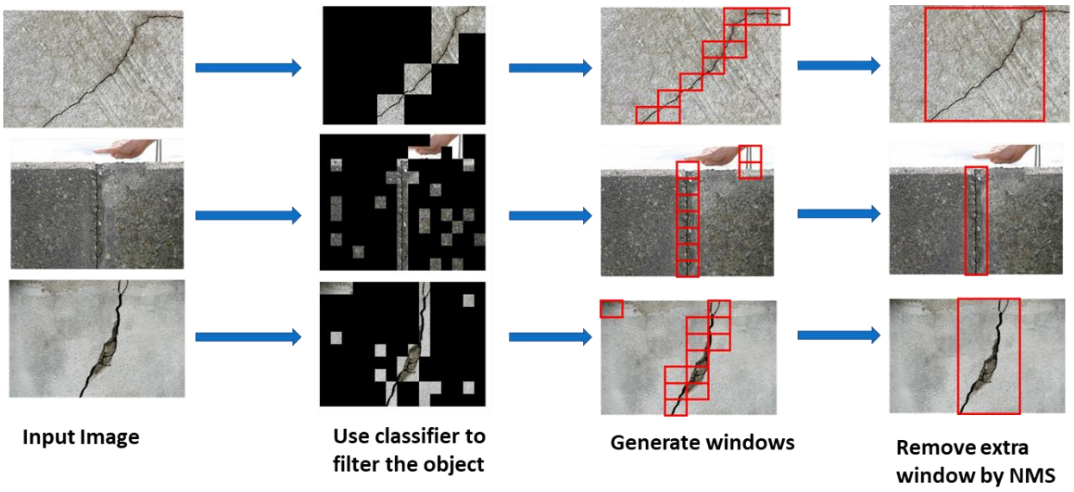

Input Image
Use classifier to filter the object
Generate windows
Remove extra window by NMS

Figure 2.6 Process of sliding window method

First, the input image is divided into different sizes of the window, then the pre-trained classifier is used for classification. Each window will be scored by the classifier in order to estimate the probability of the object existing in the window. For instance, if the current window has a high classification probability, which is higher than a certain threshold, it is considered that the target object has been detected, a prediction box will be generated. When there are numbers of windows, more similar prediction boxes may be output on the same object. To make the results more concise, the similar prediction bounding boxes can be remove but leave one unique detection box by using the non-maximum suppression (NMS) (Hosang et al., 2017) method. However, the disadvantage of the sliding windows is it using the method of similarity exhaustion to search subregions of an image, which most subregions are not contained objects. As a result, the method is time-consuming and inefficient which not ideal for real time object detection.

2.4.2 Feature Extraction

Different objects often have different forms and are susceptible to light conditions, noise and background effects, it is hard to have a feature extract method that works for all objects which makes it difficult to design a feature that works for all objects. Therefore, different features extract method should be designed according to the actual detection scenarios and requirements. Common feature extractor includes Scale-invariant feature transform (SIFT) (Lowe, 1999), Histogram of Oriented Gradient (HOG) (Dalal and Triggs, 2005), Haar-like (Papageorgiou et al., 1998), etc.

SIFT is a classic feature extract method (Lowe, 1999) which uses the difference of Gaussian function to build different scale spaces for detecting the feature points under these scales, while the local feature points are being processed. It has a very high tolerance for image rotation change, illumination brightness change and other interfering factors, and can reduce the impact of object deformation during processing. Therefore, the SIFT method has excellent robustness when facing interference.

HOG is a feature extraction method that has better performance on extract partial information of object shape which ideal for boundary extraction (Dalal and Triggs, 2005). The process of HOG is the image normalized to adjust the uneven brightness in the image at the initial stage, then the image is divided into different cell units to calculate its gradient histogram. In the final stage, the results are combined into the feature and the features represent the input image. HOG method focuses on the gradient information of the image and has a certain tolerance for some other interference factors. For example, it can allow a certain degree of geometric and optical deformation of the image, and also resist the influence of illumination and noise. Usually, the HOG method is combined with the SVM classifier which has excellent performance on pedestrian detection.

The Haar-like method is proposed by Papageorgiou (1998), which is a feature extraction method based on the grayscale of the image. The feature of the object is described according to the change of the grayscale between different pixel points in the image and it is usually used on face recognition.

2.4.3 Classifier

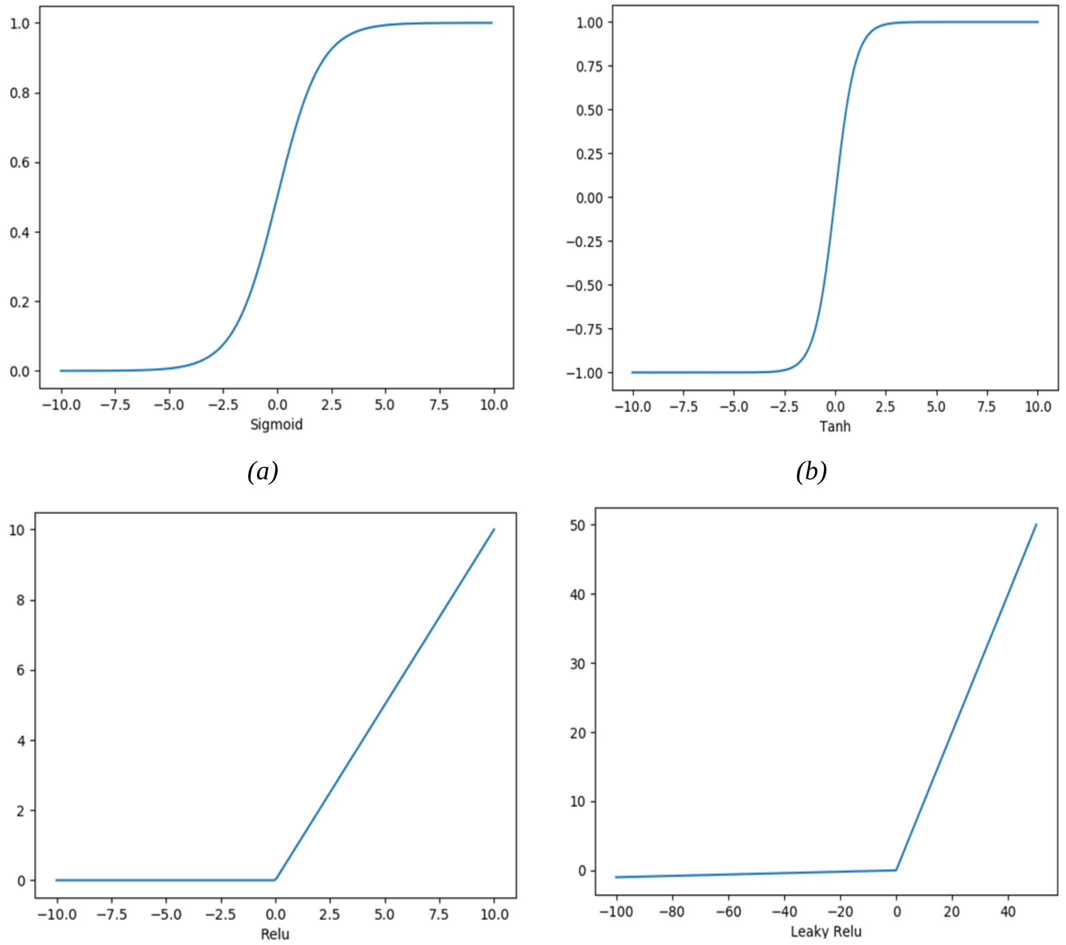
In object detection, there are many conventional classifiers with different emphases. Classifiers play an important role in distinguishing the object and the non-object and the selection of classifiers determines whether the object can be quickly and accurately distinguished according to the image features. Therefore, it is very important to select the appropriate classifier for the specific object detection application.

Common classifiers including Naive Bayes (Ravindran et al., 2013), decision tree (Quinlan, 1986), random forest (Tin Kam, 1995), Adaboost (Freund and Schapire, 1999), support vector machine (SVM) (Cortes and Vapnik, 1995), etc. Both of them have their own advantages and disadvantages in different application scenarios. For instance, The SVM is mainly used to classify the data sets, the method finds a partition hyperplane and separates the samples into different categories. It can be used for multi-class text classification, text recognition and segmentation and other scenarios. The decision tree method is based on the tree structure to determine the input data. and based on tree decision it processes the data and output the probability for the potential outcome. Usually, it used in business aspects such as stock market and lottery market to provide the recommendation to customers.

2.4.4 The Activation Function of the Neural Network

Activation function, also known as excitation function, is an important part of the neural network. It is used to perform a linear or nonlinear transformation on the input data. Generally, the only non-linear activation function is used in the neural network due to the final expression would turn to linear if the linear function is used which limited function expression. Therefore, it is necessary to use nonlinear activation function to function mapping to the data which could improve the expression ability of deep neural networks and it can theoretically approach any function.

Most neural networks are optimized by some form of gradient descent, and activation functions need to be differentiable. In addition, complex activation functions may cause some gradient disappearance or explosion problems. Therefore, neural networks tend to deploy several specific activation functions to avoid problems. Figure 2.7 demonstrates four common activation functions in the neural networks.



(a) *(b)*

(c) *(d)*

Figure 2.7 Common activation function (a) Sigmoid. (b) Tanh. (c) ReLU. (d) Leaky ReLU

1. Sigmond

The sigmoid function, also known as the logistic function, has an output range of (0,1) for any input. The equation is:

$$Sigmond(z) = \frac{1}{1+e^{-z}} \tag{2–1}$$

Sigmoid is very much like a smooth version of the step function.

It has many advantages:

1. It's non-linear.
2. Unlike binary output, sigmoid can input any value between 0 and 1 which can be used to represent the probability.
3. The output value of sigmoid is in a certain range, which means that it does not output infinite numbers.

However, the sigmoid activation function also has disadvantages, when the input value is infinity close to 1 or 0, the gradients disappear would happen which means the network stop learning from data. And another disadvantage of the sigmoid is it using the exponent arithmetic which is computation consuming.

2.Tanh Function

Tanh or hyperbolic tangent is another activation function that commonly used in deep neural networks. Similar to the sigmoid function, it also converts the input to an output range. For any input, the tanh will produce a value between -1 and 1. The equation is:

$$Tanh(z) = \frac{e^{z}-e^{-z}}{e^{z}+e^{-z}} \tag{2–2}$$

The tanh activation function is similar to the sigmoid function. It is a nonlinear function and the output is in a range, where it is (-1, 1). Also, it has the same drawbacks as sigmoid such as the problem of the gradient disappearing, and it also consumes large computation when performing exponent arithmetic.

3. ReLU Function

ReLU as the most popular activation function in CNN dut to it is efficient. When processing for any input is less than 0, the output would be 0 while other input value does not change. The equation is:

$$ReLU(z) = \max(0, z) \tag{2–3}$$

ReLU is a non-linear activation function and is faster than sigmoid and tanh (Krizhevsky et al., 2012). In addition, ReLU is known for avoiding the gradient vanishing problem. However, ReLU has a fatal flaw called "dying ReLU ". Dying ReLU is the permanent death of neurons in a network due to their inability to function in forwarding propagation. Specifically, this problem occurs when the activation function of neurons outputs zero in forwarding propagation, resulting in a zero gradient for its weight. Thus, when performing backpropagation, the weights of the neurons will never be updated and specific neurons will never be activated.

4. Leaky ReLU Function

In order to solve the phenomenon of dying ReLU, the Leaky ReLU is proposed, the output of this function has a small slope to the negative input. Since the derivative is always non-zero, this reduces the presence of dying neurons, allows gradient-based learning, and solves the problem of neurons not learning when the ReLU function enters a negative range.

2.5 Methods Based on Traditional Image Processing

Under the category of image processing, researchers have done a lot of research on the effectiveness of various algorithms for tunnel defect recognition. Some researchers use simple classical image processing methods such as image segmentation and morphological processing to process defects images, while others use complex theories such as wavelet transform, NSCT algorithm and machine learning based on mathematical theories. The research on various algorithms are summarized as follows:

Attard (2018) proposed a detection method that combining the difference of neighborhood image, binary pixel comparison and optical flow method, which can detect the defects of tunnels and predict the propagation of cracks.

Muduli (2013) designed an improved crack detection algorithm based on image fusion. For cracks image boundary detection, the improved canny operator and HBT were used to detect the boundary of the image contain defects. Then, Haar wavelet is used to decompose and the image fusion algorithm is used to the unified fusion for reduce the sensitivity of background noise, enhanced boundary effect and adapt to crack image features to improve the detection accuracy of cracks detection. It has a better performance compared with other boundary detection methods, however, when the image interfered by the noise, the detection accuracy of the algorithm would decrease.

Lee (2008) proposed an efficient recognition method based on computer vision for concrete bridge cracks. The median filter is used to filter the image noise, and the isolated noise points are filtered, then the morphological corrosion operation with multiple iterations is used to ensure the connectivity between the crack’s segmentation areas. Through this processing, the connected cracks information is obtained. The fracture tracking method is based on the 8 of the points of the pixel intensity in the neighbourhood to determine the position of the next pixel point according to the direction of the minimum intensity. This algorithm has better performance for single crack identification, but has poor performance for multiple cracks.

2.6 Deep Learning

With the development of artificial intelligence, the concept of deep learning is proposed. Deep learning is not an independent learning method (Hinton et al., 2012, Krizhevsky et al., 2012, LeCun et al., 2015), it also uses supervised and unsupervised learning methods to train the deep neural network. However, due to the rapid development of this field in recent years, some unique learning methods (such as the residual network) have been proposed (He et al., 2016), therefore more researchers regard it as a learning method alone.

The original deep learning was a learning process that used deep neural networks to solve the problem of feature expression. Deep neural network itself is not a new concept, which can be roughly recognized as a neural network structure containing multiple hidden layers. In order to optimize the training effect of the deep neural networks, the connection method and activation function of neurons need to adjust according to the specific tasks. There were many ideas in the early period during artificial intelligence, but due to insufficient training data and low computing capacity, the result could not satisfy the requirement. Figure 2.8 shows the relationship between artificial intelligence, machine learning and deep learning.

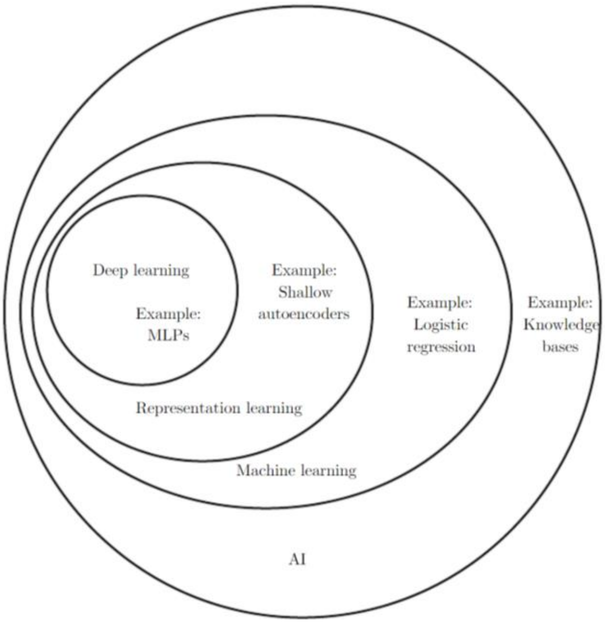
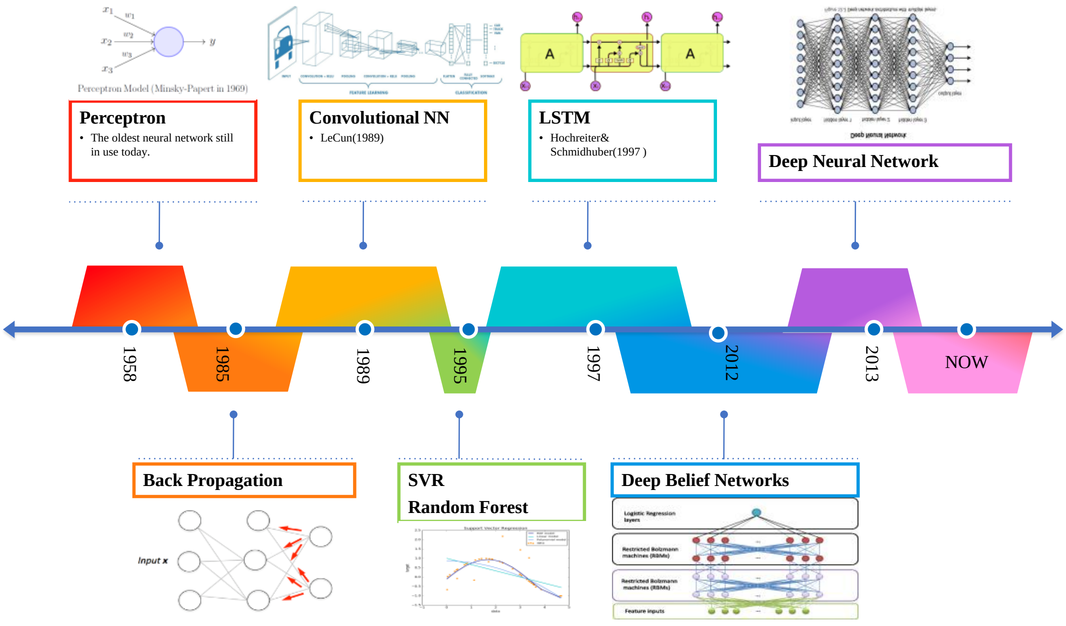


Figure 2.8 Relationship between artificial intelligence, machine learning and deep learning (Goodfellow et al., 2016)

Machine learning has been researched for many years and various model has been developed for multiple purposes. Minsky and Papert (1969) developed the first practical model Perceptron which is the oldest neural network (Schmidhuber, 1992). Recently, due to the development of computational power and artificial intelligence technology, representational learning methods represented by deep learning, have gradually emerged recent years (Bengio and Delalleau, 2011). Compared with traditional machine learning, deep learning has many advantages such as strong adaptability and required less feature extraction (could self-feature extraction). It has been successfully applied in many information technology aspects, such as image classification (Maggiori et al., 2017), speech recognition (Hinton et al., 2012), natural language processing (Yin et al., 2017) and the time sequence analysis (Guo et al., 2016). Although the architecture of deep neural networks is complex and widely used, most of them depend on some simple deep models, such as deep belief network (DBN) (Hinton et al., 2006), convolutional neural network (CNN) (LeCun et al., 2015), Generative Adversarial Networks (GAN) (Goodfellow et al., 2014) and recurrent neural network (RNN) (Graves et al., 2013). Figure 2.9 demonstrates the brief history of machine learning.



Perceptron Model (Minsky-Papert in 1969)
Perceptron
• The oldest neural network still in use today.
Convolutional NN
• LeCun(1989)
LSTM
• Hochreiter& Schmidhuber(1997)
Deep Neural Network
1958
1985
1989
1995
1997
2012
2013
NOW
Back Propagation
Input x
SVR
Random Forest
Deep Belief Networks
Logistic Regression layers
Restricted Boltzmann machines (RBMs)
Restricted Boltzmann machines (RBMs)
Feature inputs

Figure 2.9 Belief history of machine learning

In 2012, Alex Krizhevsky (2012) designed the AlexNet neural network based on the convolutional neural network and won the ILSVRC challenge in the lmageNet, and the deep learning convolutional neural network began to be valued.

CNN can iteratively learn feature expression from a large number of sample images. Its network units are divided into three categories, which are convolution layer, pooling layer and fully connected layer. The convolutional layer is used to extract image feature express. The pooling layer generally follows the convolutional layer and integrates the features extracted by the convolution in the previous layer, and extract the most representative features while reducing the parameters, in order to reduce the training time and enhance the network learning efficiency. Figure 2.10 shows the structure of CNN.

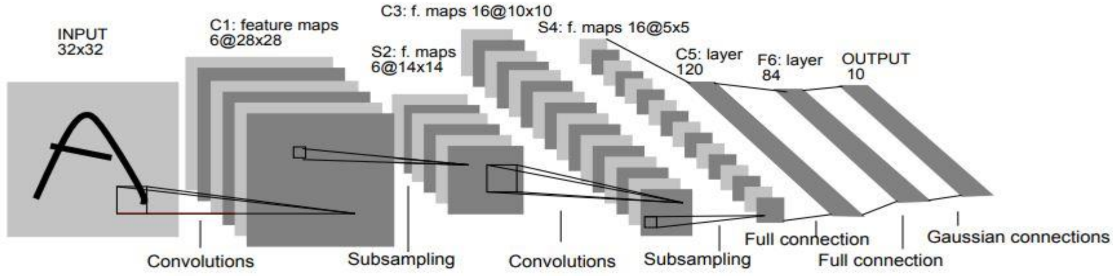


Figure 2.10 Structure of CNN (Lecun et al., 1998)

The fully connected layer is equivalent to the hidden layer in the artificial neural network and it is usually placed behind the convolution layer and the pooling layer to extract the features from the network for further analysed and obtain the corresponding concise expression, which acts as a "classifier" in the whole network. The CNN network structure (Hinton et al., 2012) has created three characteristics of CNN: Local receptive field, weight sharing and sub-sampling.

CNN is one of the most advanced deep learning methods at present, showing great potential in natural language information processing such as speech, image, video, and text (Collobert and Weston, 2008, LeCun et al., 2015). Since AlexNet (Krizhevsky et al., 2012) achieved outstanding results in the ImageNet competition in 2012, deep learning has attracted attention in the industry and has been used in all walks of life and demonstrated strong capabilities. In image processing, CNN networks such as Resnet (He et al., 2016) and U-Net (Graves, 2012, Ronneberger et al., 2015) have achieved significant results and successfully applied in areas such as security, transportation, medicine (Han and Ye, 2017), and finance (Sundermeyer et al., 2012) Figure 2.11 demonstrates the structure of Long-short term memory.

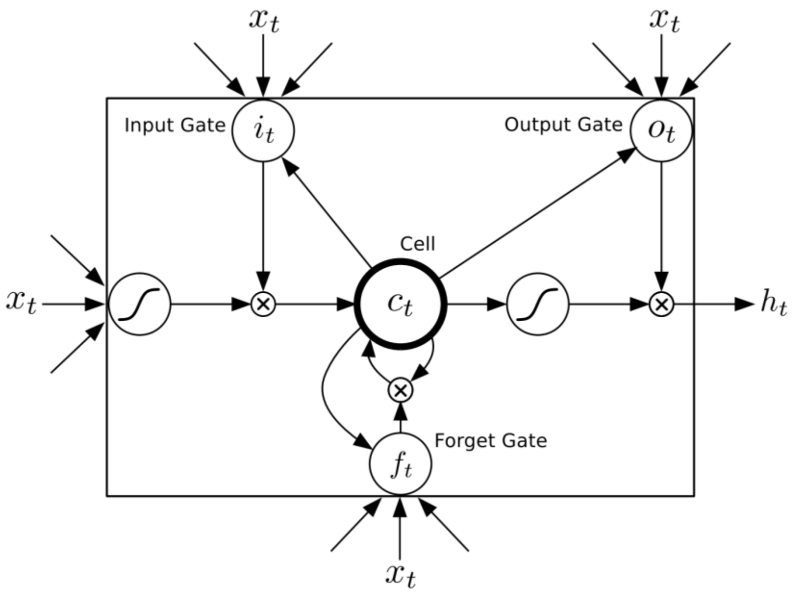


Figure 2.11 Structure of LSTM (Hochreiter and Schmidhuber, 1997)

LSTM is of variety Recurrent neural network (RNN) (Hochreiter and Schmidhuber, 1997) while it is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence. This allows it to exhibit temporal dynamic behaviour. Unlike feedforward neural networks, RNNs can use their internal state (memory) to process sequences of inputs. However, RNN faces the problem of gradient disappearance when training with backpropagation algorithms, which causes RNN to fail to capture long-term dependencies. LSTM changed the internal structure of RNN, introduced the door to control the input of information, and successfully solved the problem of RNN, which has become a mainstream architecture of RNN. Table 2.6 summarizes the different models of TBM operation parameters predictions by conventional machine learning.

Table 2.6 Different models of TBM operation parameters predictions by conventional machine learning

Method	Prediction	Parameters	Model	Reference
Deep Learning Based Method	Penetration rate	Mechanical data and Geological Data	DNN	(Koopialipoor et al., 2019)
	Penetration rate	Mechanical data and Geological Data	LSTM	(Gao et al., 2020)
	Torque, Thrust, Penetration rate	Mechanical data and Geological Data	RNN	(Gao et al., 2019)
	Field Penetration Index (FPI)	Mechanical data and Geological Data	DBN	(Chen et al., 2020)

Note. DNN= Deep Neural Network, LSTM= The Long Short-Term Memory, RNN= Recurrent neural network, DBN= Deep Belief Network.

Some researchers apply the deep learning based algorithms on TBM operating parameters prediction. Koopialipoor (2019) build deep neural networks (DNN) model to predict the penetration rate of the TBM, and the DNN model demonstrated better performance for penetration rate estimation compared with the ANN model and. Gao (2020) propose using LSTM to predict the penetration rate of TBM, and the network model has the better overall performance especially in the rapidly increasing period of the penetration rate compared with RNN based model and traditional time-series prediction model autoregressive integrated moving average with explanation variables (ARIMAX), the overall performance on proposed model is better. Gao (2019)using three different varieties of RNN (including traditional RNN, LSTM networks and gated recurrent unit (GRU) networks) to predict the TBM penetration rate, torque and thrust. The experimental results show that the proposed three kinds of RNN-based predictors can provide accurate prediction values of some important TBM operating parameters during the next period including the torque, the velocity, the thrust and the chamber pressure. Compared with the conventional methods, the application of the deep learning algorithm could utilize more data hence improved the accuracy of prediction.

2.7 Types of Object Detection Based on Deep Learning

Because of the conventional methods cannot identify the specific objects in the image quickly and accurately, the conventional image processing methods are not very practical in defect detection. With the development of computer science, object detection algorithms based on deep learning have been further developed, which makes it possible to detect TBM lining defects by computer vision.

Object detection is the process of recognizing and classifying a specific object in an image. Not only the category information of the object in the image needs to be obtained, but also the position information of the object should be pointed out. For the application of object detection on different objects, the shape, size, texture and other features of the object itself are different, and its position in the image is uncertain, so the detection performance on different objects is different.

The object detection based on the deep learning method has many advantages over the conventional detection method. For instance, it has powerful feature extraction and learning ability, also it could achieve automatic object detection without manual extraction of features. The convolution neural network with the specific structure can achieve fast detection speed and high accuracy, which draws lots of attention in the research field of object detection.

There are two main types of object detection algorithms based on deep learning. One is based on the two-stage detection algorithm, and another one is based on the one-stage detection algorithm. Specifically, the idea of the two stage detection method is to extract the bounding boxes that may contain objects at the initial stage (region proposal) and then conduct feature extraction and classification operations on these generated bounding boxes. R-CNN, Fast R-CNN, Faster R-CNN, etc (Girshick et al., 2014, Girshick, 2015, Ren et al., 2017) are object detection algorithm based on the two stage method.

The idea of the one stage detection method is to directly perform regression processing by the neural network on the entire input image, outputs the prediction box and category information of the corresponding object in the image. YOLO series algorithm and Single Shot MultiBox Detector (SSD) (Liu et al., 2016, Redmon et al., 2016, Redmon and Farhadi, 2017) series algorithm is a typical base on one stage end-to-end algorithm idea, which directly generate the bounding box of the objects in the image by using the regression method and get the category information at the same time, without using the method such as region proposal and sliding window to extract features, therefore it has faster detection speed.

In general, the two-stage object detection algorithm usually has a high accuracy rate, because its focus is on processing the extracted candidate box, and it will often get a more accurate detection box while the single-stage object detection algorithm is faster, due to its focus is to get the position and category information of the detection box by "looking once" the input image through regression operation.

2.8 Overview of Two Stage Object Detection Methods

Because the background and foreground of the image has significant differences, while the condition of detection is relatively complex. The image may contain multiple interference sources, which is not suitable for the recognition of the entire image. Hence, the object detection method is more suitable for defect recognition and locating.

At present, the object detection method based on deep learning can be divided into two ways. The first one is a two-stage algorithm represented by the R-CNN algorithm series. Those kinds of algorithms need to generate a large number of bounding boxes at the interested region during the initial stage, and then uses the detection branch network in the network structure to convolute and pooling the interested region. At the last stage, the classification network is used to identify the specific category of the object and the corresponding score value. Figure 2.12 shows the typical process of two stage algorithms.

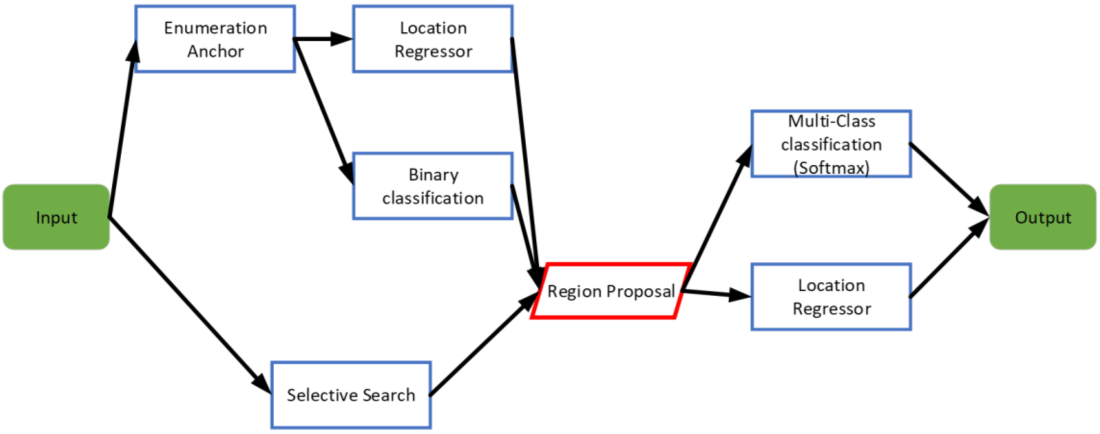


Figure 2.12 Typical two stage algorithm process

In general, the two-stage object detection algorithms are more focused on the accuracy of detection, while the other method is a one-stage end-to-end algorithm represented by the YOLO series, which more focused on the speed of detection.

2.8.1 R–CNN

In 2014, Girshick (2014) proposed the R-CNN algorithm, which surpassed the end-to-end method OverFeat algorithm proposed by Yann Lecun at the same period, and its algorithm structure became the classic structure of the subsequent two stages algorithm. Figure 2.13 shows the process of R-CNN.

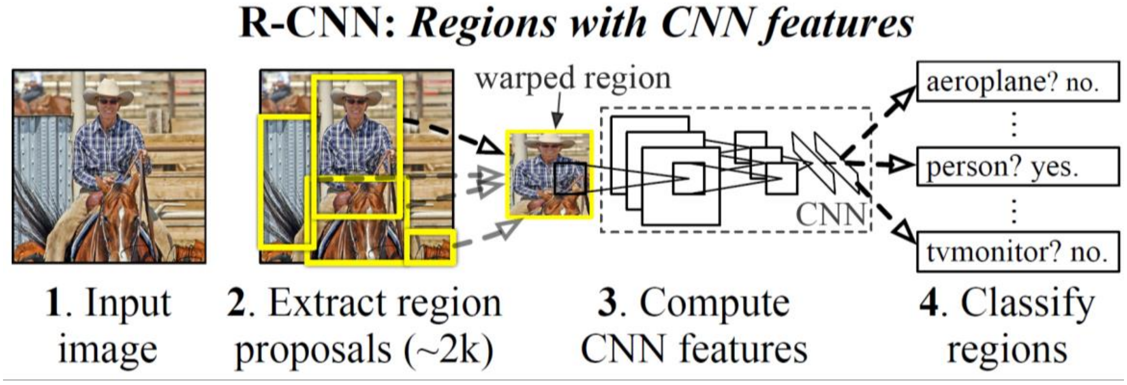


Figure 2.13 Process of R-CNN (Girshick et al., 2014)

R-CNN algorithm uses the selective search algorithm to evaluate the feature similarity of adjacent image subblocks, by scoring the similar image areas after merging, the candidate blocks of the region of interest are selected as a sample and input into the convolutional neural network structure, eigenvector is formed by positive and negative sample features composed of network learning candidate and calibration blocks, then the classifier designed by support vector machine is used to classify the eigenvector. Finally, the border regression operation is completed for the candidate and calibration blocks to achieve the object detection locating purpose. Although the R-CNN algorithm achieves 50% performance improvement compared with the traditional object detection algorithm, it has some drawbacks: The candidate blocks of positive and negative samples in the training network are generated by the conventional algorithm, which limits the speed of the algorithm. The convolutional neural network needs to carry out feature extraction for each generated candidate region at one time, which leads to a lot of repetitive operations, restrict the performance of the algorithm.

2.8.2 Spatial Pyramid Pooling Network (SPP-Net)

To address the problem of repetitive operation in R-CNN architecture, in 2015, He (2015) proposed an algorithm called SPP-Net, by adding a spatial pyramid pooling structure between the convolutional layer and the fully connected layer to replace method in the R-CNN algorithm which the candidate blocks were clipped and scaled to make the size of the image subblocks consistent before the input of the convolutional neural network. Figure 2.14 shows the structure of the spatial pyramid pooling layer.

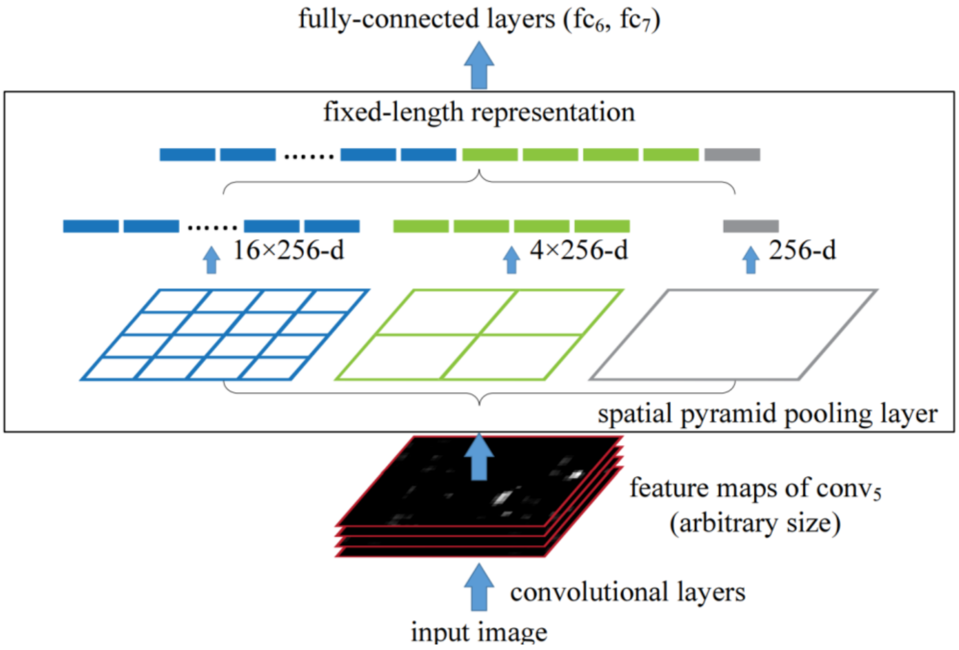

fully-connected layers (fc6, fc7)
fixed-length representation
16×256-d
4×256-d
256-d
spatial pyramid pooling layer
feature maps of conv5
(arbitrary size)
convolutional layers
input image

Figure 2.14 Spatial pyramid pooling layer (He et al., 2015)

The spatial pyramid pooling structure can effectively avoid the problem of incomplete clipping and shape distortion (He et al., 2015) caused by the R-CNN algorithm, more importantly, it solves the problem of repeated feature extraction of images by the convolutional neural network, greatly improves the speed of producing candidate blocks, and reduce the total amount of computation. However, the image size of the training data is not consistent with the R-CNN algorithm, which leads the candidate blocks to have a larger region of interest perception field, and the weight cannot be updated efficiently by using the backpropagation method.

2.8.3 Fast R-CNN

To solve the problem of the SPP-net algorithm, Girshick (2015) proposed an improved algorithm called Fast R-CNN in late 2015. Based on the SPP-net algorithm structure, a pooling layer structure of Region of Interest (ROI) pooling is designed, which effectively solves the problem that the R-CNN algorithm must cut and scale the image area to the same size. Figure 2.15 shows the process of the Fast R-CNN.

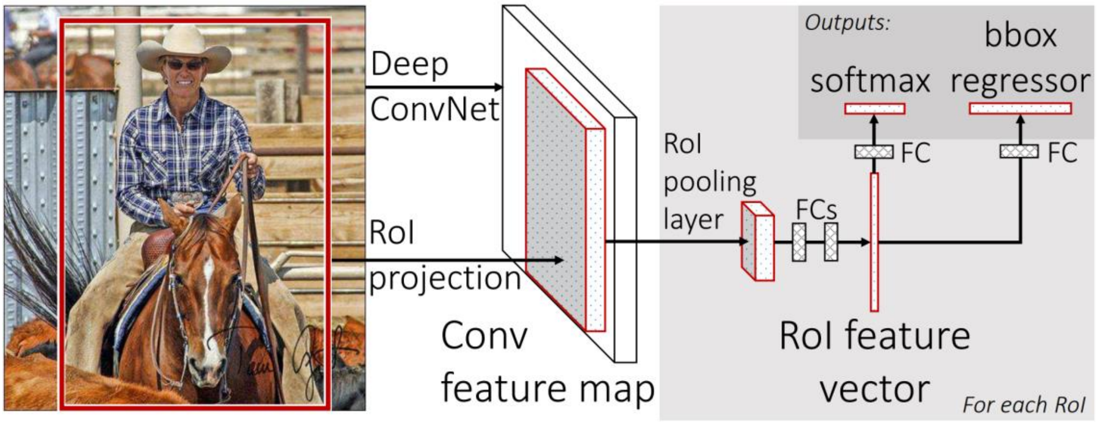


Figure 2.15 Process of Fast R-CNN (Girshick, 2015)

The algorithm proposes the idea of the multi-task loss function, combines classification loss and border regression loss with unified training and learning, and outputs corresponding classification and border coordinates. Moreover, instead of requiring additional disk space to store features of the middle layer, gradients can be propagated directly through the region of interest pooling layer. However, it still does not solve the problem of generating positive and negative sample candidate blocks by the selective search algorithm.

2.9 Overview of One Stage Object Detection Methods

Due to the existence of RPN in network structure, the two-stage method represented by the R-CNN algorithm, although it has higher detection accuracy, its speed has encountered a bottleneck, and it is difficult to meet the requirements of real-time detection for the specific situation. Therefore, an object detection algorithm of one stage based on the regression method was proposed. Figure 2.16 shows the typical process of one stage algorithm.

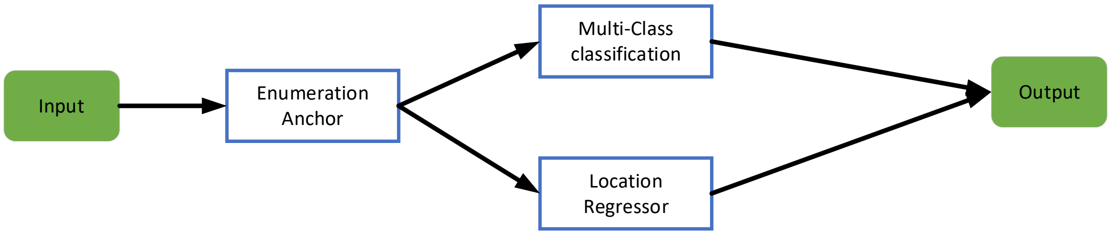


Figure 2.16 Typical one stage algorithm process

Different from the step-by-step shared detection feature training of the two-stage method, the one-stage method can achieve complete single training sharing features and the detection speed is greatly improved under the premise of ensuring a certain accuracy.

2.9.1 OverFeat

Yann Lecun (2014) came up with the famous OverFeat algorithm in 2013, it uses sliding windows and fixed size blocks to generate bounding boxes, with the multi-scale sliding window is used to increase the detection results to solve the problem of complex shape and the different size of objects in images. Finally, the convolution neural network and the regression model are used to classify and locate the object. Figure 2.17 shows the structure of OverFeat.

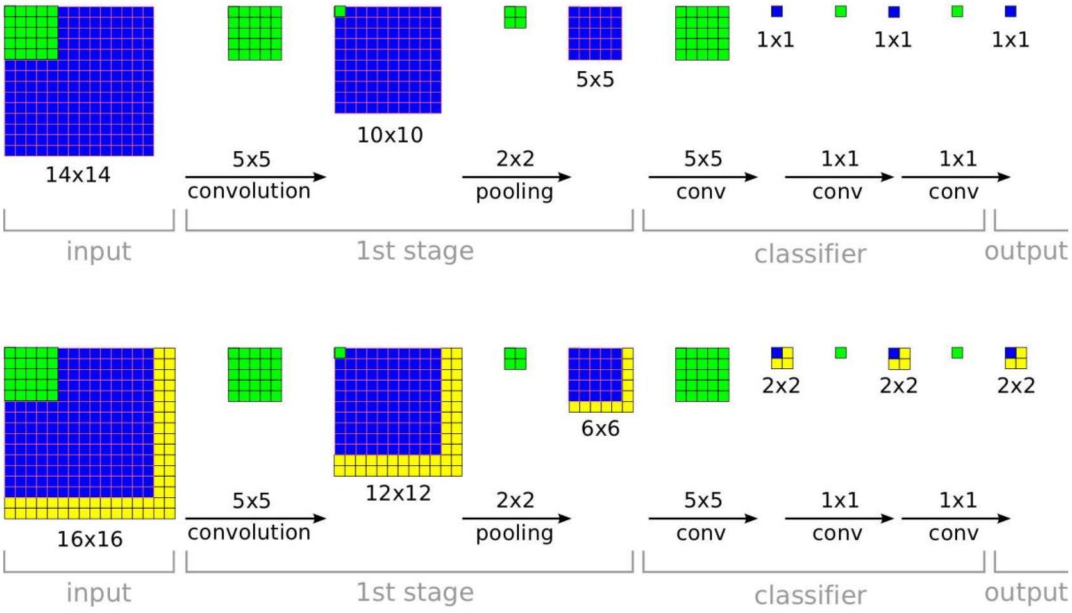


Figure 2.17 Structure of OverFeat (Sermanet et al., 2014)

For the first time, the object detection algorithm combines three computer vision tasks: classification, positioning and detection, won the title of ILSVRC 2013 task 3 (classification + positioning) (Sermanet et al., 2014) in the same year, however, it was soon replaced by the R-CNN algorithm.

2.9.2 You Only Look Once (YOLO)

The You Only Look Once (YOLO) algorithm proposed by Redmon (2016) is an one stage method based on regression which inherited the OverFeat algorithm, with detection speed could up to 45 frames per second, soon it became the state of art in end-to-end methods due to its speed advantage. Figure 2.18 shows the detection process of the YOLO algorithm.

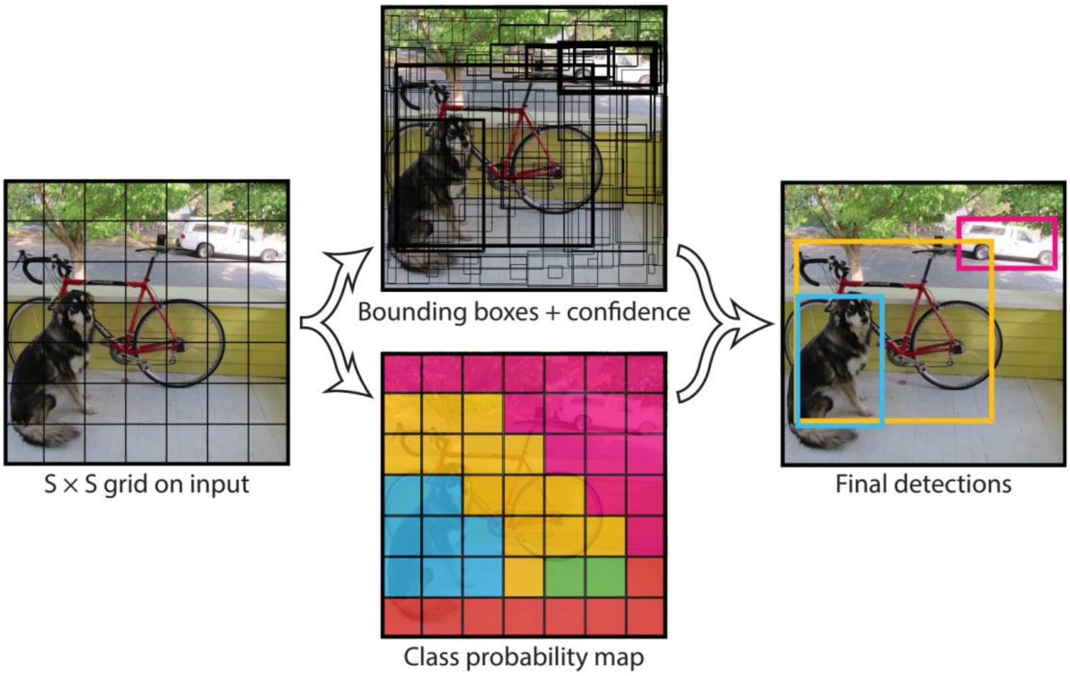


Figure 2.18 Detection process of YOLO algorithm (Redmon et al., 2016)

The YOLO algorithm makes predictions based on the global information of the image, and the overall network structure is uncomplicated. The input image was resized to a fixed size of 448×448 pixels, and the image was divided into a 7×7 grid area. Features were extracted by the convolutional neural network, and the border coordinates and confidence of each category in each grid were directly predicted. P-Relu activation function was used in the training. However, there are some shortcomings such as inaccurate positioning and recall rate, which are not as good as the region proposal method. Moreover, it has a poor detection effect for objects in close proximity and small objects, and its generalization ability is relatively weak.

For the first version of YOLO, the backbone network architecture was based on the GoogLeNet model, and some modifications have been made to it. It contains 24 convolutional layers and two fully connected layers, and it also uses the structure of the network in the network. Its detailed convolutional network structure is shown in Figure 2.19.

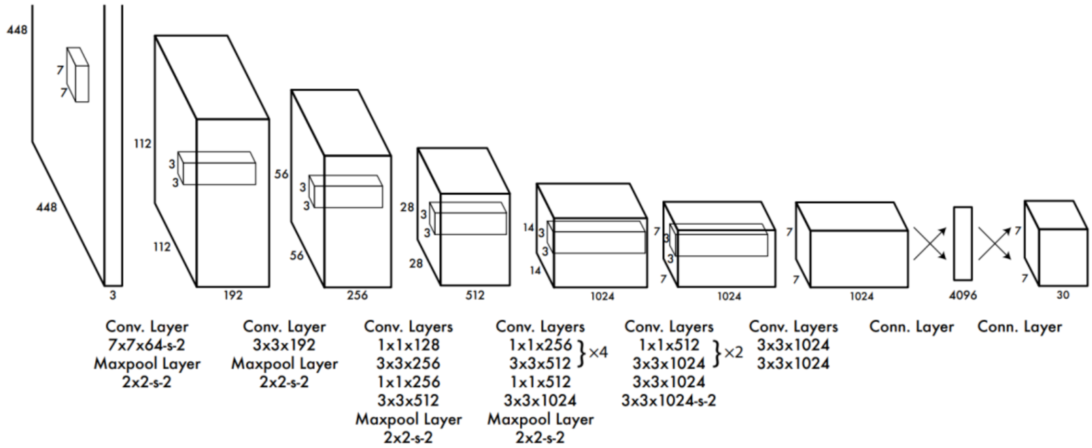


Figure 2.19 Convolutional network structure of YOLOv1 (Redmon et al., 2016)

2.9.3 Single Shot MultiBox Detector (SSD)

In order to solve the problem of the positioning accuracy for the YOLO algorithm, in December 2016, Liu (2016) proposed the Single Shot Detector (SSD) algorithm, which combined the regression idea from YOLO with the anchor box mechanism from Faster R-CNN. Figure 2.20 demonstrates the structure of the SSD.

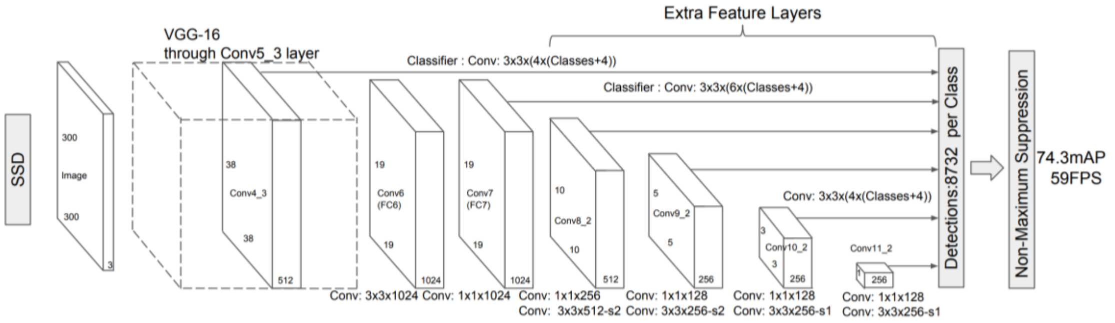


Figure 2.20 Structure of SSD (Liu et al., 2016)

By predicting the object region on the feature map of different convolutional layers, output discrete multi-scale and multi-ratio default box coordinates, and the small convolution kernel is used to predict the frame coordinate compensation of a series of candidate boxes and the confidence of each category.

In multi coordinates of the whole image, the local feature map of the multi-scale region is used for border regression, this ensures maintaining the fast characteristics of YOLO algorithm, and achieve similar positioning mechanism of Faster R-CNN. However, due to the use of multi-level feature classification, and the last convolution layer has a large range of sensing fields, thus the small target features are not obvious lead to it is difficult to detect small objects.

2.9.4 YOLOv2 & YOLO9000

After the improvement of Redmon and Farhadi (2017), YOLOv2 and YOLO9000 algorithms were proposed on CVPR in 2017 and were nominated for the best algorithm paper. The new version of the algorithm is focusing on solving the errors in recall rate and positioning accuracy. Darknet-19 was used as the backbone network, batch normalization pre-treatment was added, and fine-tuning after ImageNet pre-training model was trained with 224×224 and 448×448 in two stages. Figure 2.21 demonstrates the architecture of YOLOv2.

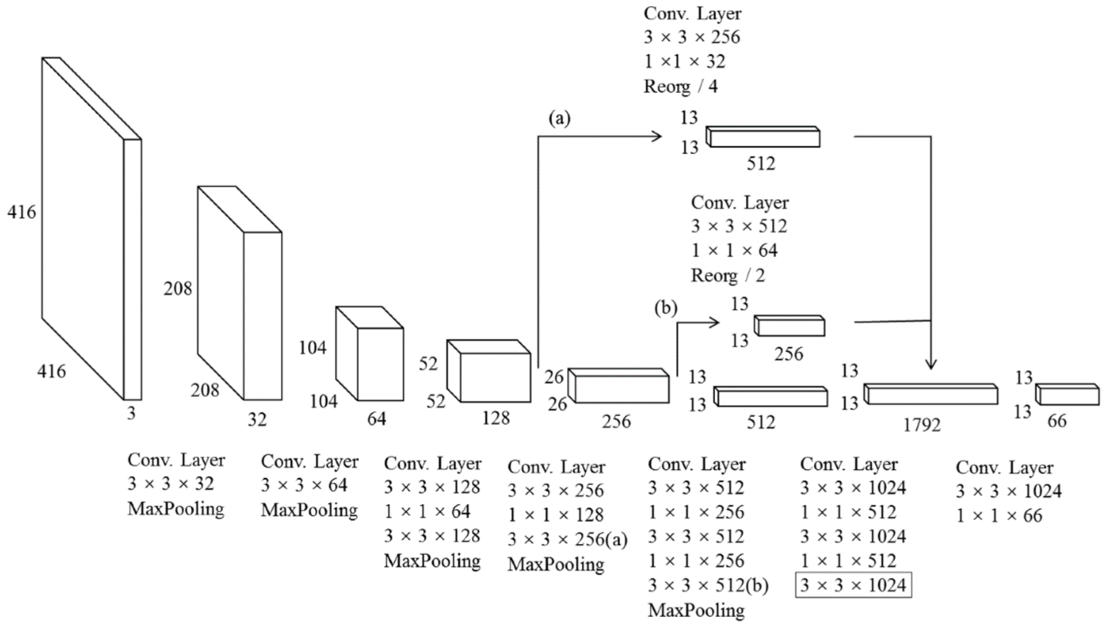


Figure 2.21 The architecture of YOLOv2 (Sang et al., 2018)

Compare with the original YOLO which use the full connection layer to directly predict the coordinates of the bounding box, YOLOv2 uses the idea of Faster R-CNN for reference, introduced the anchor mechanism and using the K - Means clustering approach in training to calculate a better anchor template. Moreover, it uses the anchor boxes operation in the convolution layer, increases the prediction of candidate boxes, and uses the strong constraint positioning method to improve the recall rate of the algorithm. Combined with the fine-grained feature of the image, the shallow feature is connected with the deep feature, which is helpful for the detection of small-sized targets.

2.10 Methods Based on Deep Learning Algorithm

The researchers introduced the idea of deep learning algorithms to deal with defects detection in the tunnel. However, the network with fewer hidden layers is often unable to extract enough image features, while the deeper network is easy to overfit. Network training and parameter tuning during deep learning will significantly affect the accuracy of detection performance.

Oullette (2004) proposed a method for fracture extraction based on the convolutional neural network. The classical Genetic Algorithm (GA) is used to make the model ignore the local minimum and converge to the global optimum. This method requires good data, sufficient training time, massive parameter calculation and adjustment.

Mallatl (2004) implemented the recognition of image texture features by using deep learning theory and achieved good classification performance. The scattering invariance algorithm of rotation, scale and deformation is used to provide a novel method for tunnel crack recognition.

Correia (2013) built a two-layer neural network. The first layer is using for classification and the second layer of the neural network is used to distinguish the cracks feature texture. A new crack marking method was designed to measure the crack width and compared it with the artificial observation results. Two networks are used to detect cracks and calculate widths respectively and achieve higher detection accuracy.

Matsuoka (2016) used the spectral clustering method to identify cracks based on the continuity features of cracks. Firstly, the Gabor filter operator is used to convolute the image and its directional energy is calculated. the average value of 8 directional convolutions is defined as the oriented energy. Then, the affinity matrix is calculated according to the information. The affinity matrix is described by the difference of adjacent pairs of pixels. Due to the continuity characteristics of cracks, the image cracks can be extracted from the affinity matrix.

2.11 Frameworks of Deep Learning

Several deep learning frameworks have been published by research institutions and companies, the open source framework could implement the algorithm of computer vision with less bug. With the frameworks, researchers could spend more time on parameters of the algorithm and optimize the framework according to different requirements, without repetitive code work, and these frameworks could satisfy most research requirements in most aspects of deep learning. Common dreamwork including TensorFlow, PyTorch, Keras, Theano, Deeplearning4j, Mxnet, etc. Table 2.7 shows the statistics of 2019 Github framework star and fork time.

Table 2.7 Summary of the statistics of 2019 GitHub framework star and fork (GitHub, 2020)

Framework	**Developer**	**Supported language**	**Stars**	**Forks**
TensorFlow	Google	Python/C+ +/Go/…	130540	75929
Caffe	BVLC	C++/Python	28495	17204
Keras	Google	Python	42504	16187
CNTK	Microsoft	C+ +	16258	4318
MxNet	DMLC	Python/C+ +/ R/…	17316	6145
PyTorch	Facebook	Lua /Python	29635	7167
Theano	U. Montreal	Python	8834	2493
PaddlePaddle	Baidu	python	9246	2484
Leaf	AutumnAI	Rust	4562	216
Lasagne	Lasgne	Python	2749	761
DL4J	Deeplearning4J	Java/Scala	10966	4697

1.PyTorch: PyTorch is a Python toolkit released by Facebook's AI team and is a deep learning tensor library using GPU and CPU optimizations. The framework formed after developer port the Torch develop kit based on Lua language to the Python language with better generality. It could replace the NumPy module and

GPU could speed up the computation of the tensors which generate by the torch if there is an appropriate GPU setting. Moreover, PyTorch has a simple user interface compared with other frameworks and due to its relatively newer code architecture, it has a relatively faster processing speed. Meanwhile, the PyTorch framework based on the Windows system can directly use the Python interface to conduct data training and testing, which is easy to operate and easy for beginners to get started, and the results are intuitive.

2. Theano is a kind of the first deep learning development framework. Theano started in 2007, just after the rise of the artificial intelligence wave in 2006. It was once regarded as the industry benchmark by developers and researchers. Theano uses Python's API interface, which is similar to the framework PyTorch (Torch is developed after Theano!). It can also efficiently define the network, optimize data calculation, computation based on GPU, and perform rapid iterative processing of mathematical expressions. Many early deep learning researchers and developers have still used the Theano framework.

3. Caffe: The lightweight open source deep learning framework for fast feature embedding developed by the University of California, Berkeley's vision and learning center (BVLC) is a complete deep learning kit that can be directly used for the build of convolutional neural network models, data training and testing, as well as cross-platform structural adjustment. Caffe means convolutional architecture for fast feature embedding.

Simple design, focus on CNN and computer vision are Caffe's obvious features. Caffe's concept is different from that of TensorFlow. Instead of the graph, Caffe uses a directed acyclic graph (DAG), which the graph-shaped network requires several components, including Blob, Layer and Net. Blob means "block" and serves as the basic unit for loading data at Caffe, and the input and output of a network are referred to by Blob. The layer is a single-layer network, which can be any one of the fully connected layers, convolutional layer and pooling layer. Different forms of network layers have different characteristics, which are completely designed and derived from the layer base class. Deep learning has forward and backward propagation algorithm, because the layer structure naturally requires forward and backward propagation operations. Net is a complete network model class in the Caffe framework, which is described as a combination of various layers.

Caffe has an efficient code process that comes from the native features of the C++ development language. Also, Caffe is powerful in GPU-based development. Its internal code is written by CUDA interface and developers develop multiple API based on other coding languages such as Python, MATLAB, etc. which also could achieve the same simplicity and efficiency.

4. TensorFlow: An artificial intelligence learning system developed by Google based on its internal deep learning development system, based on the data flow graph concept of tensor and flow, provides an excellent operating platform for machine learning personnel and deep learning researchers. The system builds a deep network model in the form of a static graph, which can be used for complex data processing. It has strong deployment capability in application scenarios, which can be deployed to the server side (supported by CPU or GPU devices). Even in some mobile apps, it can be used concisely, which reduces the cost of developing applications for companies and institutes the secondary development of users. Open source features are especially important for a wide range of users, this means that any individual or enterprise can use it to develop. Based on the extensive use of community influence, the TensorFlow framework is popular in a variety of application aspects, including speech, image recognition, etc.

5. Keras: Keras is an open source artificial neural network framework powered by python, which can be used as a high-level application program interface of TensorFlow, Microsoft CNTK and Theano to design, debug, evaluate, apply and visualize the deep learning model.

The code structure of Keras is written by the object-oriented method, which is completely modular and extensible. Its operation mechanism and instruction documents take user experience and difficulty into account and try to simplify the difficulty of the implementation of complex algorithms. Keras supports mainstream algorithms in the aspect of artificial intelligence, including neural networks with feedforward and recursive structures, and can also participate in the construction of statistical learning models through encapsulation. In terms of hardware and development environment, Keras supports multi-GPU parallel computing under multiple operating systems, which can be converted into components under TensorFlow, Microsoft-CNTK and other systems according to background Settings.

2.12 Summary

This chapter introduces the structure of TBM and its application in underground engineering, and reviewes the conventional method for TBM parameters prediction and object detection. Then, algorithms based on the deep learning concept are introduced, some commonly used activation functions and deep learning framework are briefly described, and their application scope, advantages, and disadvantages are pointed out. Based on existing literatures, the summaries are as follows:

1. The LSTM model is suited for TBM operating parameters prediction not only based on time series prediction but also could deal with a large volume of data.
2. For lining defects detection, the different algorithm has different structures, while both two stage based and one stage based algorithm has their advantages, disadvantages and application scenarios. Therefore, both R-CNN and YOLO could apply to lining defects detection.

Chapter 3: Prediction of TBM Operating Parameters

This chapter shows the experimental results of the proposed LSTM-based model of the prediction for TBM operating parameters. Due to the increase of TBM data volume, the traditional empirical method could not effectively use the large numbers of TBM data and conventional machine learning algorithms cannot effectively train the model with large numbers of data and unable to meet the requirements of fast and accurate prediction of TBM operating conditions. Therefore, the prediction model based on deep learning is proposed.

3.1 Long Short-Term Memory (LSTM) Model

LSTM is a variant of RNN. In RNN, there is a chain structure of repetitive neural network modules, which records the time change characteristics of data and transmits them to the next time node, thus the neural network has the memory function. However, due to the gradient descent algorithm adopted by the recurrent neural network, there are problems of gradient vanishing and gradient explosion. Therefore, Hochreiter (1997) proposed the LSTM neural network.

LSTM is an improved recurrent neural network, which is very suitable for dealing with the problems highly related to time series (Hochreiter and Schmidhuber, 1997). Compare with RNN, the LSTM can carry out long-term memory is mainly due to its unique cellular structure. The traditional RNN only has a very simple cell structure, such as a tanh layer, while LSTM has a unique cell structure, as shown in Figure 3.1.

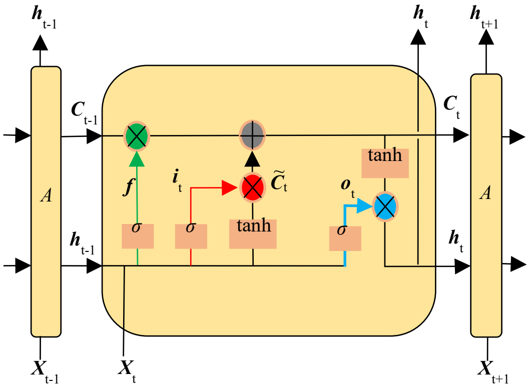


Figure 3.1 Structure diagram of Long Short-Term Memory cell

Figure 3.1 illustrates the detailed cell structure of LSTM. It can be seen that LSTM contains three gate structures and cell states: input gate (red module), forget gate (green module), output gate (blue module) and cell state (grey module). The three gate structures have different functions in LSTM structure:

Input gate: The input gate is used to update cell status. Initially, the information of the hidden state of the previous layer and the current input information is transferred to the specific function. The value is adjusted to a specific range (normally 0 to 1) to determine which information needs to be updated. If the value is closer to 0 which means it is less important. In another way, it is closer to 1 which means the value is important.

$$\boldsymbol{i}_{\mathrm{t}} = \sigma[\boldsymbol{W}_{\mathrm{i}} \cdot (\boldsymbol{h}_{\mathrm{t\text{-}1}}, \boldsymbol{x}_{\mathrm{t}}) + \boldsymbol{b}_{\mathrm{i}}] \tag{3–1}$$

Cell state: It is used to control the filtering and updating of information, and decide whether to keep information. Firstly, the cell state of the previous layer is multiplied by the forgetting vector point by point. If it is multiplied by a value close to zero, it means that the information needs to be discarded in the new cellular state. Then the value is added to the output value of the input gate point by point, and the new information generated by the neural network is updated to the cell state. At this point, the updated cell state is obtained.

$$\tilde{\boldsymbol{C}}_{\mathrm{t}} = \tanh[\boldsymbol{W}_{\mathrm{C}} \cdot (\boldsymbol{h}_{\mathrm{t\text{-}1}}, \boldsymbol{x}_{\mathrm{t}}) + \boldsymbol{b}_{\mathrm{C}}] \tag{3–2}$$

$$\boldsymbol{C}_{\mathrm{t}} = \boldsymbol{f}_{\mathrm{t}} \times \boldsymbol{C}_{\mathrm{t\text{-}1}} + \boldsymbol{i}_{\mathrm{t}} \times \tilde{\boldsymbol{C}}_{\mathrm{t}} \tag{3–3}$$

Forget gate: The function of the forget gate is to decide which information should be discarded or retained, and selectively forget the information which has less influence on prediction parameters in cells through specific functions.

$$\boldsymbol{f}_{\mathrm{t}} = \sigma[W_{\mathrm{f}} \cdot (\boldsymbol{h}_{\mathrm{t-1}}, \boldsymbol{x}_{\mathrm{t}}) + \boldsymbol{b}_{\mathrm{f}}] \tag{3–4}$$

Output gate: The output gate is used to determine the value of the next hidden state, therefore the final output includes both cell state and input, and its results are updated to the next hidden layer.

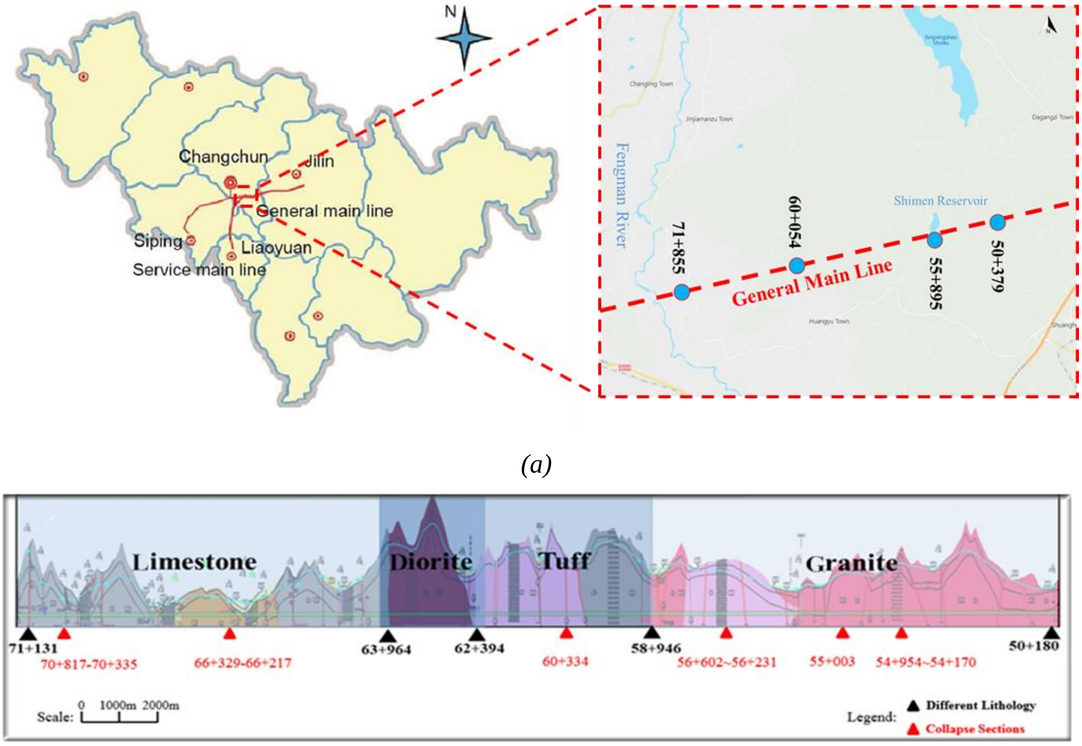
$$\boldsymbol{o}_{\mathrm{t}} = \sigma[W_{\mathrm{o}} \cdot (\boldsymbol{h}_{\mathrm{t-1}}, \boldsymbol{x}_{\mathrm{t}}) + \boldsymbol{b}_{\mathrm{o}}] \tag{3–5}$$

$$\boldsymbol{h}_{\mathrm{t}} = \boldsymbol{o}_{\mathrm{t}} \times \tanh(\boldsymbol{C}_{\mathrm{t}}) \tag{3–6}$$

In the equation: $\boldsymbol{W}_{\mathrm{i}}$ is the connection weight of the input gate; $\boldsymbol{b}_{\mathrm{i}}$ is the offset factor of the input gate; $\boldsymbol{W}_{\mathrm{C}}$ is the connection weight of cell state; $\boldsymbol{b}_{\mathrm{C}}$ is the offset coefficient of cell state; $\boldsymbol{W}_{\mathrm{o}}$ is the connection weight of the output gate; $\boldsymbol{b}_{\mathrm{o}}$ is the offset factor of the output gate; $\boldsymbol{h}_{\mathrm{t-1}}$ it is the output information of the cell state at the previous stage; $\boldsymbol{x}_{\mathrm{t}}$ is the input information of the current stage; σ is the activation function; $\boldsymbol{f}_{\mathrm{t}}$ is the output information of the forget gate; $\boldsymbol{i}_{\mathrm{t}}$ is the output information of the input gate; $\boldsymbol{C}_{\mathrm{t}}$ is the current cell state; $\boldsymbol{C}_{\mathrm{t-1}}$ is the previous cell state; $\boldsymbol{o}_{\mathrm{t}}$ is the output information of input gate; $\boldsymbol{h}_{\mathrm{t}}$ is the current output information of cell state. The optimal individual is obtained by calculating through the network, and the initial weights and thresholds of the network will be updated. After training, the network outputs the prediction results.

3.2 Project Background and Data

The data used in this thesis are all from the Jilin Yinsong Water Supply Tunnel Project. The Yinsong water supply project of the central city in Jilin Province is a self-flow tunnel with a maximum of 1.04 billion m^3 annual flow rate. The total length of the tunnel is 69.855km, and the frequent depth of the tunnel is from 50 to 100 m, while the maximum buried depth is 260 m. The construction method is using the TBM as the main method while drilling and blasting are for the auxiliary method. The datasets are provided by the China Railway Engineering Equipment Group (CREG) TBM cloud management platform, which collected and stored all data during the TBM excavation period from 2015.08 to 2017.09 including 199 parameters and up to 1 billion data. Figure 3.2 demonstrates the location and geology maps of the Yinsong project.



(a)

(b)

Figure 3.2 Location and geology maps of the study area: (a) Location of the Yinsong Project, (b) longitudinal section of the study area

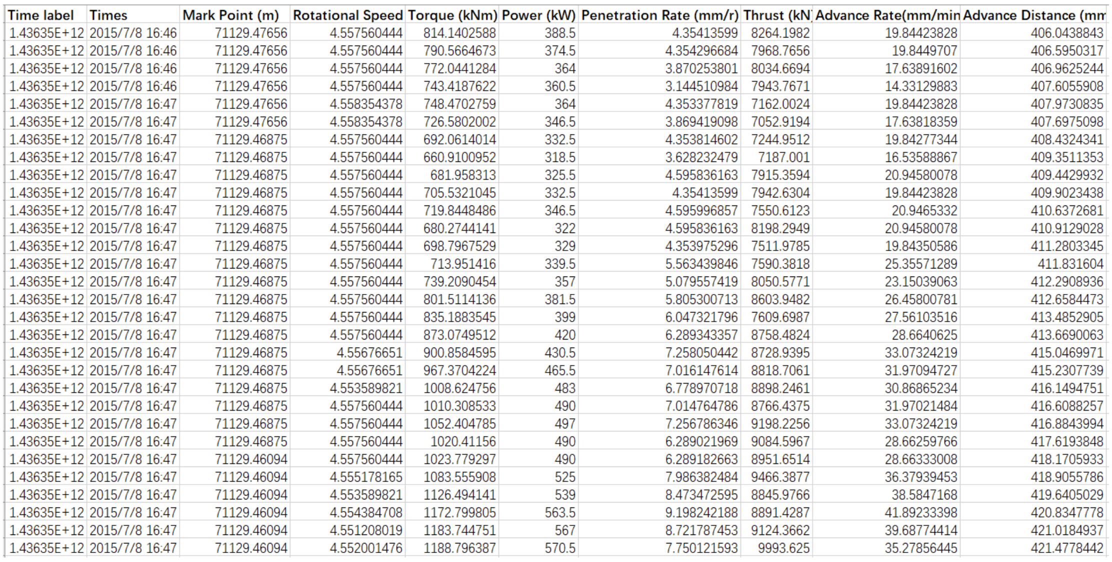
The research data in this thesis are from TBM3 (From Fengman reservoir to Yinma River) of the fourth section of the main trunk line. The crossing rock strata of the project can be divided into limestone and granite, with the starting mileage of 71 + 476 and the ending mileage of 51 + 705, with a total length of 19771m, including the length of drilling and blasting section of 2283m and the length of TBM excavation section of 17488m. The starting and ending time of TBM excavation is 31 months, which consists of 928 days. After deducting the maintenance and shutdown time in adverse geological conditions, the effective driving days of TBM are 728 days.

The TBM of this project is manufactured by the CREG. The specification of TBM is shown in table 3.1.

Table 3.1 Specification of Yinsong Project "Yongji" TBM

Parameters	**Specification**
TBM Diameter(mm)	8,030
Number of cutters	56
Cutter spacing (mm)	84
Cutter diameter (mm)	483
Maximum thrust (kN)	23,260
Maximum torque (breakout torque) (kN·m):	12,615
Cutterhead power (kW)	3,500
Cutterhead Rotation Speed (RPM)	0-7.6
Boring stroke (m)	1.8

During the excavation period, sensors are installed on the TBM to record each second operating data of TBM (e.g., thrust, torque and speed.) for analysis. The example of record data sets is shown in Figure 3.3.



Time label	Times	Mark Point (m)	Rotational Speed	Torque (kNm)	Power (kW)	Penetration Rate (mm/r)	Thrust (kN	Advance Rate(mm/min	Advance Distance (mm
1.43635E+12	2015/7/8 16:46	71129.47656	4.557560444	814.1402588	388.5	4.35413599	8264.1982	19.84423828	406.0438843
1.43635E+12	2015/7/8 16:46	71129.47656	4.557560444	790.5664673	374.5	4.354296684	7968.7656	19.8449707	406.5950317
1.43635E+12	2015/7/8 16:46	71129.47656	4.557560444	772.0441284	364	3.870253801	8034.6694	17.63891602	406.9625244
1.43635E+12	2015/7/8 16:46	71129.47656	4.557560444	743.4187622	360.5	3.144510984	7943.7671	14.33129883	407.6055908
1.43635E+12	2015/7/8 16:47	71129.47656	4.558354378	748.4702759	364	4.353377819	7162.0024	19.84423828	407.9730835
1.43635E+12	2015/7/8 16:47	71129.47656	4.558354378	726.5802002	346.5	3.869419098	7052.9194	17.63818359	407.6975098
1.43635E+12	2015/7/8 16:47	71129.46875	4.557560444	692.0614014	332.5	4.353814602	7244.9512	19.84277344	408.4324341
1.43635E+12	2015/7/8 16:47	71129.46875	4.557560444	660.9100952	318.5	3.628232479	7187.001	16.53588867	409.3511353
1.43635E+12	2015/7/8 16:47	71129.46875	4.557560444	681.958313	325.5	4.595836163	7915.3594	20.94580078	409.4429932
1.43635E+12	2015/7/8 16:47	71129.46875	4.557560444	705.5321045	332.5	4.35413599	7942.6304	19.84423828	409.9023438
1.43635E+12	2015/7/8 16:47	71129.46875	4.557560444	719.8448486	346.5	4.595996857	7550.6123	20.9465332	410.6372681
1.43635E+12	2015/7/8 16:47	71129.46875	4.557560444	680.2744141	322	4.595836163	8198.2949	20.94580078	410.9129028
1.43635E+12	2015/7/8 16:47	71129.46875	4.557560444	698.7967529	329	4.353975296	7511.9785	19.84350586	411.2803345
1.43635E+12	2015/7/8 16:47	71129.46875	4.557560444	713.951416	339.5	5.563439846	7590.3818	25.35571289	411.831604
1.43635E+12	2015/7/8 16:47	71129.46875	4.557560444	739.2090454	357	5.079557419	8050.5771	23.15039063	412.2908936
1.43635E+12	2015/7/8 16:47	71129.46875	4.557560444	801.5114136	381.5	5.805300713	8603.9482	26.45800781	412.6584473
1.43635E+12	2015/7/8 16:47	71129.46875	4.557560444	835.1883545	399	6.047321796	7609.6987	27.56103516	413.4852905
1.43635E+12	2015/7/8 16:47	71129.46875	4.557560444	873.0749512	420	6.289343357	8758.4824	28.6640625	413.6690063
1.43635E+12	2015/7/8 16:47	71129.46875	4.55676651	900.8584595	430.5	7.258050442	8728.9395	33.07324219	415.0469971
1.43635E+12	2015/7/8 16:47	71129.46875	4.55676651	967.3704224	465.5	7.016147614	8818.7061	31.97094727	415.2307739
1.43635E+12	2015/7/8 16:47	71129.46875	4.553589821	1008.624756	483	6.778970718	8898.2461	30.86865234	416.1494751
1.43635E+12	2015/7/8 16:47	71129.46875	4.557560444	1010.308533	490	7.014764786	8766.4375	31.97021484	416.6088257
1.43635E+12	2015/7/8 16:47	71129.46875	4.557560444	1052.404785	497	7.256786346	9198.2256	33.07324219	416.8843994
1.43635E+12	2015/7/8 16:47	71129.46875	4.557560444	1020.41156	490	6.289021969	9084.5967	28.66259766	417.6193848
1.43635E+12	2015/7/8 16:47	71129.46094	4.557560444	1023.779297	490	6.289182663	8951.6514	28.66333008	418.1705933
1.43635E+12	2015/7/8 16:47	71129.46094	4.555178165	1083.555908	525	7.986382484	9466.3877	36.37939453	418.9055786
1.43635E+12	2015/7/8 16:47	71129.46094	4.553589821	1126.494141	539	8.473472595	8845.9766	38.5847168	419.6405029
1.43635E+12	2015/7/8 16:47	71129.46094	4.554384708	1172.799805	563.5	9.198242188	8891.4287	41.89233398	420.8347778
1.43635E+12	2015/7/8 16:47	71129.46094	4.551208019	1183.744751	567	8.721787453	9124.3662	39.68774414	421.0184937
1.43635E+12	2015/7/8 16:47	71129.46094	4.552001476	1188.796387	570.5	7.750121593	9993.625	35.27856445	421.4778442

Figure 3.3 Example of normal excavation period record data

As shown in Figure 3.3, the data sets contain more than millions of records. The raw data of TBM records are massive and unclassified. Therefore, eliminating invalid data from datasets that consist of billions of records of data and properly classify data into different groups and purpose is a fundamental work for TBM machine learning. Figure 3.4 illustrates the composition of raw data.

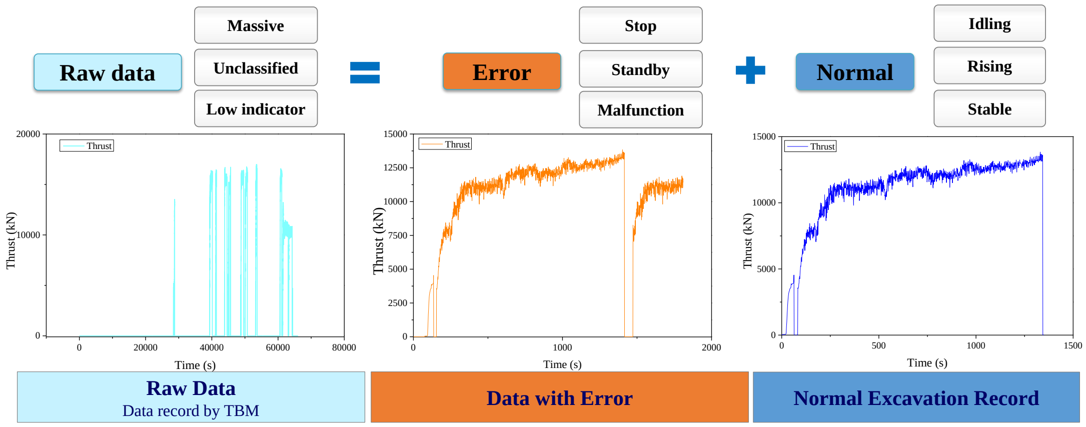


Figure 3.4 Composition of raw data

Generally, the raw data of TBM records can be divided into the following categories:

(1) Useless data including TBM malfunction and standby period.

(2) Normal excavation period data. The normal excavation period means that the excavation process during this period is steady with high efficiency. That kind of data would screen out as training samples for neural network model training.

(3) Extraordinary excavation period data. After manually remove the useless data and normal excavation period data, the remaining data can be divided into the following categories:

1. The advance movement of TBM is less than 0.1 meters.
2. The excavation process is neither steady nor high efficiency, the potential reason is the operator is inexperienced.
3. Due to mechanical defects and unexpected geological conditions, the excavation process is relatively unstable with lower efficiency.
4. The excavation period with major geological defects while excavation is difficult and potential accidents may occur.

This thesis focused on the TBM operating performance prediction during the stable excavation period, therefore, the normal excavation period data are separated out for training the model.

3.2.1 Partition Normal Excavation Period Data

The excavation period data including three periods, the idling period, rising period, and stable excavation period. This study would focus on TBM performance and rock mass parameter prediction of a stable period. Therefore, the partition of different period data is necessary. The criterion of partition the data would be the advance speed and TBM continuously steady operating more than 500 seconds could recognize stable period. Figure 3.5 illustrates the partition of the normal excavation period.

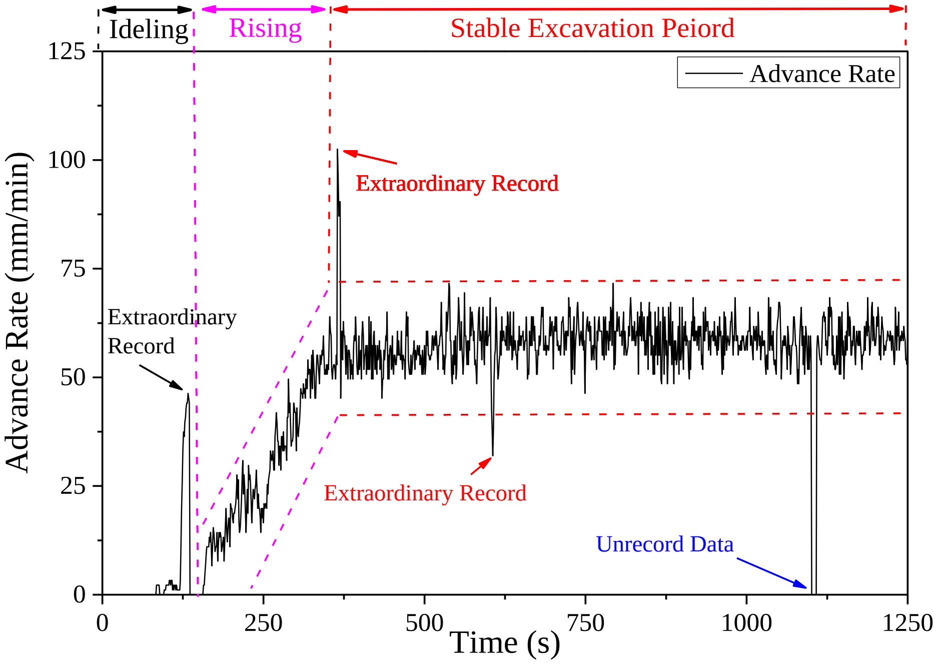
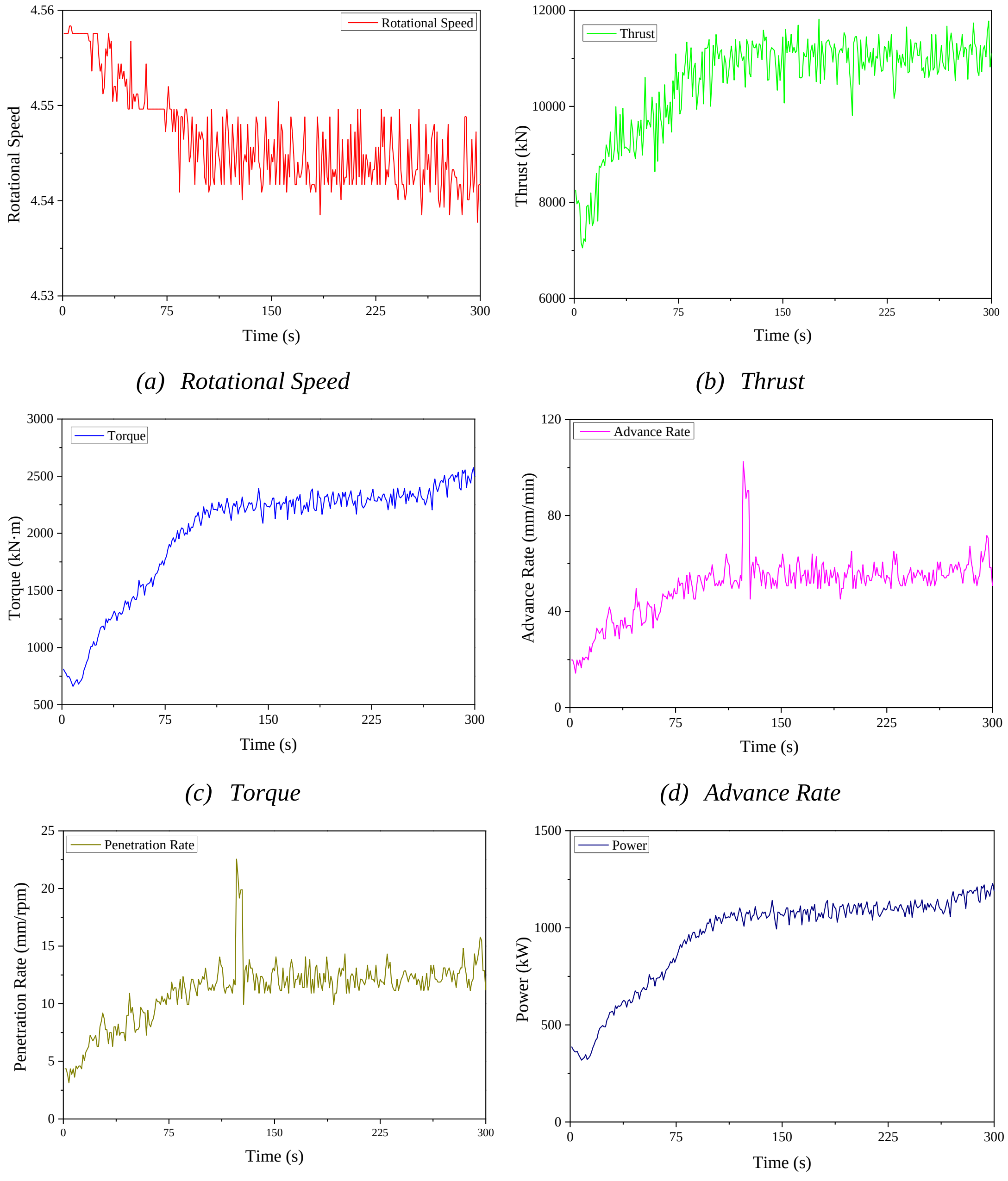


Figure 3.5 Partition of normal excavation period

3.2.2 Extraordinary Data of Advance Speed Correction.

The analysis of the TBM operating parameters indicates that there is some abnormal value during the excavation period. Those extraordinary values must be processed otherwise it will seriously affect the subsequent data analysis.

Generally, the deviation of advance speed should not exceed 5 times of the average. However, in the datasets, there is some extremely high value that TBM could not achieve. According to the manufacturer CREG’s engineering report, this phenomenon may cause by fault sensor measurement. The value of the abnormal will be replaced with a previous normal record manually. Moreover, during the excavation, there are few data are missing. The reason for the missing may cause by the failure of data collection or storage under specific situations, such as data storage failure, memory corruption, transmission failure and so on. The missing values are processed by linear interpolation. Figure 3.6 demonstrates the processed TBM record data.



(a) Rotational Speed

(b) Thrust

(c) Torque

(d) Advance Rate

(e) Penetration Rate

(f) Power

Figure 3.6 Record data of TBM during stable excavation period

3.3 Model Selection

3.3.1 Prediction Model

In this study, the LSTM neural network model is used to predict TBM parameters. Compare with backpropagation neural network (BPNN) and traditional RNN, when the BPNN model is used to predict a set of data, due to the back-propagation characteristic, it does not consider the impact of previous data sets, while RNN has better performance in dealing with short-term dependence problems. However, once the data volume increases, the gradient disappears phenomenon may occur (could not update the data to the cells). LTSM is a time-recursive neural network. It is suitable for processing and predicting important events with relatively long intervals and delays in time series. Also, it can learn faster when dealing with time series data.

Figure 3.7 shows the structure of the model, which contains three parts, the input layer, hidden layer and output layer.

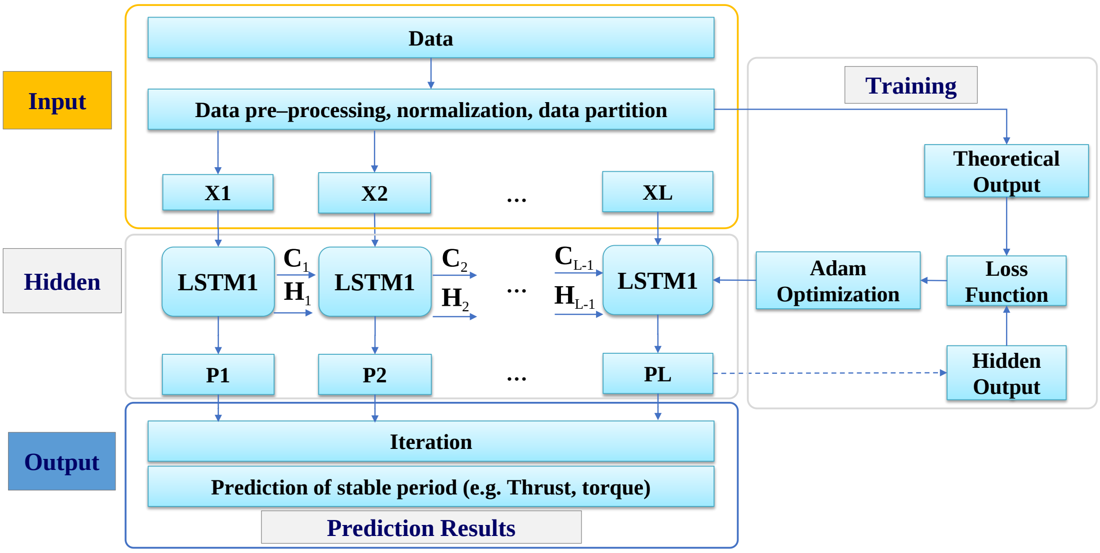


Figure 3.7 LSTM Structure (X: Parameters. C: Cell status. H: Hidden layer. P: Hidden layer output.)

3.3.2 Input Data and Output Prediction

The data in neural network could roughly divide into two part, the input data and output prediction. Generally, the input data is the data that currently recorded known data, and the output prediction is the data generated by the neural network after input the known data. In this thesis, the input data can be dividing into two parts. One is TBM operating parameters collected during excavation, another group is the rock mass parameters. Six categories of TBM operation data are selected as input data for train the proposed model which are Torque (kN·m), Thrust (kN), Power (kW), Penetration (mm/rpm), PR (mm/min) and RPM and five categories of rock data are selected as input data which are Uniaxial Compressive Strength (UCS), Brittleness Index (BI), Brazilian Tensile Strength (BTS) and Distance between Planes of Weakness (DPW) and orientation of discontinuities (α). And output prediction is one of the torque, thrust, penetration rate or advance rate of TBM operating parameters. The category of output prediction will not use as input data. For instance, if the output prediction of neural network is torque, the input data would consist the rock data and TBM operation data exclude the torque. Figure 3.8 demonstrate the input data and output prediction of the neural network.

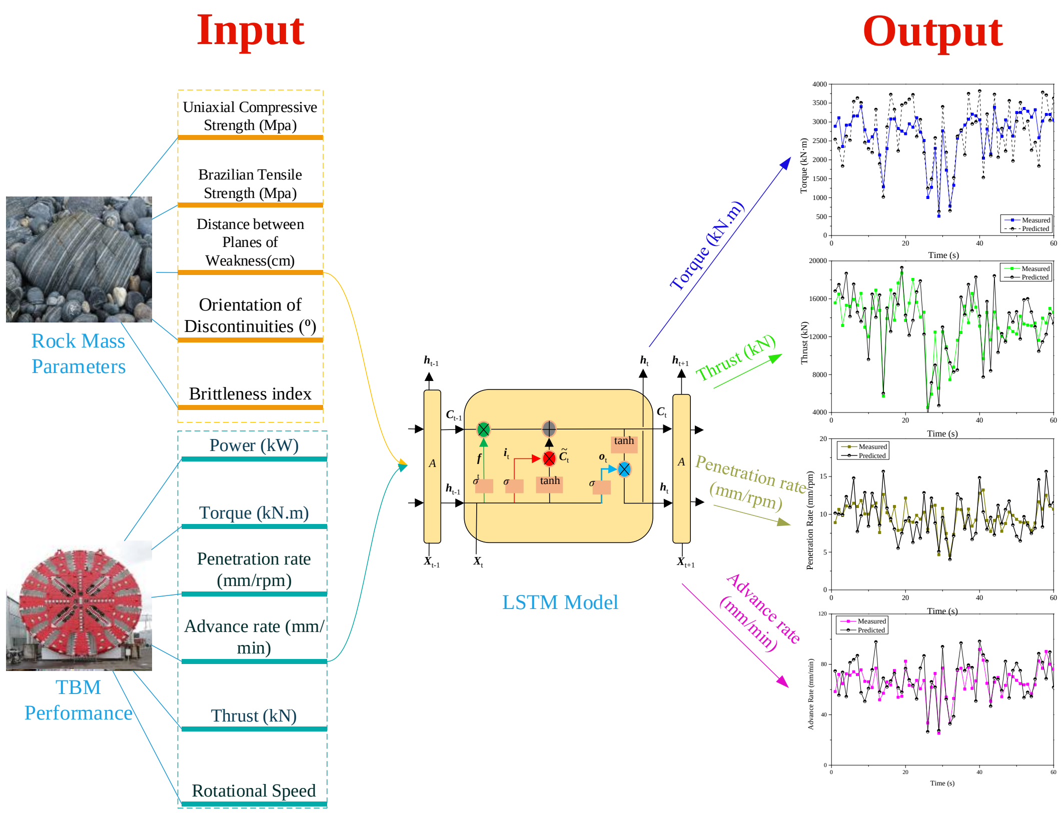


Figure 3.8 Input data and output prediction structure of the neural network

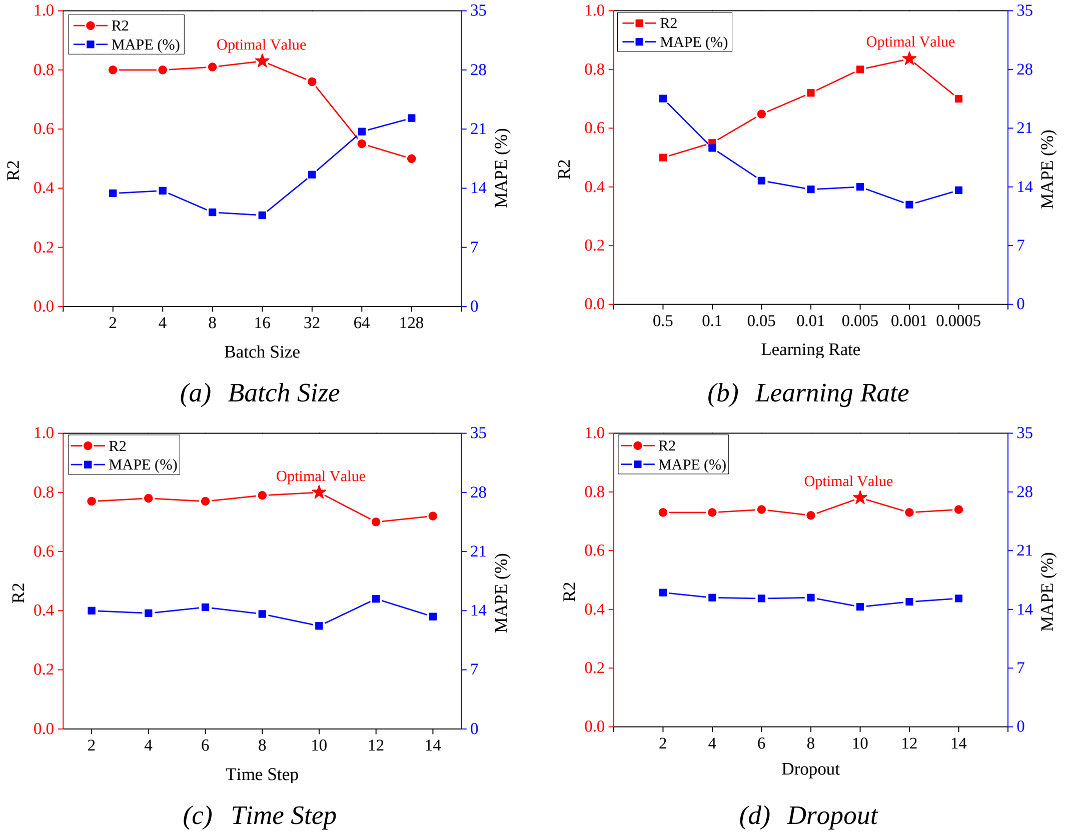
1800 different stable excavation period is selected for training the model, and the input data are dividing into three sets, one is the training set with 1440 period to training the neural network model while others are validation set with 360 periods. The purpose of the validation set is to improve the model to achieve higher accuracy by examining the prediction results at the end of each training period. The test set is used to evaluate the prediction accuracy of the model after the entire training period ends, while the data of the test set will not feed into the model during the training period.

3.3.3 Model Parameter Setting

Before the model training of deep learning, the software environmental selection and construction of the scientific research platform should be considered. The Windows system has a higher versatility of command scripts programs and there are a large number of supporting programs, so it does not need to be familiar with the new compiling environment, therefore it is very suitable for doing some research experiments related to the field of deep learning, thus the Python language in the Windows operating system is chosen to build the deep learning platform for scientific research.

Specifically, the deep learning platform used for object detection of this thesis is Windows 10, Python 3.7.4, Keras 2.13, platform hardware configuration is CPU Intel i7 5960X, 64GB of memory, configuration for graphics cards consist two NVIDIA GTX1080Ti, 11GB of video memory, while CUDA10.0 and NVIDIA cuDNN7.4.2 for GPU acceleration. Considering the graphic card of the deep learning platform is not designed for large sample deep learning, to avoid the phenomenon of out of memory, the network structure parameters of LSTM have adjusted appropriately, hence insufficient video memory would not be caused by a large number of samples during the period of the training model.

The hyperparameters are the parameters used to modify the algorithm when the model is established, and these parameters will not be changed in the process of neural network training. Before training the models, the values of hyperparameters need to be assigned. The hyperparameter setting of the model is critical during neural network training. Proper model parameter setting could not improve the prediction accuracy, but also reduce the training time which could improve the efficiency. Through multiple pre-training, the optimal value of super parameters can be determined. Figure 3.9 shows the results of hyperparameters pre-training.



(a) Batch Size

(b) Learning Rate

(c) Time Step

(d) Dropout

Figure 3.9 Results of hyperparameters pre-training

The results of the hyperparameters pre-training process indicate the optimal value for the model. Moreover, there are some optimization function could apply to the network to achieve specific tasks. The specific model parameter settings are shown in Table 3.2:

Table 3.2 Summary of neural network training environment

Model parameter setting:	**Dataset:**
Batch size: 16 Optimization: Adam Learning rate: 0.001 Time step: 10 Dropout: 0.5 Loss function: Mean Squared Error L2 Regularization	1800 stable excavation with 6 TBM operating parameters Train: 80% Validation: 20% Test: 180 random data

Loss function: After the output is calculated by the model, the neural network obtains the difference between the predicted value and the actual value of the output by calculating the loss function value, then performs backpropagation and update weight. Common loss functions including Mean Squared Error (MSE), Mean Absolute Error (MAE), Cross Entropy and so on. Among the above, MSE has been widely used in regression problems of neural networks. The formula is:

$$MSE = \frac{\sum_{i}^{n}(y_i - y_i^{'})}{n} \tag{3–7}$$

In the equation, y_i is the predicted value while $y_i^{'}$ is the actual value.

Optimization algorithm: The optimization algorithm is the method for adjusting parameters according to loss function and gradient during backpropagation. Common algorithms including Stochastic Gradient Descent algorithm (SGD), Momentum algorithm, Adam algorithm with adaptive learning rate, RMSProp algorithm, AdaDelta algorithm and so on. According to the characteristics of the dataset, the model needs to maintain a relatively large learning rate in the early stage and a small learning rate in the later stage. Therefore, the Adam algorithm is adopted due to it can calculate the exponential mean of the gradient with attenuate and have an adaptive learning rate.

L2 Regularization: In order to prevent model over-fitting due to weight in the model being too complex, L2 Regularization adding a parameter that is related to the complexity of the model after loss function. The formula is:

$$L2 = \frac{1}{2}\lambda\theta_i^2 \tag{3–8}$$

In the equation, λ is the setting parameter, θ is the parameter of the training model.

Batch Normalization: Batch Normalization is the method of transforming the distribution of input data to a form which is more suit for the activation function. It not only could boost up the training of neural networks. But also reduce impact during weight adjustment from the front layer to the later layer, to improve the network.

Dropout: The dropout method refers to randomly drop hidden layer cells according to a certain probability during the training process of the deep learning network, to provide different training object for each epoch in networks. Apply dropout can alleviate the model's dependence on specific neurons, thereby preventing overfitting.

3.4 Results

Three formulas are used to calculate the differences between prediction values and realistic values to measure the accuracy of the trained model. The formulas are:

Root Mean Square Error (**RMSE**)

$$RMSE = \sqrt{\frac{1}{n}\sum\nolimits_{n}^{1}(y_i^{'} - y_i)^2} \tag{3–9}$$

In the equation, y_i is the predicted value while $y_i^{'}$ is the actual value.

Mean Absolute Error (**MAE**)

$$MAE = \frac{1}{n}\sum\nolimits_{i=1}^{n}\left|y_i - y_i^{'}\right| \tag{3–10}$$

In the equation, y_i is the predicted value while $y_i^{'}$ is the actual value.

Mean Absolute Percentage Error (**MAPE**)

$$MAPE = \frac{1}{n}\sum\nolimits_{i=1}^{n}\left|\frac{y_i^{'} - y_i}{y_i^{'}}\right| \tag{3–11}$$

In the equation, y_i is the predicted value while $y_i^{'}$ is the actual value.

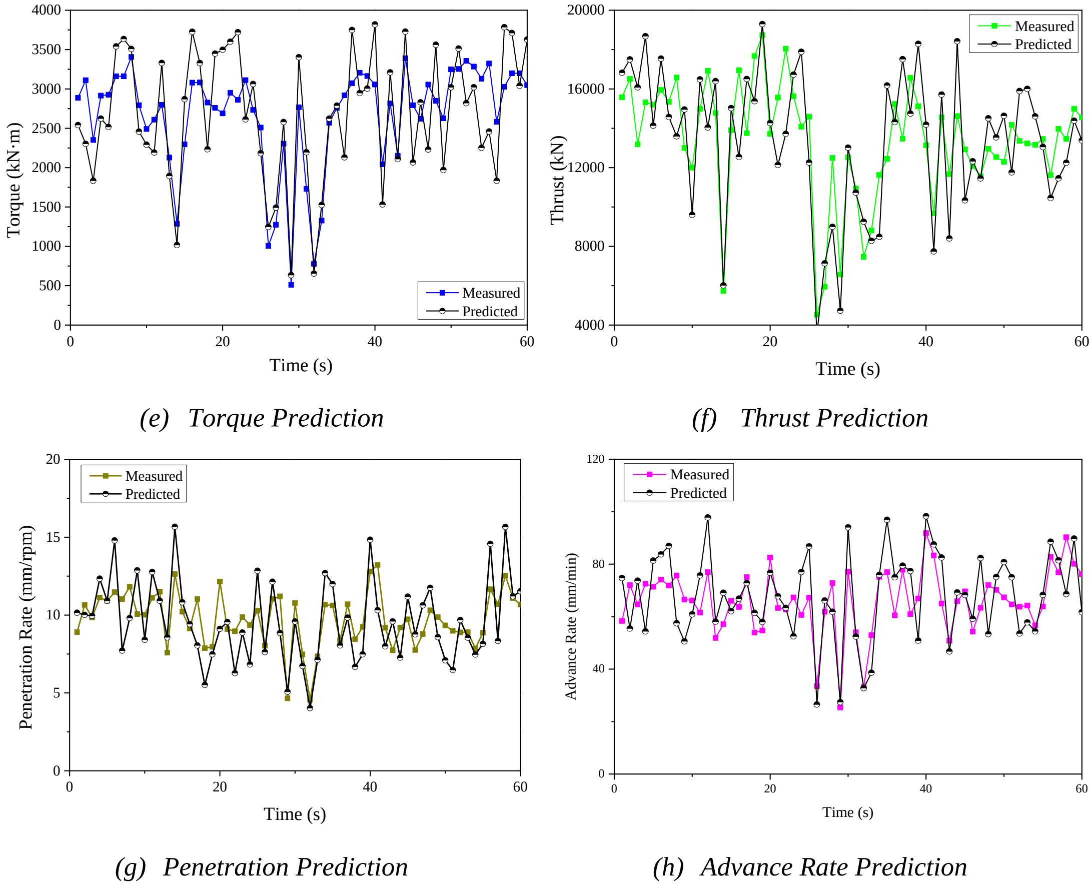
Goodness of Fitting (***R*²**)

$$R^2 = 1 - \frac{\sum(y_i^{'} - \hat{y}_i)^2}{\sum(y_i^{'} - \bar{y}_i)^2} \tag{3–12}$$

In the equation, y_i is the predicted value, $\bar{y}_i$ is the mean value and $y_i^{'}$ is the actual value.

(1) Results of Prediction of TBM Performance Parameters

Figure 3.10 illustrates the prediction of TBM operating parameters by optimized LSTM model, which contains both measured and predicted data.



(e) Torque Prediction

(f) Thrust Prediction

(g) Penetration Prediction

(h) Advance Rate Prediction

Figure 3.10 Results for TBM performance prediction

It can be seen from Figure 3.10 that the model can accurately predict the results on the test set, including servals extreme high value or very low value points in terms of average torque and average thrust. Those fitting proves the accuracy and effectiveness of the model. There are two main reasons explained the accuracy of the model: Firstly, the dropout layer is used in the network model to effectively prevent over fitting. Secondly, cross validation after prediction is adopted in the network structure, which effectively extracts the features relevant to the output and improves the prediction accuracy. Table 3.3 demonstrates the result for TBM operating parameters prediction.

Table 3.3 Results for TBM operating parameters prediction

Summary	**Parameters**	**RMSE**	**MAE**	**MAPE (%)**	$\boldsymbol{R^2}$
Without optimization	Torque	367.12	281.43	10.2%	0.89
	Trust	2143.337	755.2	9.5%	0.86
	Penetration	1.12	0.86	9.26%	0.85
	Advance rate	8.38	6.24	11.15%	0.82
With optimization	Torque	354.12	262.93	9.85%	0.94
	Trust	1956.265	705.3	8.68%	0.90
	Penetration	1.07	0.74	8.75%	0.91
	Advance rate	7.67	6.02	10.28%	0.89
Overall improve percentage		**6.30%**	**7.67%**	**6.34%**	**6.46%**

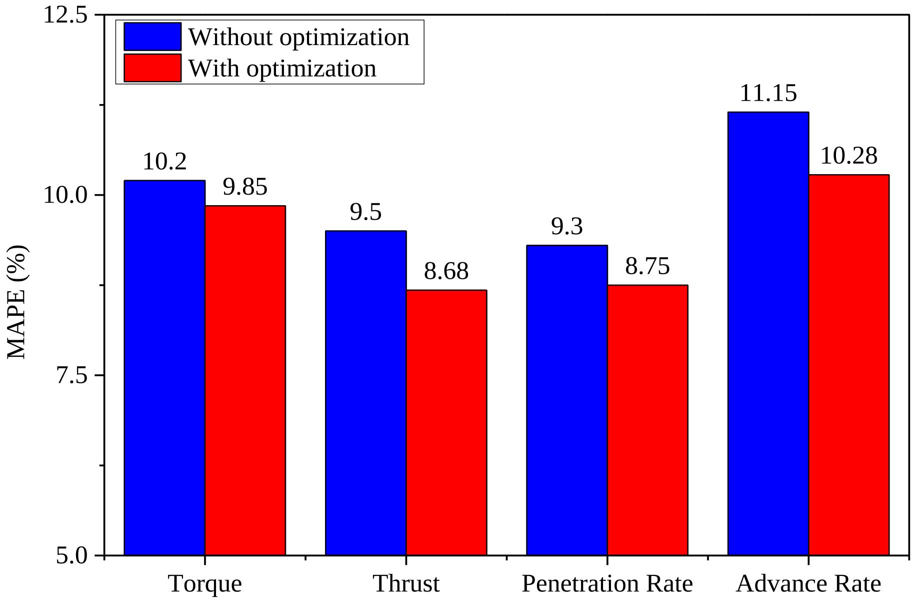
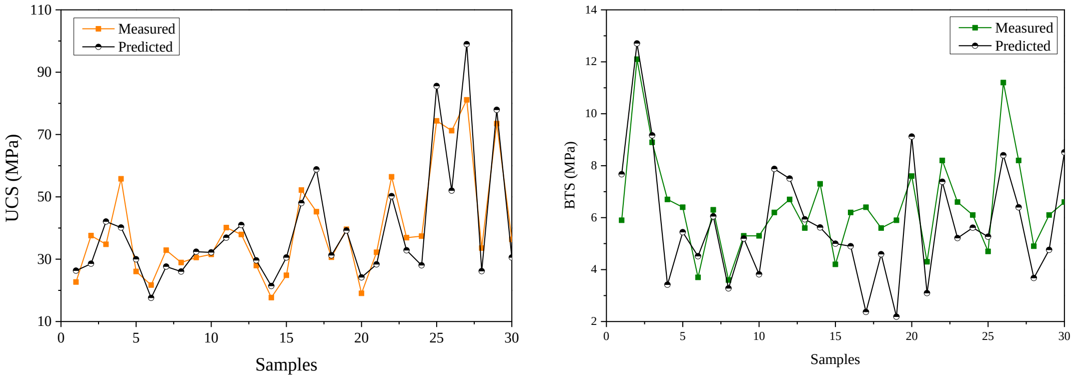


Figure 3.11 Results comparison of TBM performance prediction errors

Table 3.3 shows the statistics of evaluation indexes of TBM tunnelling parameters prediction results in stable section and Figure 3.11 illustrate the performance comparison between the original LSTM model and Optimized LSTM model. Four indexes, root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE) and goodness of fit R^2, are selected to evaluate the prediction effect of the model. It can be seen from the table that the goodness of fit of the four driving parameters is relatively high, among which the fitting degree of three equipment control parameters is above 0.9, the goodness of fit of cutter head torque *T* is 0.94, and the goodness of fit of penetration rate *n* is 0.91, which indicates that the prediction accuracy of equipment parameters is high. For the other two equipment parameters: the goodness of fit of total thrust F is 0.90, while the advance speed V is 0.89, the prediction accuracy is slightly lower than the other three parameters. Considering the magnitude range of each parameter of TBM, the other three error indicators RMSE, MAE and MAPE the average absolute relative errors are less than 12%. In general, after the improvement and optimization, the LSTM model based on RNN achieved higher prediction rate for TBM operating parameters.

(2) Results of Prediction of Rock Mass Parameters

During the excavation process, it is also important to predict the current geological condition. The accurate prediction of the geological condition of the construction working face can not only further confirm the engineering geological and hydrogeological conditions until the construction is carried out successfully, but also reduce the probability and prevent geological disasters. At the same time, it can provide the geological basis for optimizing engineering design and provide geological data for maintenance after completion of excavation. The rock parameters are obtained by experiments on the core obtained during the tunnel excavation period (e.g., UCS, BTS and Bi).



(a) UCS Prediction *(b) BTS Prediction*

Figure 3.12 Results for rock mass parameter prediction

Figure 3.12 illustrates the comparison of the measured value and the prediction results of the model. The model can accurately predict the results on the test set, however, for servals extreme high value or very low value point in terms of average the prediction results has relatively higher errors.

Table 3.4 Results for rock mass parameter prediction

Summary	Parameters	RMSE	MAE	MAPE (%)	R^2
Without optimization	UCS	13.66	10.56	25.22%	0.78
	BTS	0.66	0.47	8.47%	0.90
With optimization	UCS	12.17	10.12	22.43%	0.80
	BTS	0.62	0.43	7.73%	0.93
Overall improve percentage		**8.48%**	**6.34%**	**9.90%**	**2.95%**

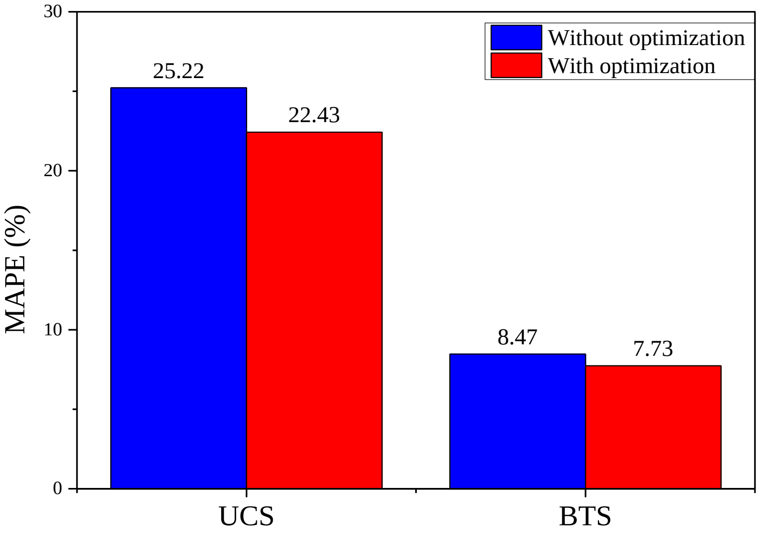


Figure 3.13 Results comparison of rock mass parameters prediction errors

Table 3.4 shows the statistics of evaluation indexes of rock mass parameters prediction results in the stable section and Figure 3.13 illustrate the performance comparison between the original LSTM model and optimized LSTM model. Four indexes, root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE) and goodness of fit R^2, are selected to evaluate the prediction effect of the model as same as previous. The prediction results indicated that the model has accurately predicted the BTS with the goodness of fitting 0.93. However, the results of UCS prediction have a higher error rate than BTS. This may recognize as the influence index of UCS is more complex. Both accuracies of prediction are improved after optimization of the model, which proves the optimization of the model is effective.

3.5 Conclusion

This chapter mainly focuses on the application of neural networks in TBM operating parameters prediction during the tunnel excavation period. The reason for choosing LSTM is that it is an algorithm based on the RNN, which is very effective for data with sequence characteristics and it can extract timing information and semantic information in data, while the LSTM solves the gradient explosion problem existing in RNN by introducing the gate system.

In consideration of the lithology, surrounding rock type and groundwater activity of the excavation scope, through the machine learning of TBM data and rock data, the equipment operating parameters of the TBM stable operating section are predicted. The conclusions are as follows:

(1) Based on the existing TBM operation data, the improved LSTM machine learning method based on RNN can effectively predict the TBM tunnelling parameters. The mean absolute percentage errors (MAPE) are less than 12% and the fitting degree R^2 is greater than 0.89, which has relatively high prediction accuracy and can provide guidance for the parameter selection of TBM during the stability excavation section.

(2) The hyperparameters in the LSTM model are optimized to achieve better prediction results for TBM operating parameters, and the improved algorithm had higher prediction accuracy compared with the original. Through the improvement and optimization of the prediction model, the average prediction accuracy is increased by 7%, which proves that the improvement of the model is effective.

(3) The LSTM model proposed in this thesis has high prediction accuracy for thrust and torque T in stable excavation section, and the goodness of fitting is greater than 0.9. Among other equipment parameters, the prediction accuracy of penetration is higher, and the fitting degree is 0.91; the goodness of fitting of prediction of propulsion speed V is 0.89, which is lower than the other three parameters. In terms of rock mass parameters, the goodness of fitting for BTS is achieved 0.93, however, the error prediction of UCS is significantly higher than other parameters. It is preliminarily speculated that the influence factor of UCS is more complex.

Chapter 4: TBM Segments Defects Detection Based on Two Stage Object Method

This chapter introduces the R-CNN algorithm and its application to TBM lining defects detection. Also, its specific parameters are adjusted to improve detection performance. Faster R-CNN is an algorithm based on the Fast R-CNN algorithm with adjusted and optimized network structure. From the structural aspect, Faster R-CNN belongs to the two-stage algorithm, and compare with the previous two-stage algorithm, Faster R-CNN has integrated feature extraction, proposal extraction, bounding box regression and classification into one network, which significantly improves the comprehensive performance, and improves the detection accuracy and speed in detection.

4.1 Faster R- CNN

After the R-CNN and Fast R-CNN, Girshick (2017) proposed a new algorithm called Faster R-CNN in 2016. Faster R-CNN solved the problem of timing is not ideally caused by that Fast R-CNN still needed to extract the region proposal in advance. Figure 4.1 shows the structure of Faster R-CNN.

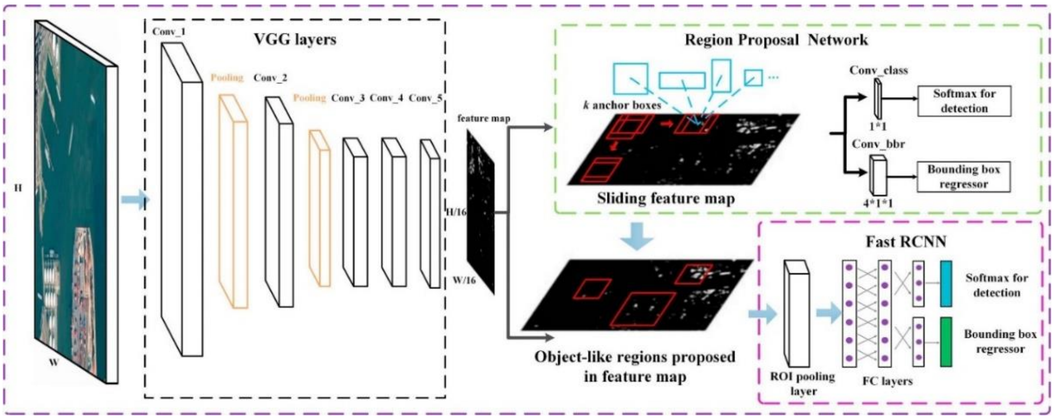


Figure 4.1 Structure of Faster R-CNN (Deng et al., 2018)

In the structure aspect, Faster R-CNN has integrated feature extraction, proposal extraction, bounding boxes regression and classification into the Region Proposal Network (RPN), which greatly improves the comprehensive performance, especially in terms of detection speed.

1. Feature Pyramid Network (FPN)

Feature Pyramid Network (FPN) is a network structure that comprehensively uses the information of multi-scale feature images to detect the object (Lin et al., 2017). Many algorithms are using the top feature map of FPN for object detection such as Fast R-CNN, R-FCN and other algorithm based one region proposal (two-stage algorithm). One disadvantage of the method is that the top feature map may not contain the feature of interested objects. Some small objects in the image after a convolution layer of the neural network and max pooling layer, the receptive field of the feature map at the top layer will become extremely large, and it will gradually lose its sensitivity to small objects. Figure 4.2 demonstrates the structure of the FPN.

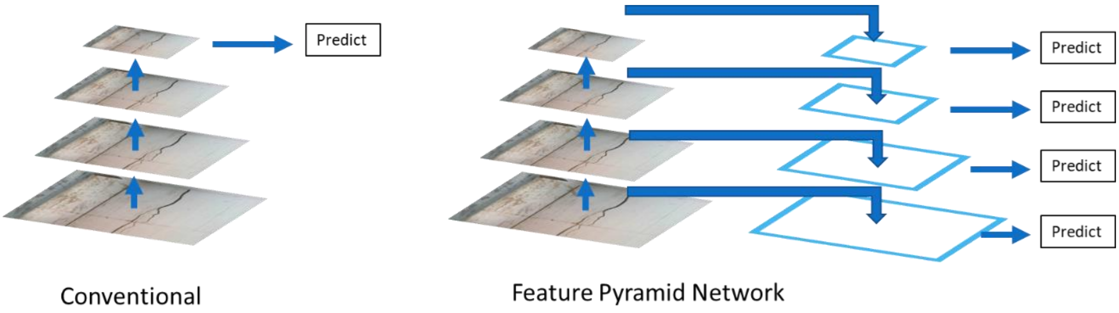


Figure 4.2 Structure of Feature Pyramid Network

A typical FPN consist of 3 layers, which are:

1. Bottom-top pathway: It refers to a process in which ordinary CNN features are condensed and expressed layer by layer from bottom to top. The lower layer reflects the shallower level of image information features such as boundary; The higher layer reflects the deeper image features such as object shape and specific category.
2. Top-bottom pathway: In general, the feature map size of the upper feature output is relatively small, but it can represent the picture information of a larger

dimension (and higher level). This kind of high-level information has been proved by experiments to play an important role in subsequent tasks such as object detection and classification. Therefore, when processing the information of each layer, the high-level information of the previous layer will be referred as its input.

3. The representation relation between the CNN layer features and the output of each level: FPN author point out that the use of 1x1 Conv can generate better output features, which can effectively reduce the number of channels in the middle layer. Finally, these 1x1 Convs result in the same number of channels for each feature map with different dimensions.

FPN mainly solves the multi-scale problem in object detection. Through simple network connection changes, it greatly improves the performance of small object detection without increasing the computation amount of the original model.

2.Region Proposal Networks (RPN)

Classic detection methods are time-consuming to generate bounding boxes. For example, OpenCV Adaboost uses the sliding window and the image pyramid to generate bounding boxes. Another one is R-CNN using the selective search method to generate the bounding box. However, Faster R-CNN abandons both the traditional sliding window and selective search method, directly use Region Proposal Network to generate the bounding boxes, which is a huge advantage of Faster R-CNN and it can significantly improve the generation speed of bounding box.

RPN was first proposed by Ren (2017) and applied to the Fast RCNN algorithm, it is combined with the Fast R-CNN algorithm to greatly improve the computation efficiency of the network. It is a self-contained, fully convoluted network that could take any size of the original image, outputs a bunch of bounding boxes, and scores each box. The method of generating the regional proposal box is to use a small sliding window to convolve the sliding on the top layer of the convolutional network and transfer the acquired feature vectors into the regression layer and classification layer of the bounding box. The purpose of the two fully connected layers is different. For the regression layer, it is to generate bounding boxes, while for the classification layer, it is to score the region proposed. The figure 4.3 shows the structure of region proposal networks.

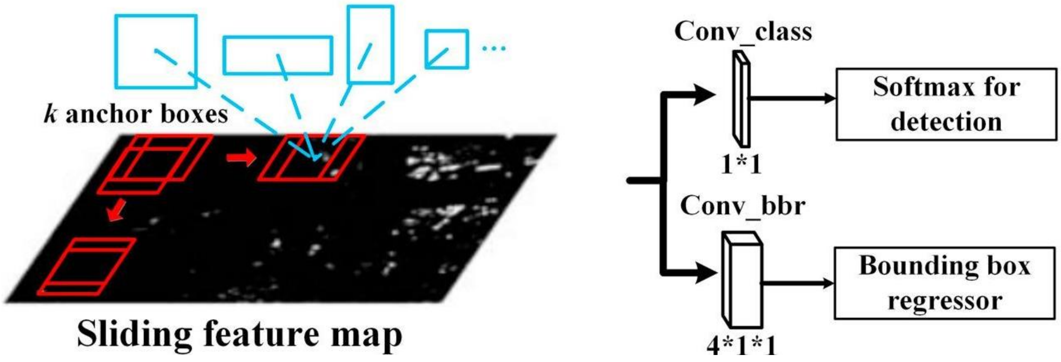


Figure 4.3 Structure of region proposal networks (Deng et al., 2018)

It clearly shows that the RPN is divided into two lines, the top one getting the positive and negative categories with the SoftMax category anchors, the bottom one is used to calculate the evaluation box regression offset for anchors to get an accurate proposal. The final proposal layer is responsible for combining the positive anchors and the corresponding evaluation box regression offset to get the proposals while eliminating proposals that are too small and out of bounds. When the entire network moving to the proposal layer, the function equivalent to object positioning is completed.

3. Anchors

Anchor plays an important role in RPN. Anchors are a set of rectangles generated by the RPN code, and the multi-scale approach is often used in instrumentation. Figure 4.4 shows the typical anchor in the R-CNN network.

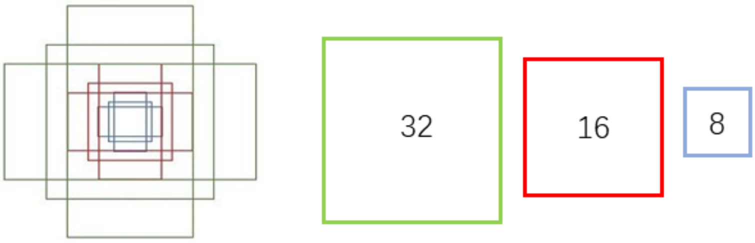


Figure 4.4 Schematic of Anchors with size 32,16 and 8

In Faster R-CNN, iterate over the feature maps are computed by convolutional layers, with the nine anchors as the initial detection box for each point. Since it is not accurate to get the bounding boxes by anchors, the bounding box position needs to be modified twice by bounding box regression. The parameters of the nine anchors are not shared, but the parameters of each sliding window position are shared. Figure 4.5 shows the anchor in RPN.

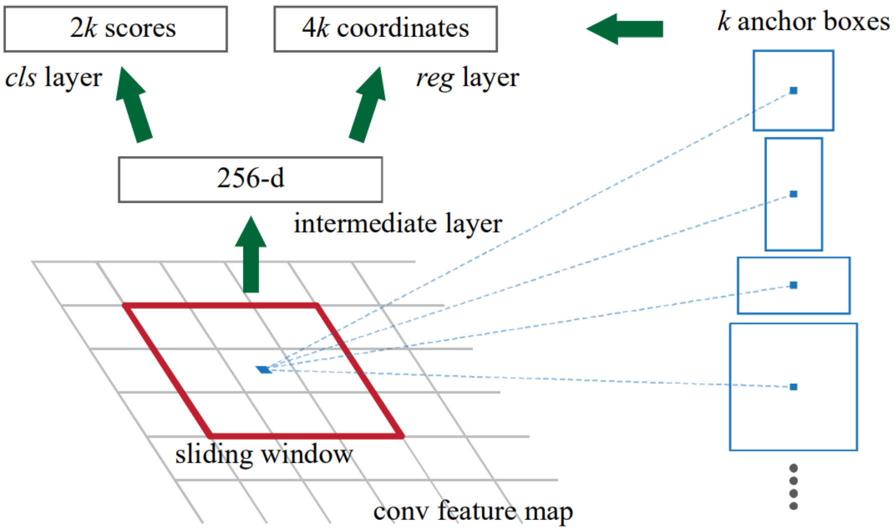


Figure 4.5 Anchors in region proposal networks (Ren et al., 2017)

In general, RPN is to set dense candidate anchor on the scale of the original drawing, and then use CNN to determine which anchor is the positive (anchor with the object) and which is the negative (anchor without the object). The structure of the network could roughly define as binary classification.

Table 4.1 Neural network structure comparison

	R-CNN	**Fast R-CNN**	**Faster R-CNN**
Test time per image with proposals	50 seconds	2 seconds	0.2 seconds
Speedup	1x	25x	250x
mAP (VOC 2007)	66.0	66.9	66.9

Table 4.1 shows that Faster R-CNN designed a network RPN to extract candidate regions, which replaced the time-consuming selective search algorithm and greatly improved the detection speed. Compared with the previous algorithm, the detection speed was qualitatively improved.

4. Post Processing

Since anchors generally overlap, there is also overlap between proposals for the same object. In order to solve the problem of overlapping proposals, the NMS algorithm is used to discard the proposal with a higher score and the Intersection over Union (IoU) is greater than the preset threshold.

After filtering the IOU threshold, the NMS further iteratively optimizes the boundary of the bounding box. Firstly, the score of probability information of all reserved prediction boundary boxes is sorted, then all prediction boundary boxes are traversed expect one with the highest score, the percentage of overlap reigion between boxes is calculated, next delete the prediction bounding boxes which are larger than a certain threshold value, repeat this step repeatedly in order to filter out a large number of false, inaccurate, overlapping prediction bounding boxes, and retain the prediction boundary box that appropriately describes the object position. Figure 4.6 describes the process of NMS.

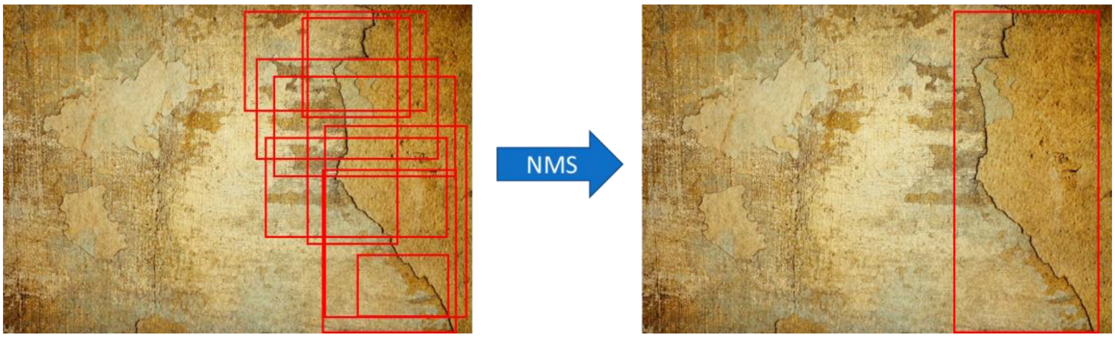


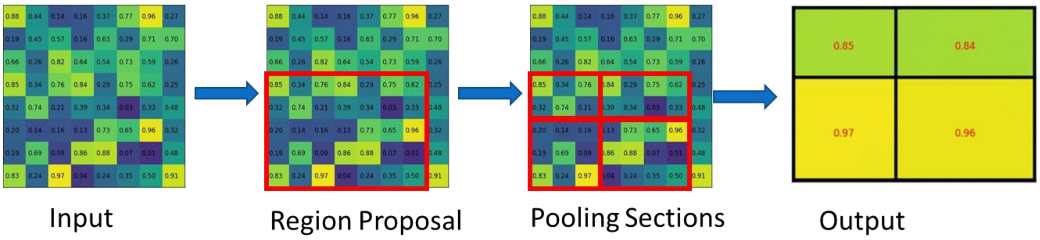
Figure 4.6 *Process of Non-Maximum Suppression (NMS)*

While the NMS may seem simple, the default of the IoU threshold needs to be handled with caution. If the IoU value is too small, too many proposals of objects may appear. If the IoU value is too large, some proposals of the objects may be lost. In the Faster R-CNN, the typical IoU value is 0.7.

After NMS processing, proposals are sorted according to scores. The proposal with the highest score would appear as a bounding box to indicate the object's location. The number of proposals generated during RPN is set by the user. In the thesis of Faster R-CNN, the number of proposals is set as 2000.

5. Region of Interest (RoI) Pooling

After RPN processing, there are several different size object proposals without the class scores. How to use these boundary boxes and classify them is the key issue under current situation. One of the simplest ways is to crop each proposal and send it into the pre-trained base network to extract the features, then the features will be extracted to train the classifier, but this method requires all proposals to be computed which is inefficiently and slowly. Therefore, the region of interest pooling (RoI Pooling) was proposed in Faster R-CNN to solve this problem. Figure 4.7 shows the RoI pooling in Faster R-CNN.



0.88	0.44	0.14	0.16	0.37	0.77	0.96	0.27
0.19	0.45	0.57	0.16	0.63	0.29	0.71	0.70
0.66	0.26	0.82	0.64	0.54	0.73	0.59	0.26
0.85	0.34	0.76	0.84	0.29	0.75	0.62	0.25
0.32	0.74	0.21	0.39	0.34	0.03	0.33	0.48
0.20	0.14	0.16	0.13	0.73	0.65	0.96	0.32
0.19	0.69	0.09	0.86	0.88	0.07	0.01	0.48
0.83	0.24	0.97	0.04	0.24	0.35	0.50	0.91

0.85	0.84
0.97	0.96

Figure 4.7 *Region of Interest (RoI) pooling in Faster R-CNN*

Faster R-CNN uses the conv feature map to speed up the calculation efficiency, which is RoI Pooling is used to extract the feature map of fixed size for each proposal while R-CNN is used for the classification of feature map with a fixed size.

4.2 Improvement of Faster R-CNN for Cracks Detection

Although Faster R-CNN has better detection accuracy in objects, however the algorithm is not designed for cracks detection while during the pre-test period it shows that there are several bounding boxes for a single crack in the test image, and the position of the bounding box is not very accurate, the false results of detection are relatively high and the detection results could not satisfy the requirement. Therefore, the network needs further modification to improve the accuracy in order to achieve a better detection effect.

Due to the inhomogeneity of concrete material, the texture of the normal surface is generally uneven. Therefore, the background texture of the collected fracture image itself has significant differences. In addition, the length, width and depth of cracks vary with the severity of cracks. If there are larger particles with a similar or even brighter colour to the background, the texture of the crack image will be change. Generally, the crack image is darker than the ordinary concrete surface. The number of pixels representing cracks are significantly less than that of the background and the grey value of the ordinary surface overlaps with that of crack. For the grey image with cracks taken by the camera, it often includes non-uniform illumination and slow change part formed by the imaging system, that is the change of image from middle bright to surrounding dark, which is a signal with low frequency and high amplitude. Based on the long and narrow geometric features of cracks, the grey outline features along the fracture direction and the crack width variation characteristics along the fracture length direction can be obtained.

In the process of training and testing of model with images, cracks occupy a small area in the image while most areas are backgrounds, while the shape of the cracks is irregular, and there is no uniform shape. Most of the cracks are elongated and need to be covered by a rectangle with a relatively large aspect ratio.

Considering that the size of anchors in Faster R-CNN is relatively large, initially it was not designed for detecting small objects, therefore, the size of the anchor is adjusted. Originally there are three different sizes of the anchors which are 32, 16, 8, respectively. To achieve better small object detection performance, one extra smaller size anchor is added. Figure 4.8 demonstrates the improvement of Faster R-CNN.

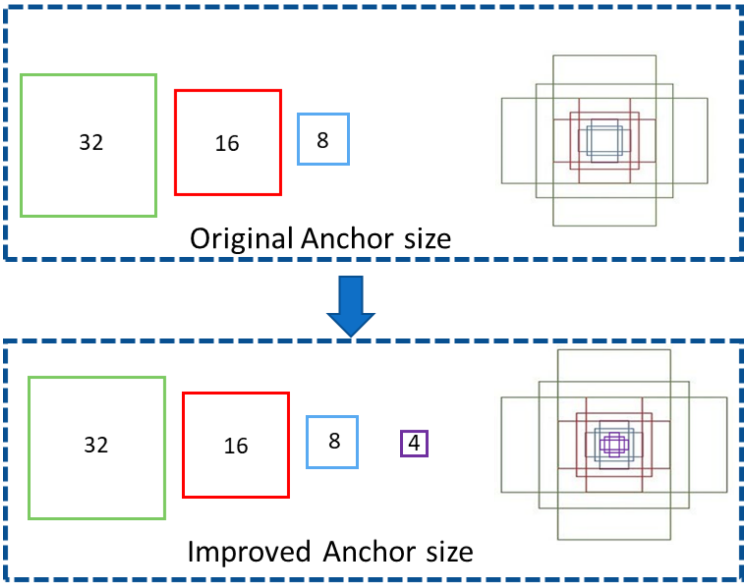


Figure 4.8 Improvement of Faster R-CNN model by adding an extra anchor

The new anchor in Faster R-CNN is modified with 4 different sizes, a relatively smaller size is added to provide the alibility of capture the smaller objects during the detection stage. Moreover, the IoU threshold was adjusted from 0.7 to 0.5 to obtain a better crack detection effect and higher detection accuracy. Figure 4.9 shows the structure of Faster R-CNN for crack detection.

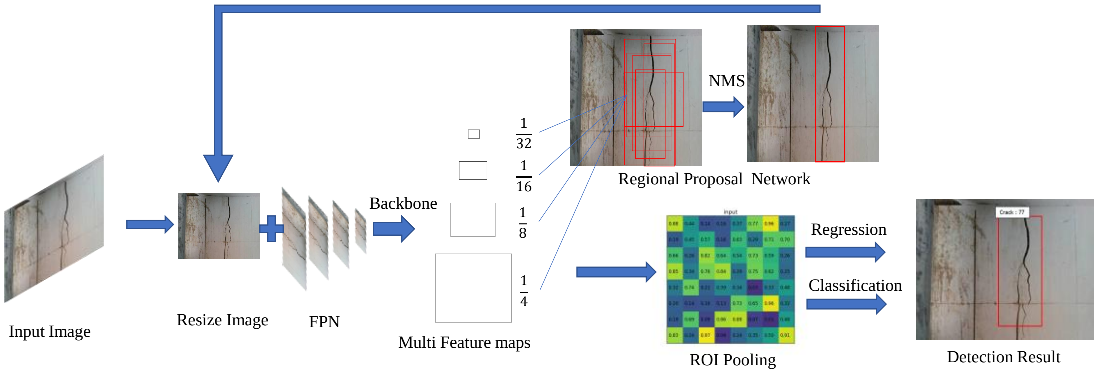


Figure 4.9 Structure of Faster R-CNN for cracks detection

4.3 Image Database, Model Training

Image annotation is an important task in deep learning models and application construction. In order to make deep learning algorithms have the ability to recognize certain objects, it is necessary to provide a large number of labeled data to train the network, and the labeled data is feed to the network to help the network to gain the ability to recognize the cracks. The method of annotation for the image is to let the professional civil engineering technicians classify the cracks manually to ensure that there is no deviation between the crack image and the non-crack image of the tunnel. Figure 4.10 demonstrates the processing of data annotation by using LabelImg software.

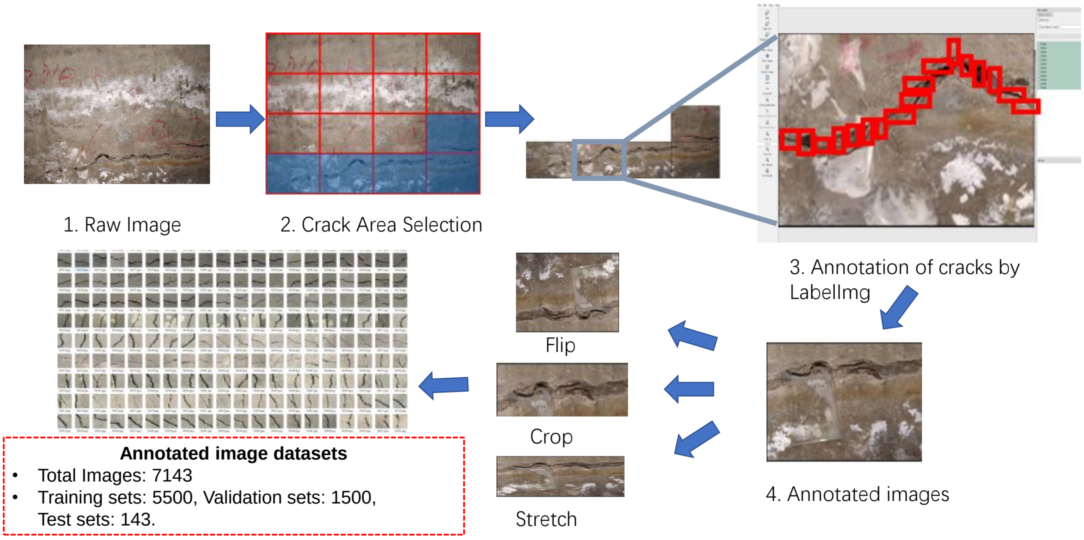
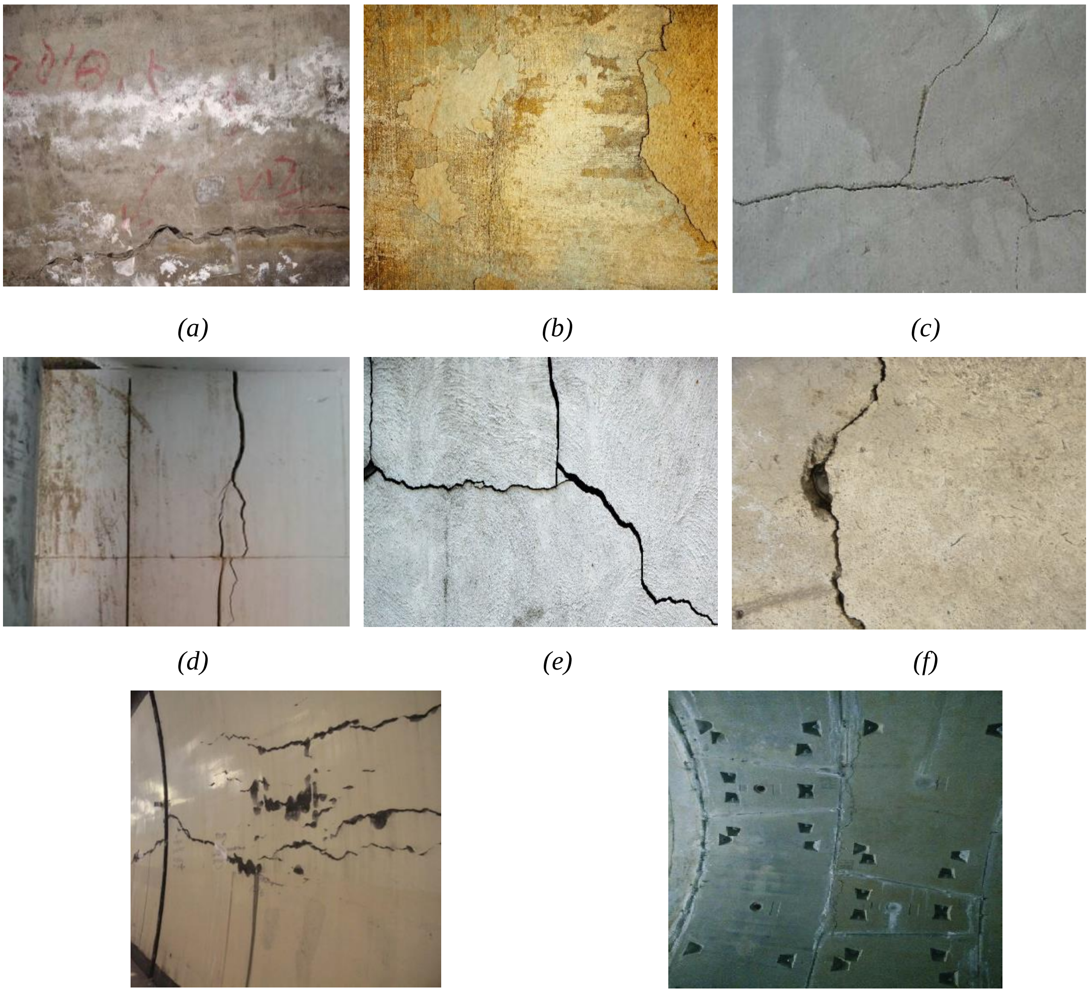


Figure 4.10 Processing of data annotation

The image database of cracks contains 7143 images, some of the images are from the online image database ImageNet. In order to effectively train and verify the detection ability of the network model, 7000 images were selected as the training database for network training and the remaining 143 were used for blind test of the model. In order to better train the model and improve the accuracy, 5500 of the 7000 training images were used as the training set and the remaining 1500 as the validation set. Figure 4.11 shows the test set of cracks.



(a) (b) (c)

(d) (e) (f)

(g) (h)

Figure 4.11 Test sets of cracks (a) Image of cracks with handwriting and white paint. (b) Image of cracks with wall peeling off. (c) Image of cracks with two different propagation direction. (d) Image of cracks with wall joint. (e) Image of cracks with multi direction. (f) Image of cracks with hole. (g) Image of tunnel with large scale cracks (DANANGToday, 2017). (h) Image of TBM segments with cracks

4.4 Results of Faster R-CNN

Before optimized the Faster R-CNN algorithm, the setting of training iterations, step length, learning rate and other parameters required by the algorithm is set, then the initial Faster R-CNN training result model can be obtained during the training of the neural network, and then the experimental result can be used as the baseline. After improving the Faster R-CNN algorithm, the results can be also obtained for comparison and the new trained model can be used for cracks detection of tunnels. The following shows the detection performance of the improved Faster R-CNN algorithm on some tunnel cracks, as shown in Figure 4.12.

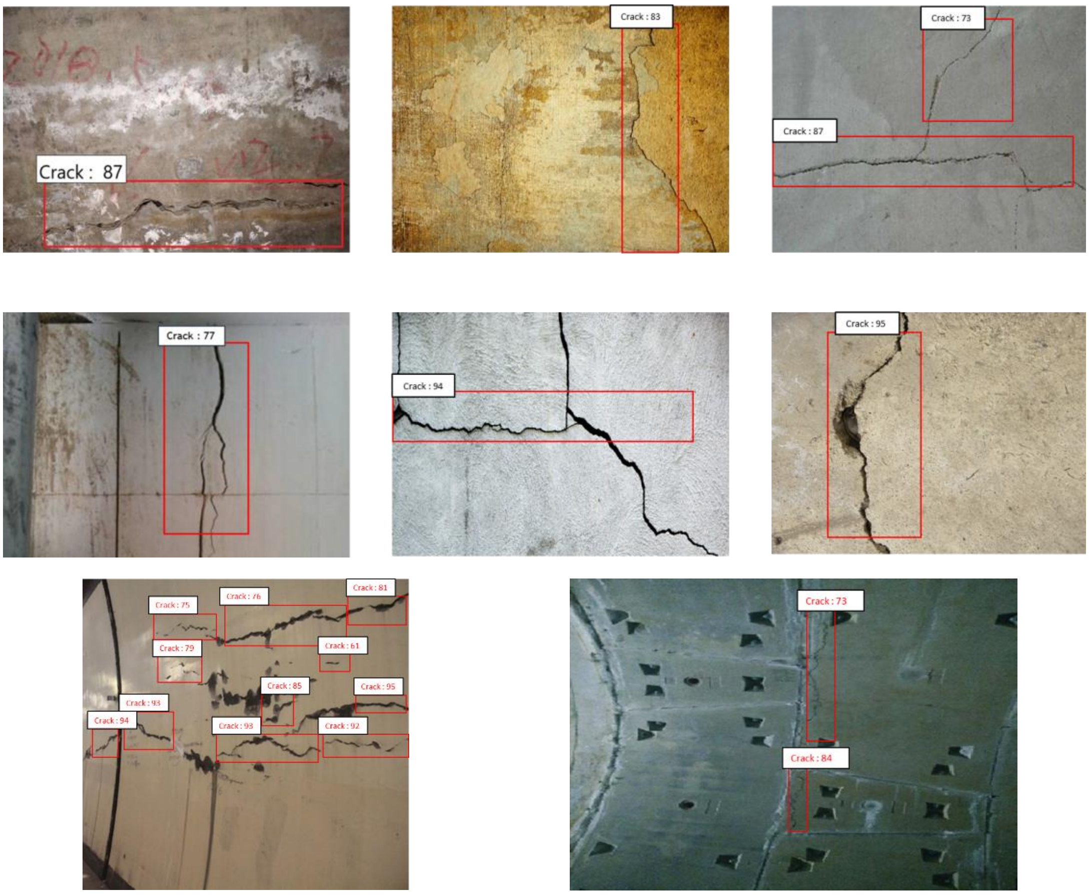


Figure 4.12 Detection performance of the model on the test set

Conduct visual analysis on the training log files generated during the training process. The curve of the Loss function chart during the training process and the final Precision and Recall (PR) curve can be obtained. The meaning of the Loss curve refers to the change in the value of the Loss function during the training process. The trend of the change is continuously downward from high to low, indicating that the training process is gradually tending to the state of convergence, and the training model in the state of convergence has a better detection effect. Figure 4.13 shows the Loss Curves of Faster R-CNN during model training and Figure 4.14 shows the P-R Curve of Faster R-CNN during model training.

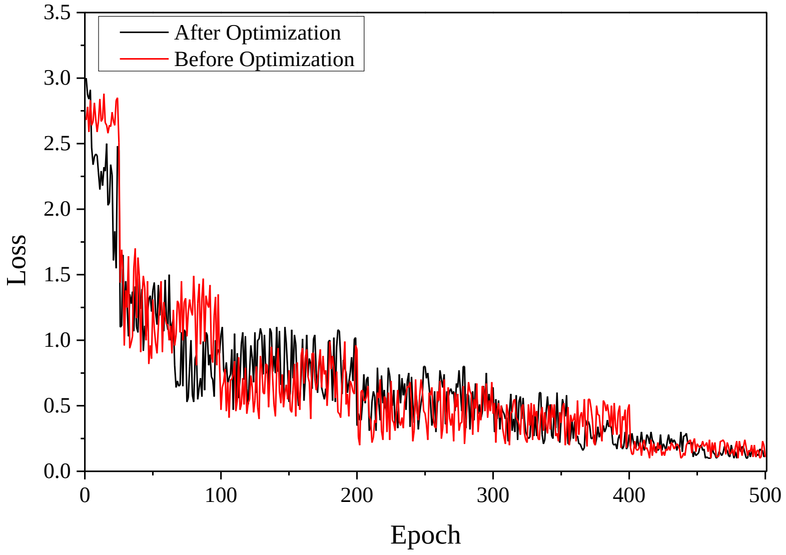


Figure 4.13 Faster R-CNN Loss Curves.

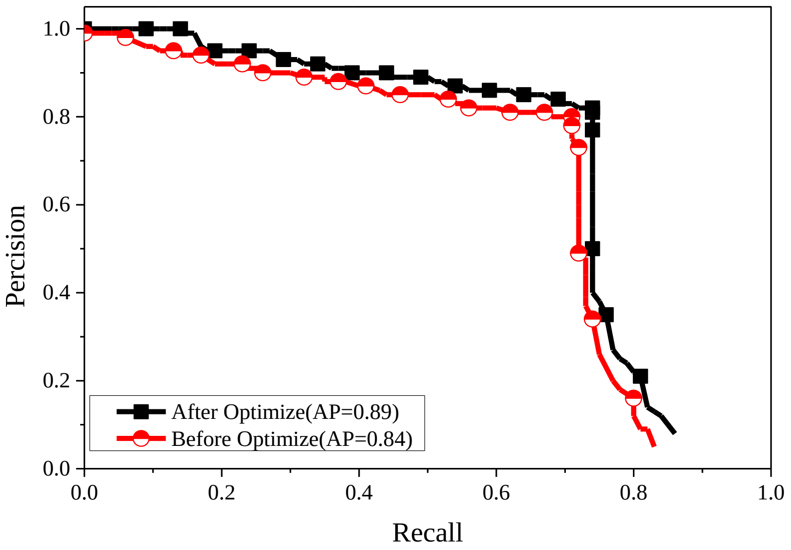


Figure 4.14 Faster R-CNN P-R Curves

The definition of PR curve is the curve of accuracy and recall, the abscissa is the recall rate (which ranges from 0 to 1) while the vertical coordinate is the precision (with values ranging from 0 to 1). When drawing the PR curve, points are divided according to the samples of each test set as the threshold, different recall rate and precision rate values can be obtained, then connect these different points and a two-dimensional PR curve are obtained. The area enclosed between the PR graph, the x-coordinate axis and the y-coordinate axis is the value of AP, which is a comprehensive index used to measure the detection effect of the object. The larger the value, the better the performance of the detection algorithm.

In the performance measurement of object detection, the common way is to use the Mean Average Precision (mAP) value to evaluate the algorithm's object detection performance. The evaluation target is not the whole image, but the prediction bounding box generated in the image to participate in the loss function calculation and forward and backward propagation during the process of network training.

True Positive (TP) indicates that the prediction bounding box is correctly labelled on the positive samples, which is the image with positive samples has been successfully predicted as positive samples.

False Positive (FP) indicates that the prediction bounding box is incorrectly labelled on the negative samples, which is the image with negative samples has been successfully predicted as positive samples.

False Negative (FN) indicates that the prediction bounding box is incorrectly labelled on the positive samples, which is the image with positive samples has been incorrectly predicted as negative samples.

True Negative (TN) indicates that the prediction bounding box is correctly labelled on the negative samples, which is the image with negative samples has been successfully predicted as negative samples. Table 4.2 shows the classification of results.

Table 4.2 Classification of results

		Actual	
		True	**False**
Predict	**True**	True Positive (**TP**)	False Positive (**FP**)
	False	False Negative (**FN**)	True Negative (**TN**)

The full name of IOU is intersection over union. IOU calculates the ratio of the intersection and union of predicted and real borders. Figure 4.15 demonstrates the IOU and how to classify TP, FP, TN and TN.

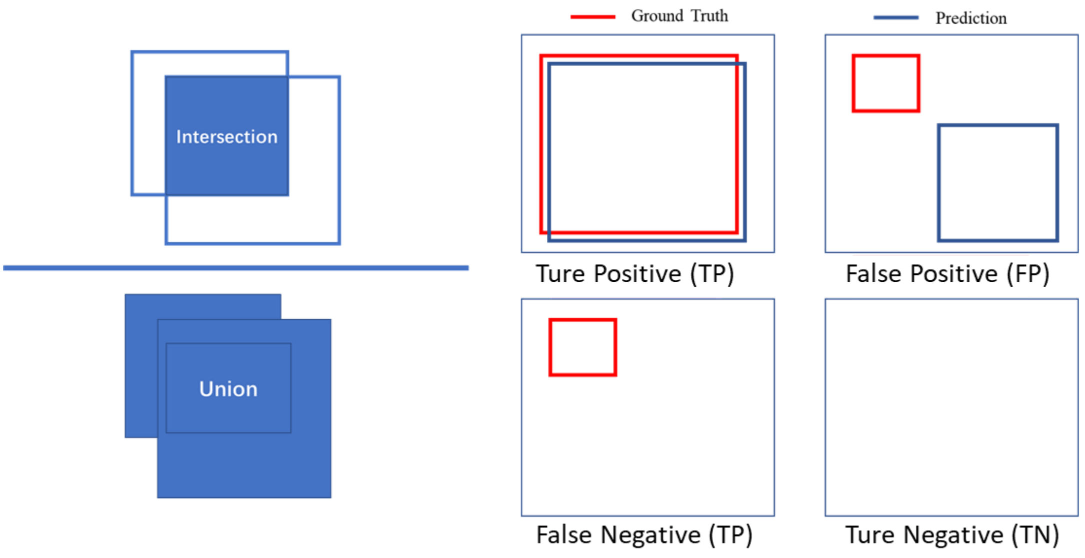


Figure 4.15 Calculation of Intersection Over Union (IOU) and TP, FP, TP and TN

IOU is a standard for measuring the accuracy of detecting corresponding objects in a specific data set. Once a bound box generated in the output result, it can be measured with IOU. In order to use IOU to measure objects of any size and shape, there are two things are required:

1. Ground-truth bounding boxes (manually mark the approximate range of the object to be detected in the training set image).
2. The output prediction range of algorithms.

This standard is used to measure the correlation between reality and prediction. the higher the value, The higher the correlation. The following equation shows the calculation process of IOU:

$$IntersectionOverUnion = \frac{Intersection}{Union} \tag{4–1}$$

The precision reflects the percentage of detected in the prediction bounding box with the correct label, which is the ratio of the number that detected category has been successfully predicted and the number of object detection of this category in the bounding box. the following equation shows the calculation process of precision:

$$Percision = \frac{TP}{TP+FP} \tag{4–2}$$

Recall reflects the percentage of correctly predicted bounding boxes containing the category of object to the number of objects with the positive labels in this image, and the calculation process is shown in the following equation.

$$Recall=\frac{TP}{TP+FN} \tag{4–3}$$

The missed inspection rate reflects the percentage of undetected in the predicted boundary box of the object category in the number of the object with the positive label in the category, which is inversely changing with the recall rate. The calculation of the missed inspection rate is shown in the following equation.

$$MissedInsepction=\frac{FN}{TP+FN}=1\text{-}Recall \tag{4–4}$$

False inspection rate reflects the percentage of incorrectly detected in the predicted boundary box of the object category in the number of the object in the category, which is inversely changing with the precision rate. The calculation of false inspection rate is shown in the following equation:

$$FalseInspection=\frac{FP}{TP+FP}=1\text{-}Percision \tag{4–5}$$

F1 Score is defined based on precision and recall, it reflected the performance of the model. The following equation shows the calculation process of the F1 Score:

$$F1Score=\frac{2PR}{P+R} \tag{4–6}$$

Average precision (AP) reflects the average accuracy of detection. When the validation set contains N images, it is the mean precision rate of the object category in each image, and the following equation shows the calculation process of Average precision:

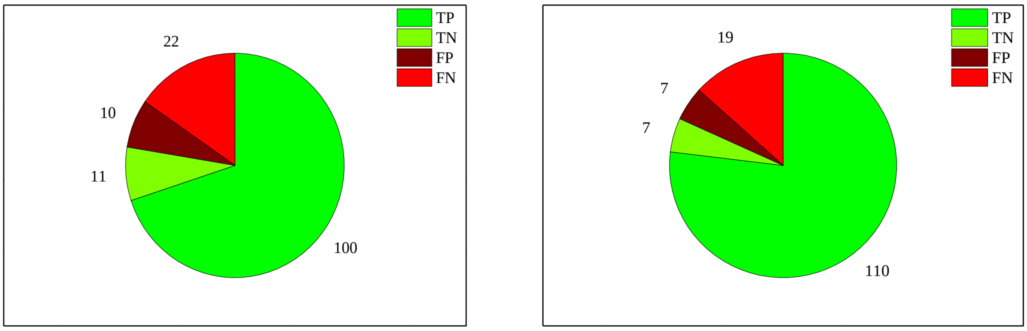
$$AP=\frac{\sum_{i=1}^{N} Percision}{N} \tag{4–7}$$

The above method provides the performance measurement for object detection, however, in the underground project defects detection, it concerned about whether all defects are detected, and whether the location and label of detection object are correct, which is recall and precision, while for safety concerned, recall is more important. Table 4.3 summarizes the detailed results of the trained model on test sets.

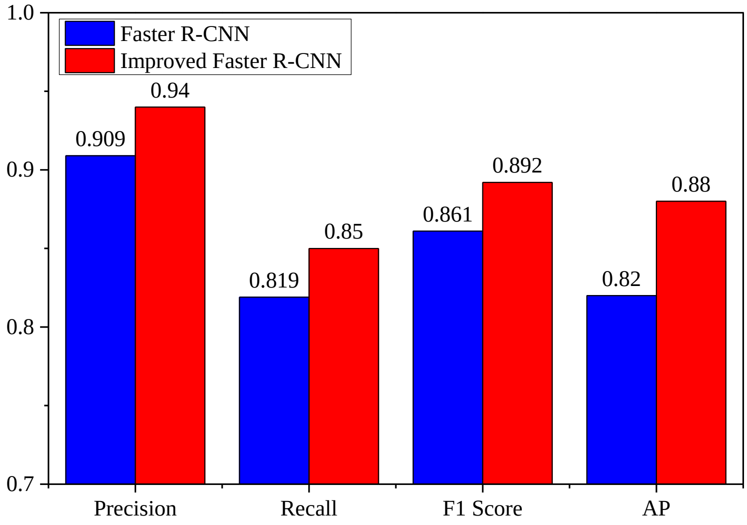
Table 4.3 Results of the trained model on tests sets

Algorithm	**Sample**	**TP**	**FP**	**FN**	**TN**	**Precision**	**Recall**	**F1 Score**	**AP**
Faster R-CNN	143	100	10	22	11	0.909	0.819	0.861	0.82
Improved Faster R-CNN	143	110	7	19	7	0.94	0.85	0.892	0.88

As shown from the table, compared with the original algorithm, the number of TP and TN is increased while the FP and FN are decreased. Four performance indicators (Precision, Recall, F1 score, AP) are all increased to varying levels.



(a) Detection results of original YOLO V3 *(b) Detection results of improved YOLO V3*



(c) Results comparison of original Faster R-CNN and improved Faster R-CNN

Figure 4.16 Detection Results of the image with Noise

Figure 4.16 demonstrates the detection results of the original model and the improved model. It clearly shows that compared with the original one, the improved model has higher performance on image sets. Both four indicators are increased which indicates the improvement of the model could increase the accuracy of crack detection and achieve higher performance.

After optimized the R-CNN, the performance index indicated that improvement could increase the performance of the model. Moreover, according to the AP index, compare with the original data, the detection accuracy is increased approximately 8% while the tested image and PR curve graph also indicate that the Faster R-CNN algorithm has a better performance compared with the original one. Overall, the improved R-CNN achieved higher detection performance on cracks.

4.5 Robustness Test

In underground engineering projects, due to the environmental effect and lighting conditions, the image is vulnerable to all kinds of interference, resulting in a decline in the detection effect. Image noise is some isolated pixels in an image, which will disturb the actual visibility of the image, making the image fuzzy. Figure 4.17 demonstrates an example of how noise disturbs the computer vision classification.

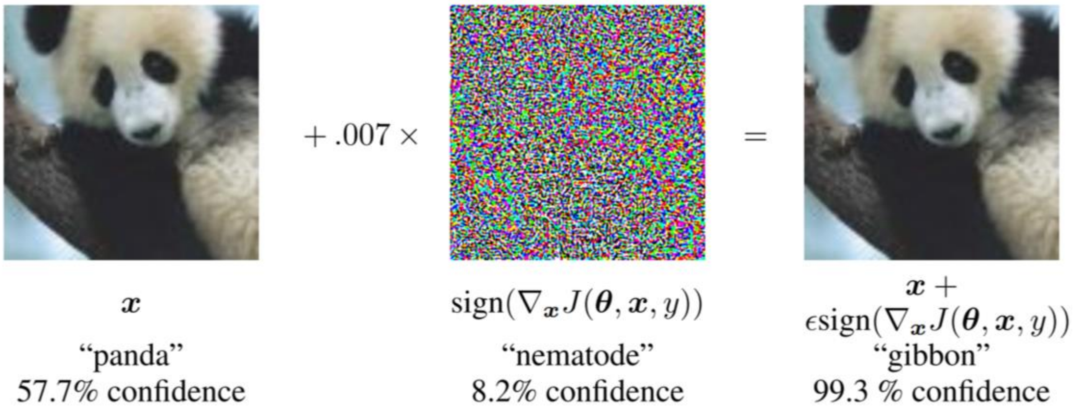
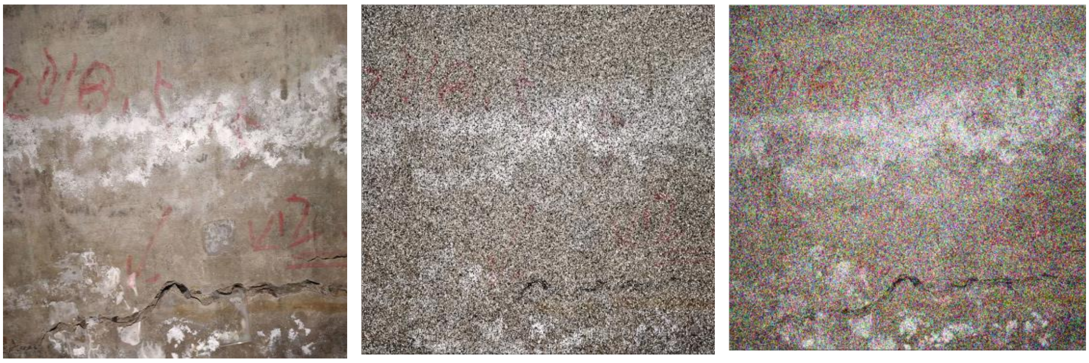

$+ .007 \times$
$=$
$\boldsymbol{x}$
"panda"
57.7% confidence
$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"nematode"
8.2% confidence
$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"gibbon"
99.3 % confidence

Figure 4.17 Noise disturbs the computer vision classification (Goodfellow et al., 2014)

The original image at left contains a panda, while the model recognized it as a panda with 57.7% confidence. However, after adding less than 1% of the noise, the neural network considered it as a gibbon. But from a point of view of human beings, there is no difference between the two pictures. Therefore, the robustness test for the model under noise interference is necessary.

Common image noises include Gaussian Noise, Salt and Pepper Noise, Poisson Noise, etc. Gaussian noise is a kind of noise whose probability density function obeys the Gaussian distribution. Salt and pepper noise, also known as pulse noise, contains salt noise and pepper noise. From the visual perception aspect, salt noise is generally white noise, while pepper noise is generally black noise. The two kinds of noise present in the image at the same time are black and white miscellaneous points. Figure 4.18 demonstrates the original image and image with a different noise.



(a) *(b)* *(c)*

Figure 4.18 Original image and image with different types of noise. (a) Original Image. (b) Image with Salt and Pepper noise. (c) Image with Gaussian Noise.

In order to verify whether the improved Faster R-CNN algorithm is robust when facing the image of the defect with noise, two different types of noises are added into the image sets at a certain rate. The 0.05 proportion of salt and pepper noise and 10ς rate of Gaussian noise are added into two separate image sets, respectively. Then these images with different noise were tested by the detection model to acquire average precision. Figure 4.19 and 4.20 shows the image with Gaussian Noise and Salt and Pepper Noise.

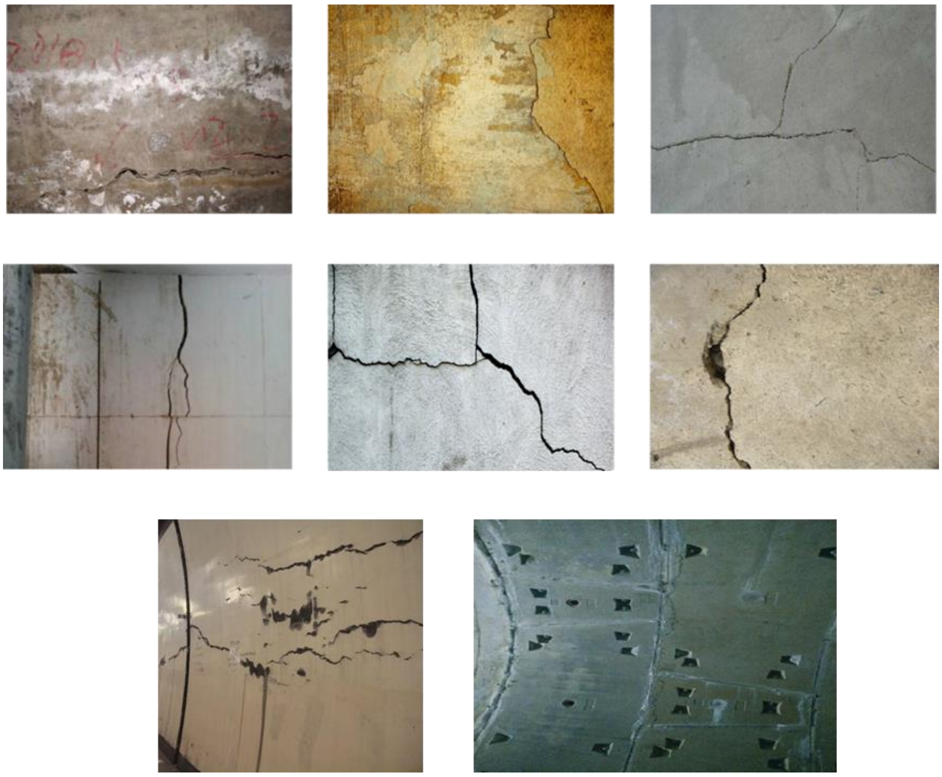


Figure 4.19 Images with Gaussian Noise

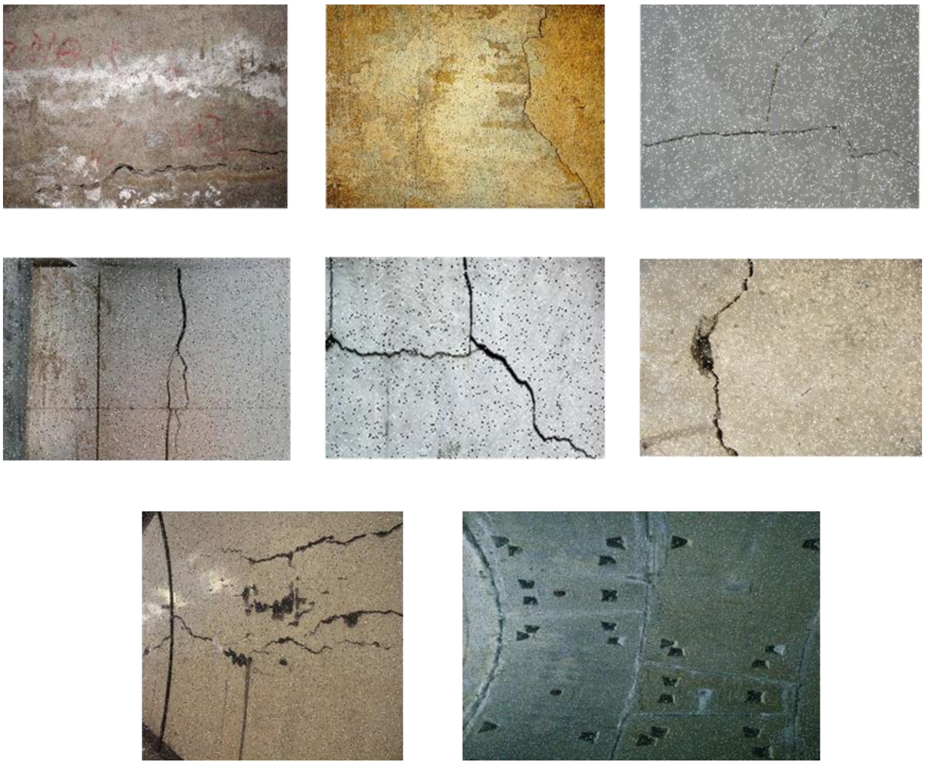


Figure 4.20 Images with Salt and Pepper Noise

After adding the noise into the test sets, Figure 4.21 indicates the AP value is dropped from 0.8873 to 0.7811, and after adding salt and pepper noise, the AP value is dropped from 0.8873 to 0.7433. the detection accuracy of the model is decreased after noises interfere.

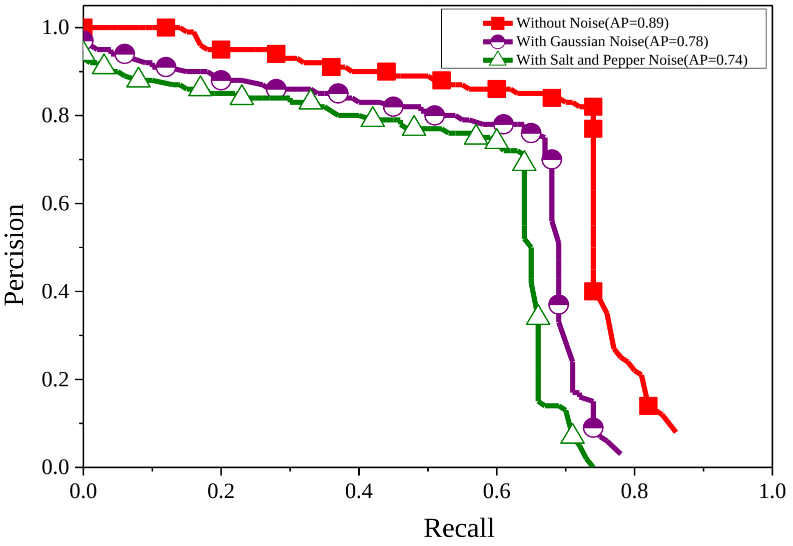
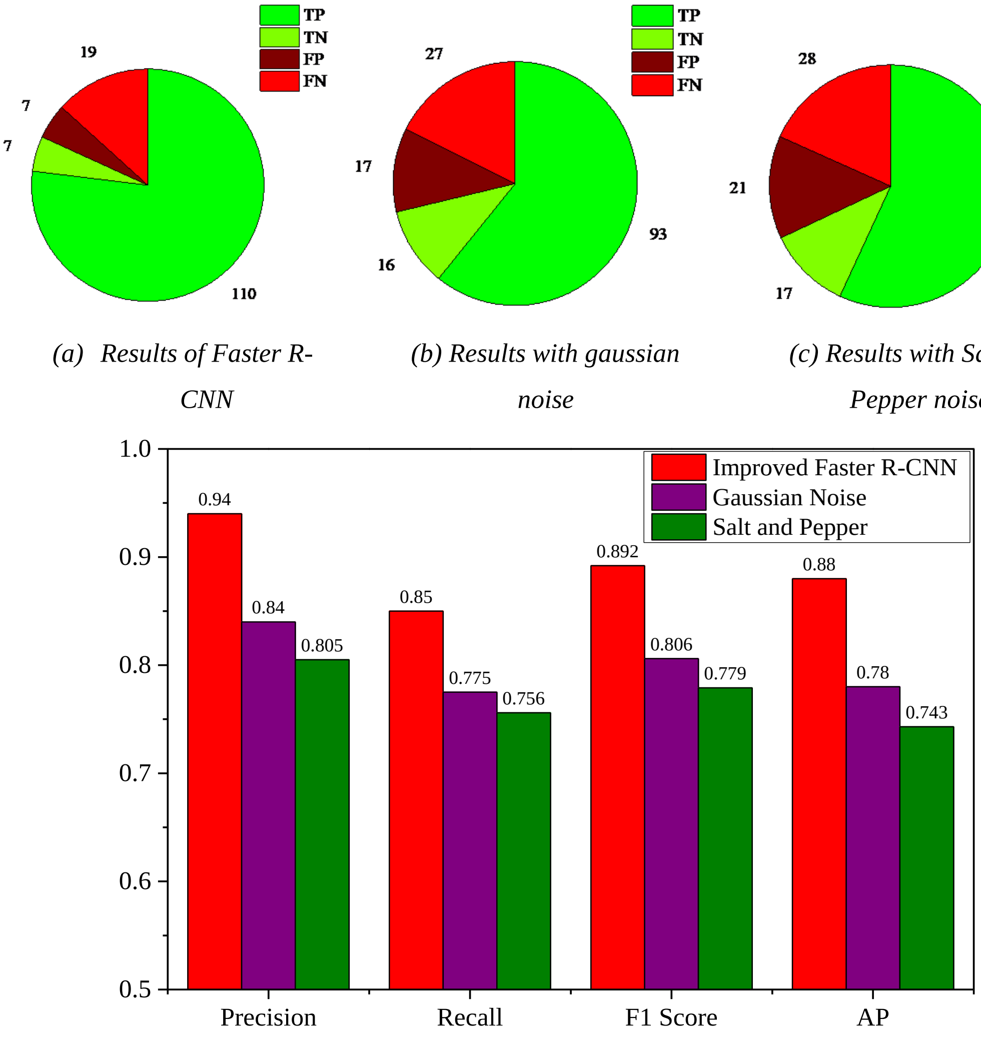


Figure 4.21 P-R curves of the model of the image with Noise

Table 4.4 summarizes the detailed detection results of the cracks and the performance indicators. As shown from the table, the number of TP and TN are decreased while the FP and FN are increased. Four performance indicators (Precision, Recall, F1 score, AP) are all decreased to a varying level.

Table 4.4 Results of the trained model on the image with noise

Noise	**Sample**	**TP**	**FP**	**FN**	**TN**	**Precision**	**Recall**	**F1 Score**	**AP**
Faster R-CNN	143	110	7	19	7	0.94	0.85	0.892	0.88
Gaussian Noise	143	93	17	27	16	0.84	0.775	0.806	0.78
Salt and Pepper	143	87	21	28	17	0.805	0.756	0.779	0.74



(a) Results of Faster R-CNN

(b) Results with gaussian noise

(c) Results with Salt and Pepper noise

(d) Results comparison of R-CNN and image with different noise

Figure 4.22 Detection Results of the image with Noise

Figure 4.22 demonstrates the detection results of the original image and image with noises. It clearly shows that compared with the noiseless data set, the model has lower performance on image sets with noise. Both noises decrease the detection accuracy of the model while the image salt and pepper noise have less performance than the image with Gaussian noise. Figure 4.23 and Figure 4.24 demonstrate the detection results of Gaussian noise and salt and pepper noise.

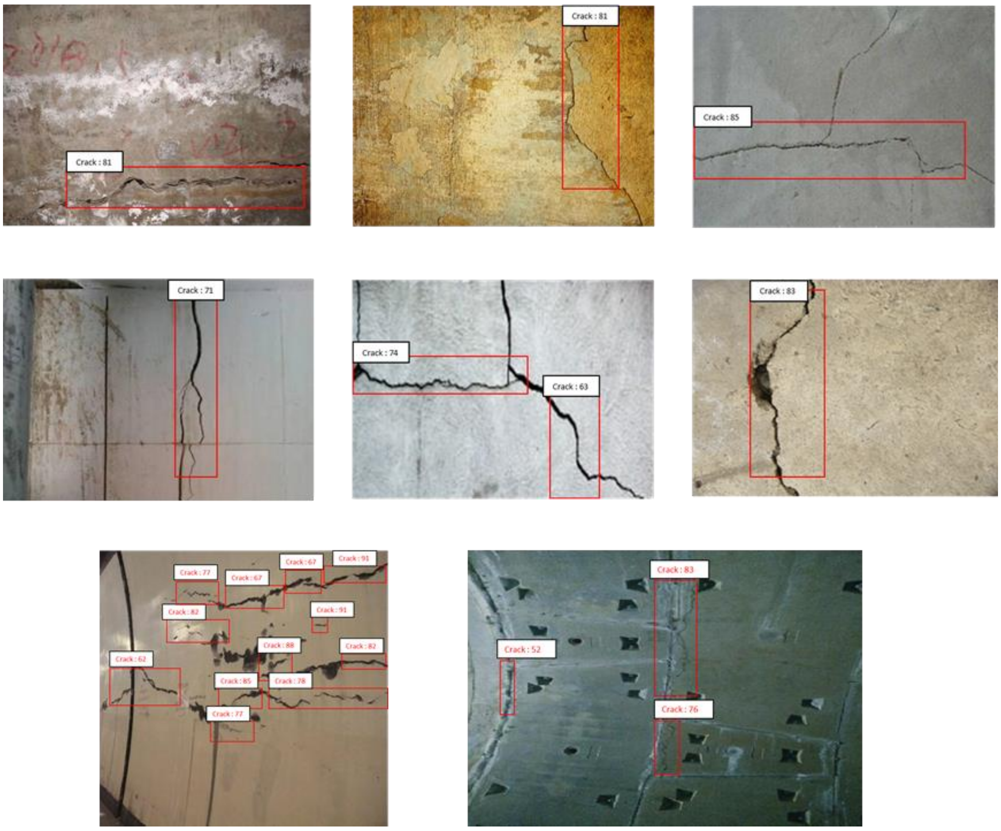


Figure 4.23 Detection results of image with Gaussian Noise

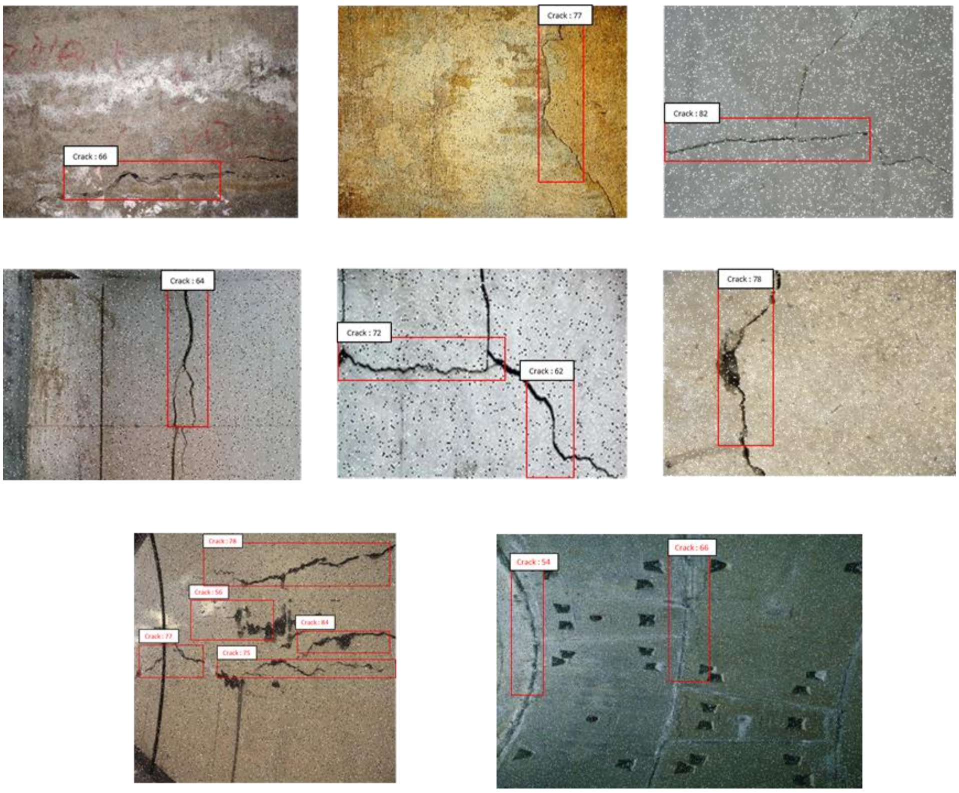


Figure 4.24 Detection results of the image with Salt and Pepper Noise

After adding noise to the test set, the performance index indicated that both noises could decrease the performance of the model. However, according to the AP index, compare with the original data set, the detection accuracy is reduced approximately 10% while the tested image and PR curve graph also indicate that faster R-CNN algorithm has robustness when dealing with noise, the precision of detection under noise disturbance has less variation and shows less affected by the noise, which can resist the noise interference to a certain range and obtain a better result of cracks detection.

4.6 Conclusion

This chapter mainly focuses on the fundamental principles of the two stages algorithms and their application in the detection of cracks of tunnels. The reason for choosing Faster R-CNN is that it is a detection algorithm based on the two-stage, which could focus more processing on the extracted bounding boxes and with further processing. The algorithm sacrifices detection speed in order to achieve excellence of detection accuracy and shows extraordinary robustness under noise. Relative to the previous object detection algorithm such as Fast R-CNN, Faster R-CNN integrated with RPN and adjust the location of RoI Pooling in the network, which provides the algorithm has the ability to share the parameter more adequately hence improve the detection speed and accuracy. The anchor size of RPN in the Faster R-CNN algorithm was improved by combining the characteristics of cracks, and the improved algorithm achieved higher detection accuracy.

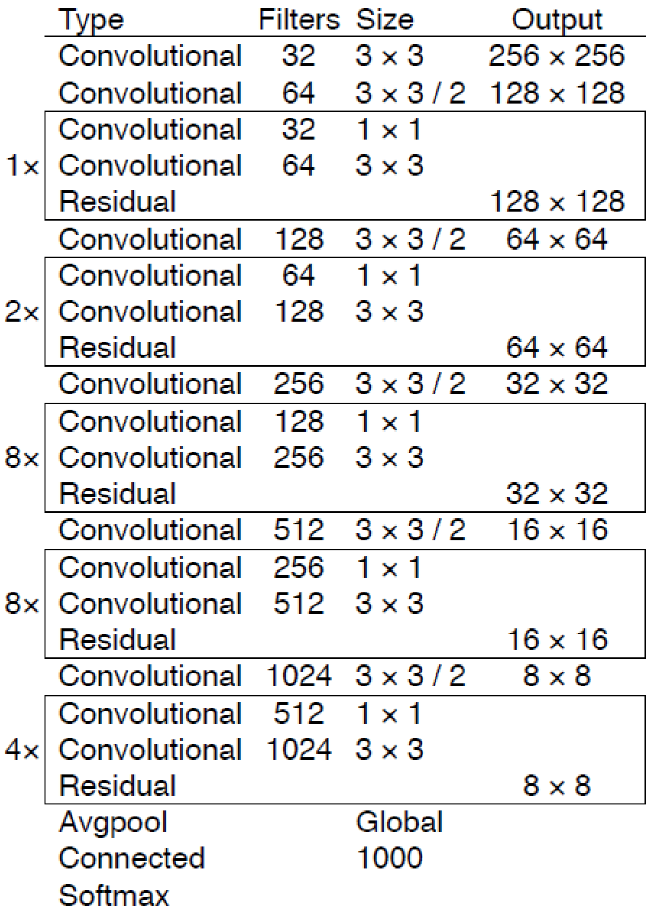
Chapter 5: TBM Segments Defects Detection Based on One Stage Method

Essentially, the Faster R-CNN algorithm is two-stage object detection, the advantage of the two-stage detection method is that it can achieve high detection accuracy. Despite the R-CNN algorithm has higher detection accuracy, its low detection efficiency makes it difficult to apply to real-time monitoring. Therefore, one stage-based You Only Look Once (YOLO) algorithm is proposed. YOLO algorithm is an end-to-end method, it regards object recognition as a regression problem. The advantage of this method is that the category information and position information of the target object in the image can be output after the detection of the neural network. Hence compare with two stage method it has a higher detection speed, which is terrific for real time detection.

5.1 You Only Look Once (YOLO) v3 Algorithm

5.1.1 YOLOv3 Structure

YOLOv3 is a further improvement of the YOLO series algorithm (Redmon and Farhadi, 2018), with the most significant change is the new algorithm use darknet-53 as a backbone network, which is deeper than the previous darknet-19 network. It is named darknet-53 because it has 52 convolutional layers plus one fully connected layer at the end of the network. In version v2, during the process of forwarding propagation of the image, the transformation of its size was carried out through the maximum pooling layer, which was operated 5 times in total. But in the v3 version, the picture in the forward propagation process, through layers of convolution processing, its size is implemented by changing the parameters of convolution kernels of step length. Figure 5.1 shows the structure of Darknet-53.



	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1×	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
2×	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
8×	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8×	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4×	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

Figure 5.1 Structure of Darknet-53 (Redmon and Farhadi, 2018)

The process of YOLOv3 is resizing the input image as a certain size based on setting, then, after five downsampling, the feature image is reduced to 13*13. In short, the backbone network will shrink the output feature image to 1/32 the size of the input image. Figure 5.2 demonstrates the structure of YOLOv3.

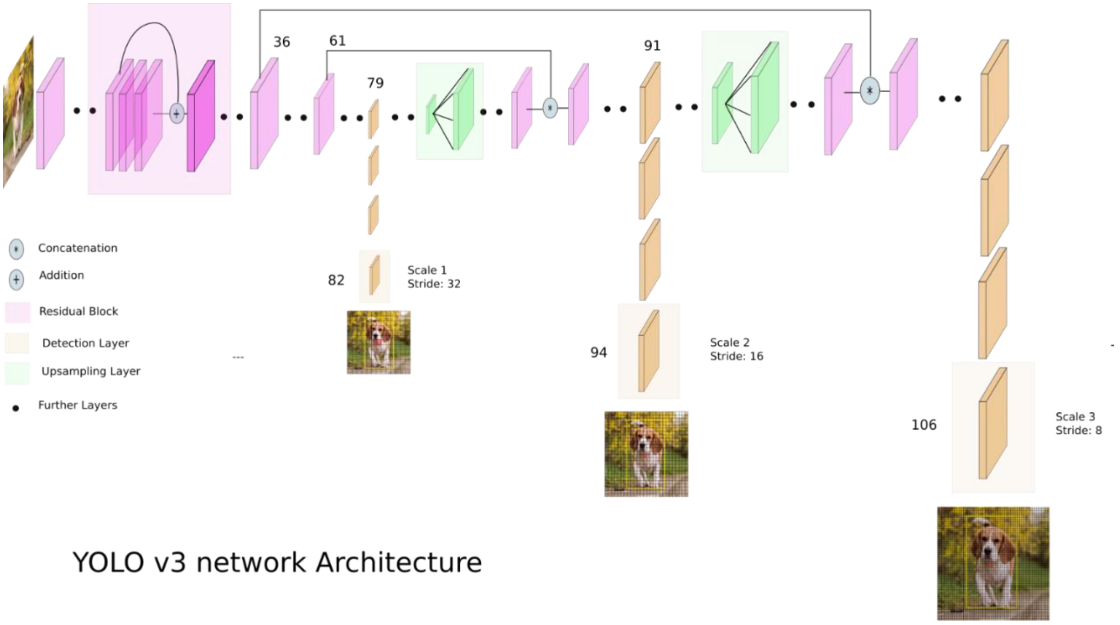


Figure 5.2 Structure of YOLOv3 (Kathuria, 2018)

There only one prediction layer in YOLOv2 while the YOLOv3 version contains 3 prediction layers, the v2 version only uses the top feature images to make predictions, which are less effective on small objects. This is the result of when the feature image becomes small, the receptive field of the neural network becomes larger which is more sensitive to the global information and larger object. Therefore, the v2 version has poor performance detecting small objects while facing large scale feature images due to it has a relatively smaller receptive field.

YOLO v3 adopts upsample and fusion method (similar to FPN), fusing 3 scales (13*13, 26*26 and 52*52), independently testing on the fusion feature map of multiple scales, and finally significantly improving the detection performance on small objects.

In YOLOv3, the anchor box is increased from 5 to 9, while its initial value is still generated by the k-means clustering algorithm. Three anchor boxes are allocated at each scale, and three bounding boxes are predicted for each cell (corresponding to three anchor boxes). Each cell outputs (1+4+C) *3 values (4 positioning information, 1 confidence score, and C conditional category probabilities), which is also the depth of the final output feature tensor at each scale. Although YOLOv3 predicts three bounding boxes per cell, there are many more bounding boxes than in previous versions because YOLOv3 uses multi-scale feature fusion (the same number of bounding boxes need to be predicted at each scale). Figure 5.3 shows the multi-scale on images in YOLOv3.

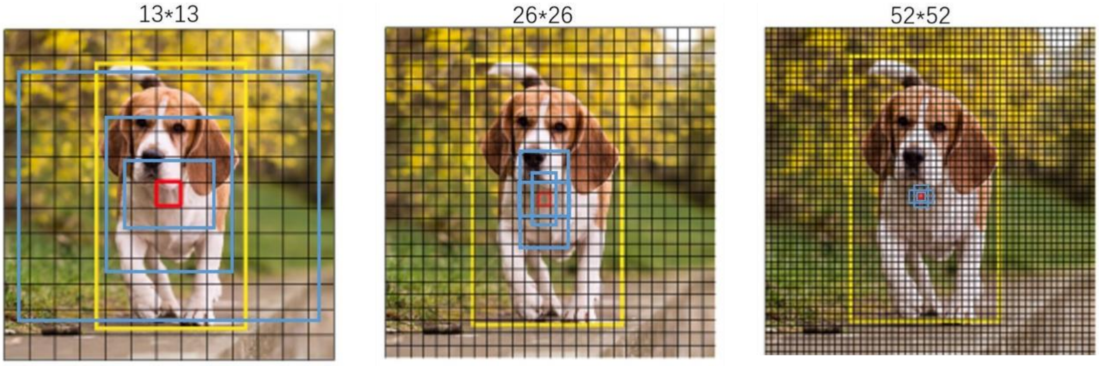


Figure 5.3 Multi-scale on images in YOLOv3 (Kathuria, 2018)

Taking the input image as 416*416 as an example, a picture in YOLOv2 needs to predict 13*13*5=845 bounding boxes, while in YOLOv3 needs to predict (13*13+26*26+52*52) *3=10647 bounding boxes.

YOLOv3 contains many residual units, and it also uses the feature pyramid network idea for reference. The network structure can be made deeper and stable while the gradient would not disappear during model training because the residual units contain skip structure functions. Due to the skip structure function of these modules, the network structure can be made deep, and the gradient will not disappear, making the network structure more robust. And because of the deep hierarchy of the network structure and the multi-scale feature map fusion method, the network has a stronger ability to extract and learn features from images and can extract crack features from multiple scales to make the network learn more fully. Therefore, the network architecture of YOLOv3 is better than the previous version. With the deeper network structure and the multi-scale feature image fusion method, the network has a stronger ability to extract and learn features from images and can extract more features from multi-scales to make the model with higher leaning ability. Therefore, the network architecture of YOLOv3 is better than the previous version which has better performance on object detection.

5.1.2 Residual Neural Network

Residual Neural Network (ResNet) was first proposed by He et al. (2016), and its basic unit is the residual block. The residual network proposed to solving the problem in a deeper network when increasing the layer of the network structure, the accuracy of the training set will not rise but it will decrease. While this phenomenon has nothing related to overfit, because if it is caused by overfitting, then model accuracy of the network in training should appear extremely high, therefore it may cause disappear gradient and makes the network training process cannot continue, lead to the training set lose accurately. Compared with the conventional plain network structure, residual networks add the skip connections between every two layers, form a residual block so that later layers can learn residuals directly from the previous. This network structure can form a deep residual network and solve the problem of decreasing accuracy during model training. Figure 5.4 demonstrates the structure of the Residual Neural Network.

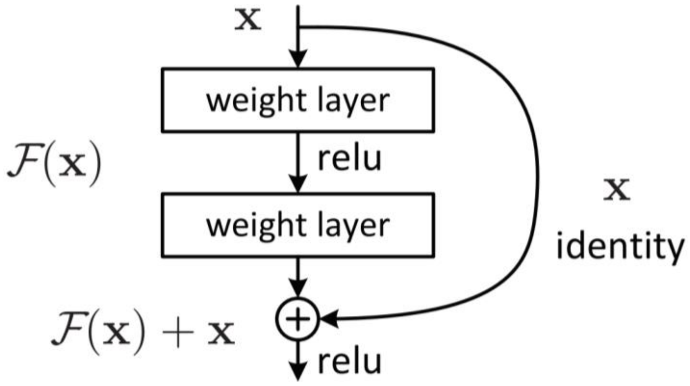

x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Figure 5.4 Structure of Residual Neural Network (He et al., 2016)

There are two mappings in ResNet, one is the identity mapping, which refers to the input data x itself, which is represented as a curve in the figure, and the other is the residual mapping, which refers to the rest part of the network. The advantage of ResNet is the network structure contains skip connection, make sure the network could be trained normally while the gradient would not disappear, the layer of the convolutional neural network can be deeper and the error rate of network training will not increase.

5.1.3 Improvement of YOLOv3 for Cracks Detection

Compare with Faster R-CNN, the YOLO algorithm has a different structure, and it has no regional proposal. Therefore, the detection accuracy of YOLO is heavily dependent on grid size. Finer grid size provides higher detection accuracy for small objects. Considering that the size of the bounding box and grid size in YOLOv3, is relatively large for detecting cracks while most of the cracks are thin and long, therefore, one extra finer layer in FPN is added to extract more detailed information to provide higher accuracy of detected smaller objects. Figure 5.5 demonstrates the improved FPN in YOLO V3.

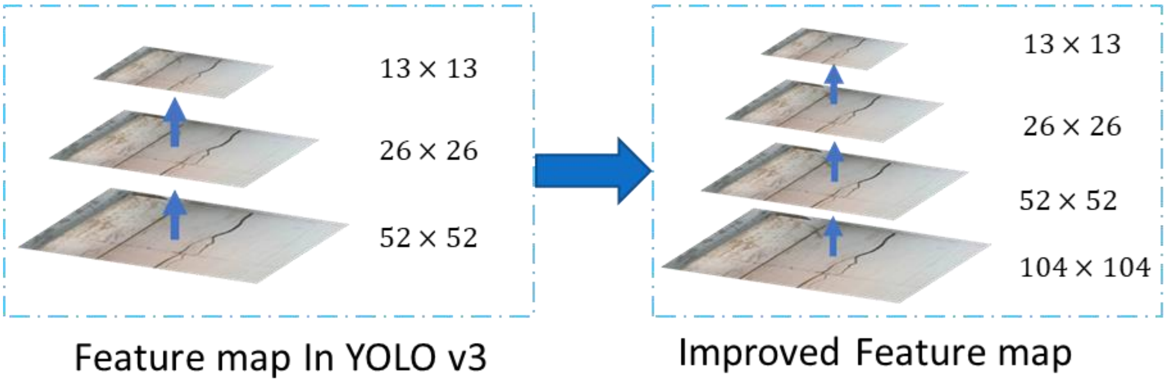


Figure 5.5 Improvement of YOLO V3 model

Originally, the feature map in YOLO v3 is three layers, while the size is 13x13, 26x26, 52x52. one extra layer is added to the feature map at the bottom with size 104x104. The finer feature map is used to achieve finer feature extraction in order to make sure model that could detect the cracks accurately. Figure 5.6 shows the structure of YOLOv3 for crack detection.

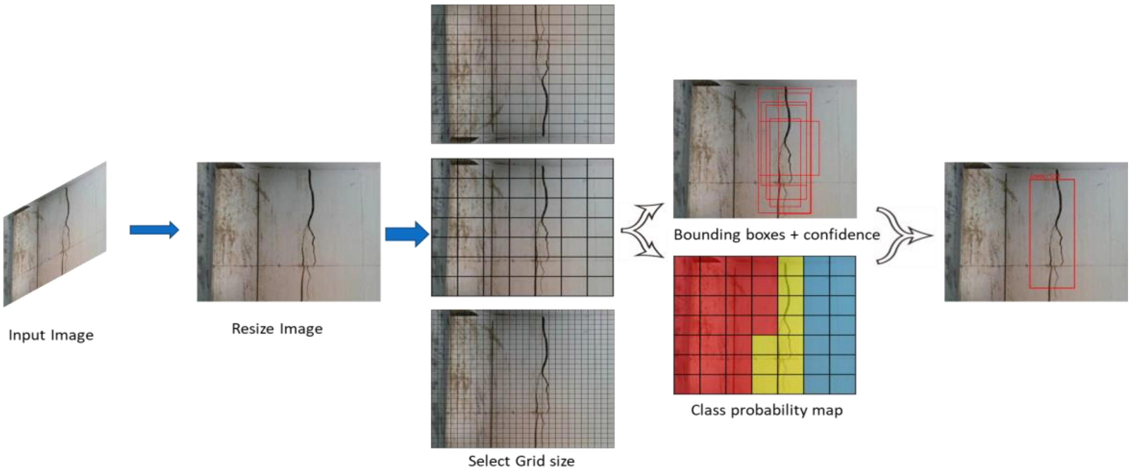
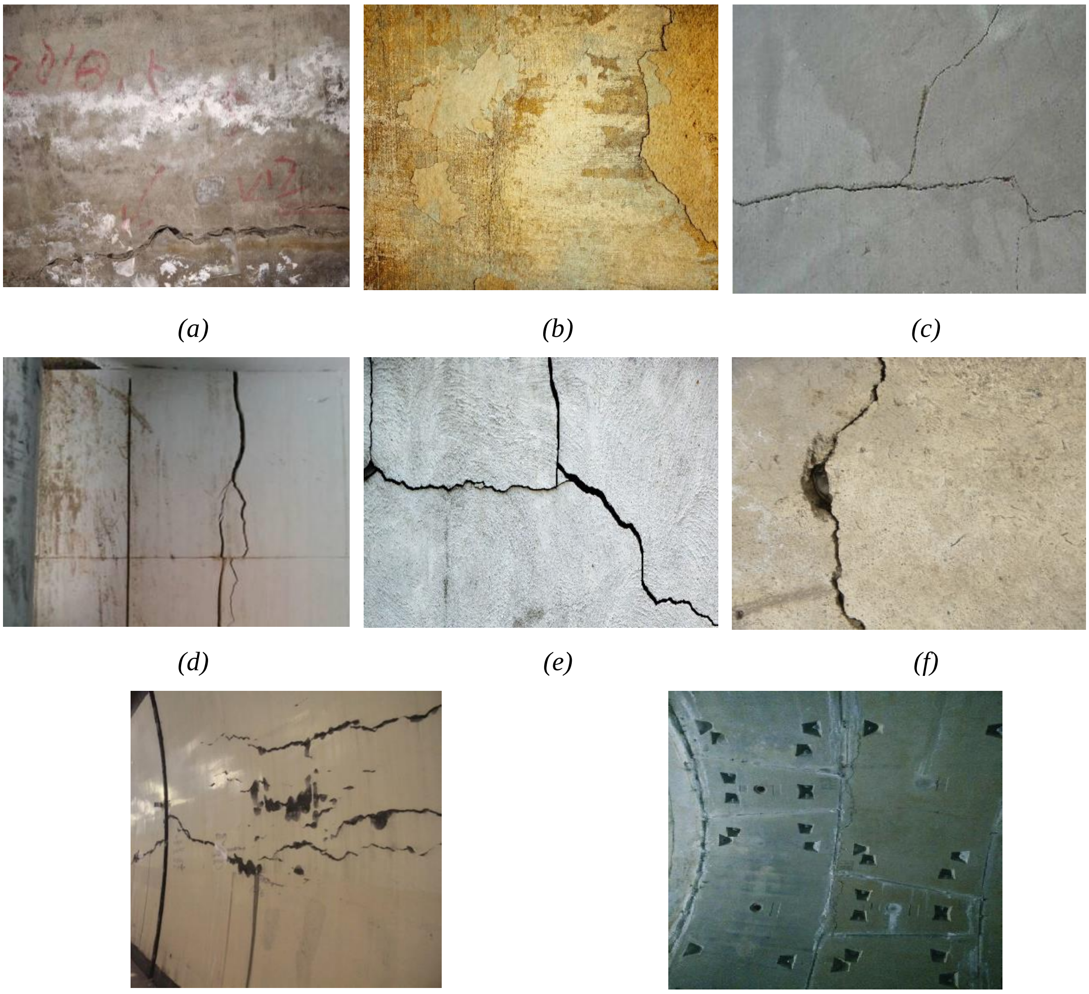


Figure 5.6 Structure of YOLOv3 for cracks detection

Moreover, the YOLO v3 algorithm is also to be tested. The reason for choose YOLOv3 is that it is a detection algorithm based on the one-stage, which could focus on detection speed and could meet the real-time monitoring requirement. Relative to the previous object detection algorithm such as YOLOv2 and SSD, v3 integrated with RPN and ResNet in the network, which provides the algorithm can build deeper network without gradient disappear hence improve the detection accuracy on the small objects, to achieve higher precision of cracks detection.

5.2 Image Database, Model Training

The Deep learning platform and data based are the same as Faster R-CNN one and the image database have remained the same. The image database of cracks contains 7143 images and 7000 images were selected as the training database for network training while the remaining 143 were used for blind test of the model. In order to better train the model and improve the accuracy, 6500 of the 7000 training images were used as the training set and the remaining 1500 as the validation set. Figure 5.7 shows the test set of cracks.



(a) *(b)* *(c)*

(d) *(e)* *(f)*

(g) *(h)*

Figure 5.7 Test sets of cracks (a) Image of cracks with handwriting and white paint. (b) Image of cracks with wall peeling off. (c) Image of cracks with two different propagation direction. (d) Image of cracks with wall joint. (e) Image of cracks with multi direction. (f) Image of cracks with hole. (g) Image of tunnel with large scale cracks (DANANGToday, 2017). (h) Image of TBM segments with cracks

5.3 Results of YOLO v3

Before optimized the YOLOv3 algorithm, the setting of training iterations, step length, learning rate and other parameters required by the algorithm is set, then the initial YOLOv3 training result model can be obtained during the training of the neural network, and then the experimental result can be used as the baseline. After improving the YOLOv3 algorithm, the results can be also obtained for comparison and the new trained model can be used for cracks detection of tunnels. The following shows the detection performance of the improved YOLOv3 algorithm on some tunnel cracks, as shown in Figure 5.8.

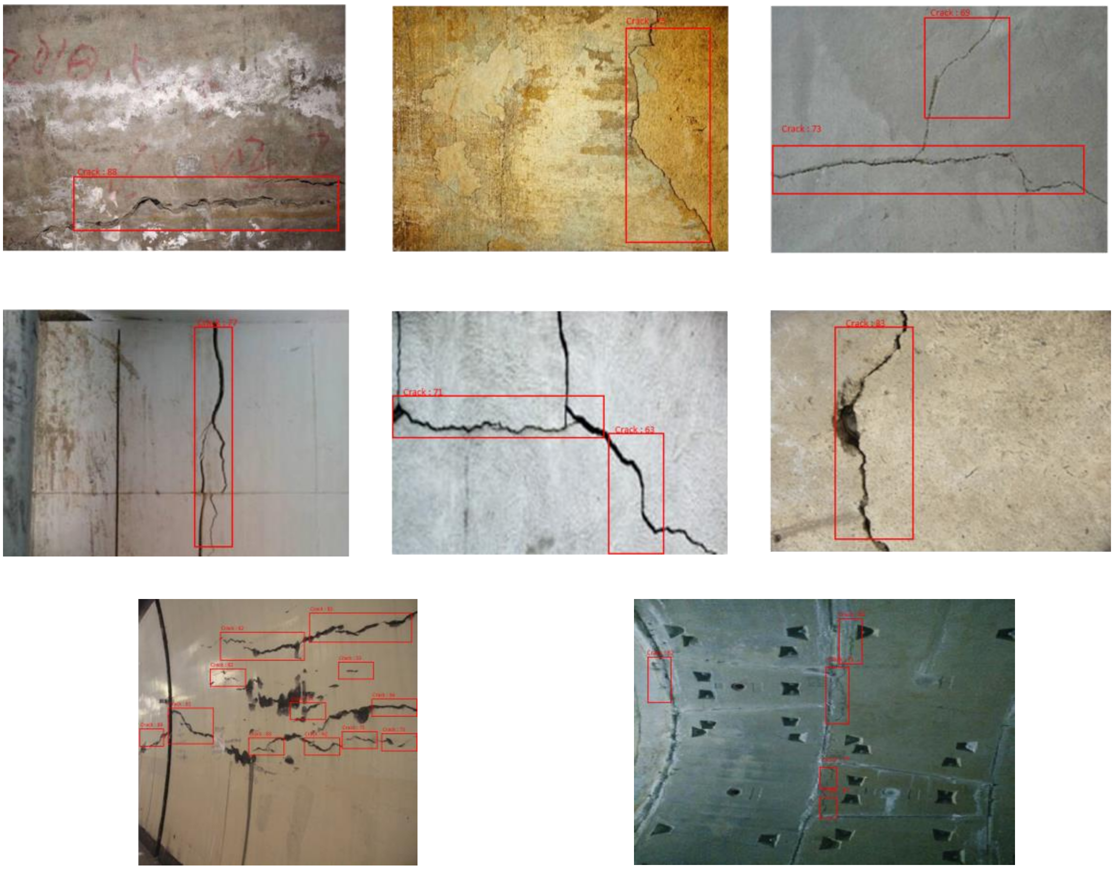


Figure 5.8 Detection performance of the model on the test set

Conduct visual analysis on the training log files generated during the training process. The curve of the Loss function chart during the training process and the final Precision and Recall (PR) curve can be obtained. The meaning of the Loss curve refers to the change in the value of the Loss function during the training process. The trend of the change is continuously downward from high to low, indicating that the training process is gradually tending to the state of convergence, and the training model in the state of convergence has a better detection effect. Figure 5.9 shows the Loss Curves of YOLOv3 during model training.

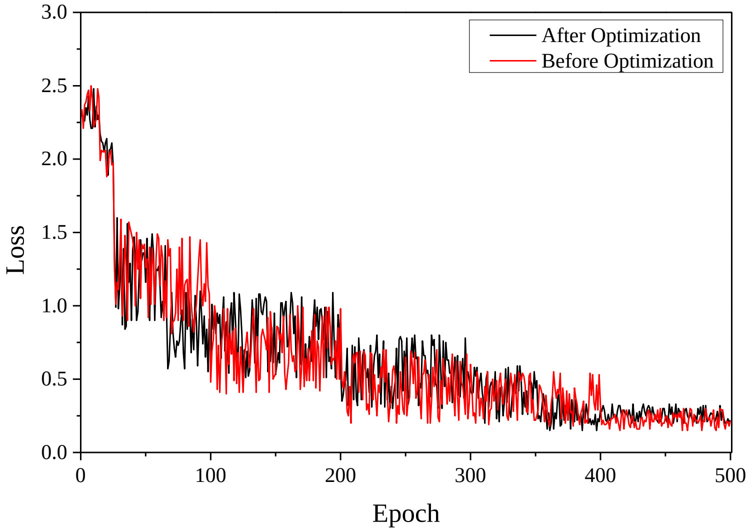


Figure 5.9 YOLOv3 Loss Curves

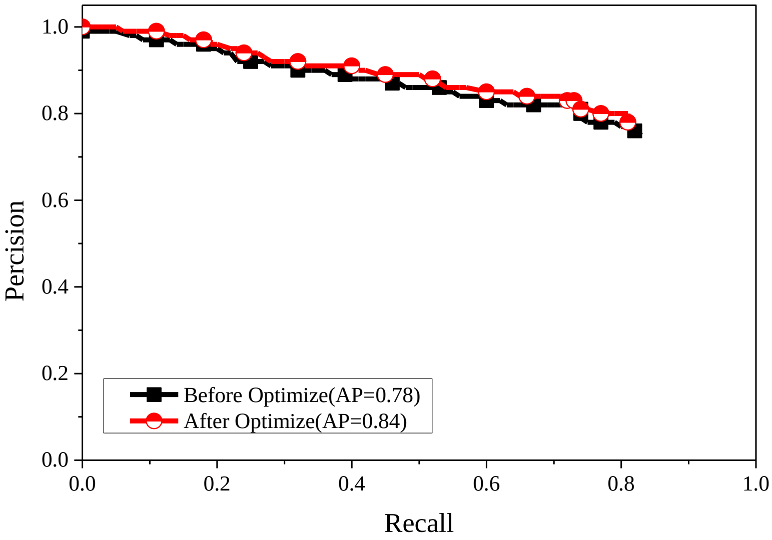


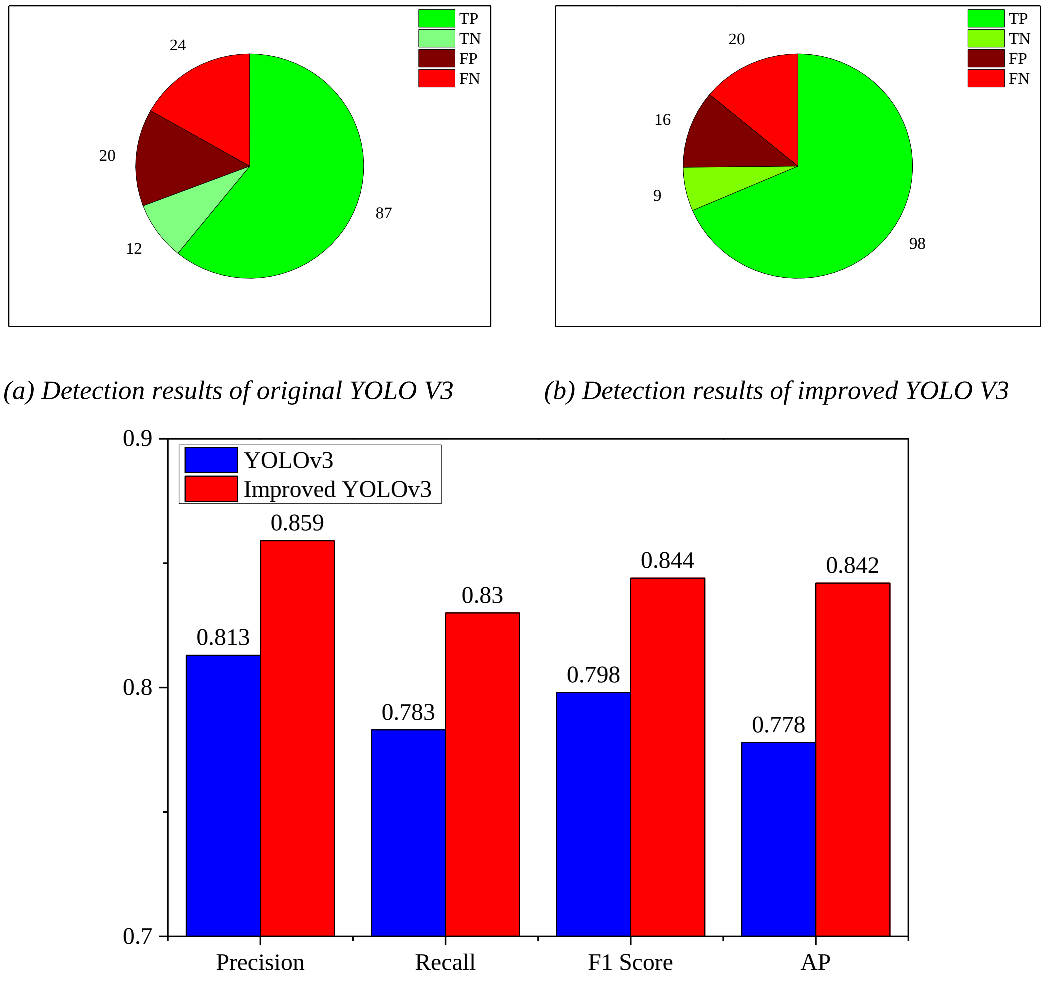
Figure 5.10 YOLOv3 P-R Curves

Figure 5.10 shows the P-R Curve of YOLOv3 during model training. The definition of PR curve is the curve of accuracy and recall, the abscissa is the recall rate (which ranges from 0 to 1) while the vertical coordinate is the precision (with values ranging from 0 to 1). When drawing the PR curve, points are divided according to the samples of each test set as the threshold, different recall rate and precision rate values can be obtained, then connect these different points and a two-dimensional PR curve are obtained. The area enclosed between the PR graph, the x-coordinate axis and the y-coordinate axis is the value of AP, which is a comprehensive index used to measure the detection effect of the object. The larger the value, the better the performance of the detection algorithm. Table 5.1 summarizes the results of the trained model on test sets.

Table 5.1 Results of the trained model on tests sets

Algorithm	**Sample**	**TP**	**FP**	**FN**	**TN**	**Precision**	**Recall**	**F1 Score**	**AP**
YOLOv3	143	87	20	24	12	0.813	0.783	0.798	0.78
Improved YOLOv3	143	98	16	20	9	0.859	0.83	0.844	0.84

As shown from Table 5.1, compared with the original algorithm, the number of TP and TN are increased while the FP and FN are decreased. Four performance indicators (Precision, Recall, F1 score, AP) are all increased to varying levels.



(a) Detection results of original YOLO V3

(b) Detection results of improved YOLO V3

(c) Results comparison of original YOLO and improved YOLO

Figure 5.11 Detection results of YOLOv3

Figure 5.11 demonstrates the detection results of the original model and the improved model. It clearly shows that compared with the original one, the improved model has higher performance on image sets. Both four indicators are increased which indicates the improvement of the model could increase the accuracy of crack detection and achieve higher performance.

After optimized the YOLOv3, the performance index indicated that improvement could increase the performance of the model. Moreover, according to the AP index, compare with the original data, the detection accuracy is increased approximately 8% while the tested image and PR curve graph also indicate that YOLO algorithm has a better performance compared with the original one. Overall, the improved YOLO achieved higher detection performance on cracks.

5.4 Robustness Test

In the underground engineering project, due to the environmental effect and lighting conditions, the image is vulnerable to all kinds of interference, resulting in a decline in the detection effect. Image noise is some isolated pixels in an image, which will disturb the actual visibility of the image, making the image fuzzy. Common image noises include Gaussian Noise, Salt and Pepper Noise, Poisson Noise, etc. Gaussian noise is a kind of noise whose probability density function obeys the Gaussian distribution. Salt and pepper noise, also known as pulse noise, it contains salt noise and pepper noise. From the visual perception aspect, salt noise is generally white noise, while pepper Noise is generally black noise. The two kinds of noise present on the image at the same time are black and white miscellaneous points. Figure 5.12 and 5.13 demonstrate the image with Gaussian Noise and Salt and Pepper Noise.

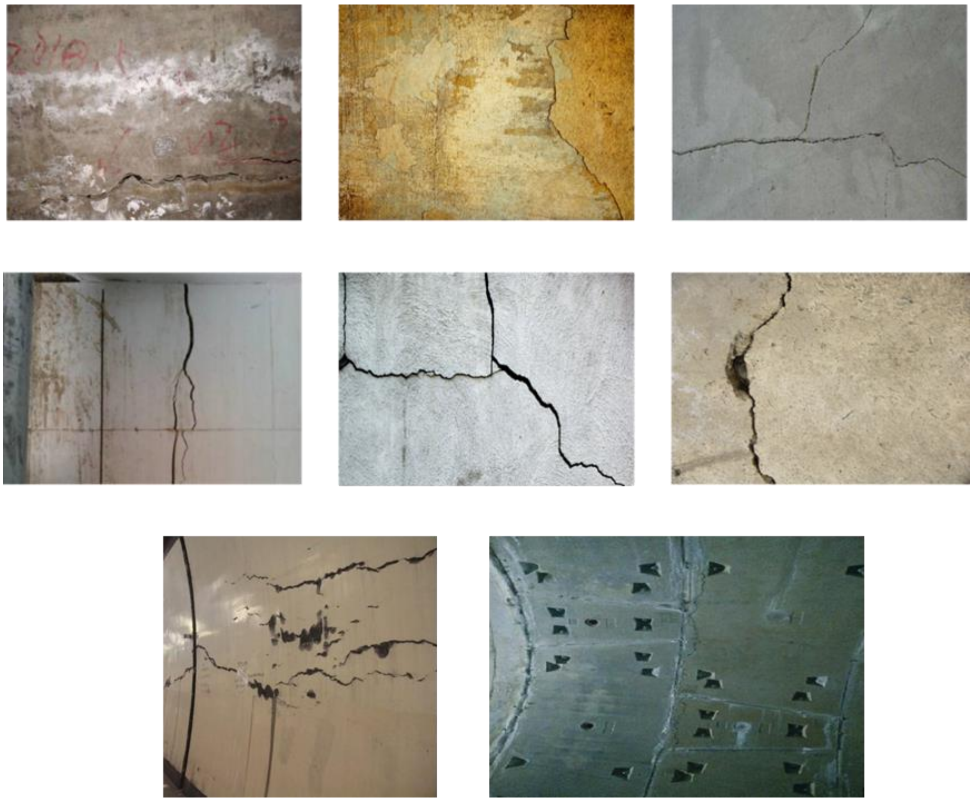


Figure 5.12 Images with Gaussian Noise

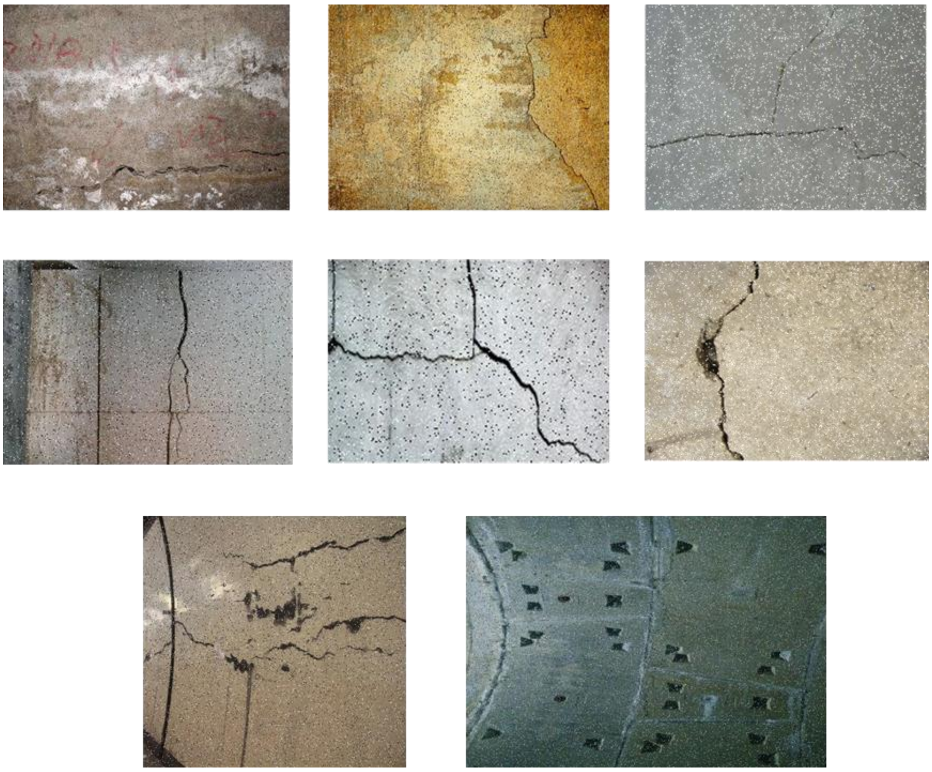


Figure 5.13 Images with Salt and Pepper Noise

In order to verify whether the improved YOLOv3 algorithm is robust when face the image of the defect with noise, gaussian noise and salt and pepper noise were added to all of the test set images and the test sets is same as Faster R-CNN, then these images with different noise were tested by the detection model to acquire average precision.

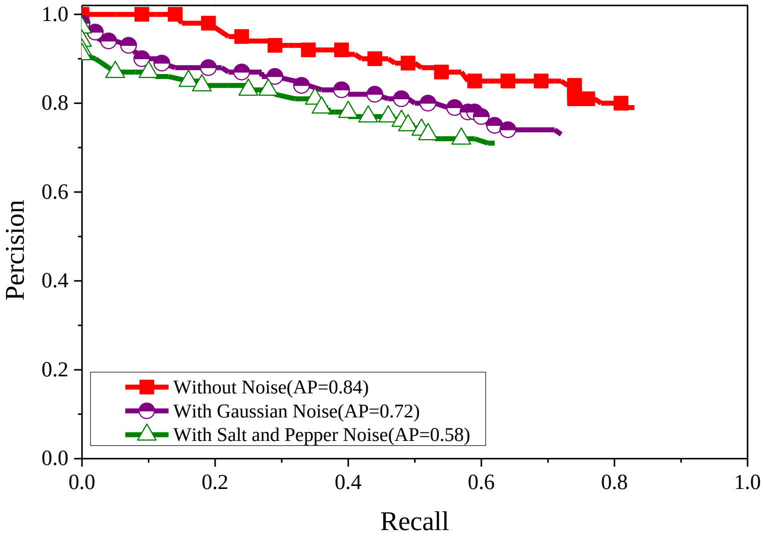


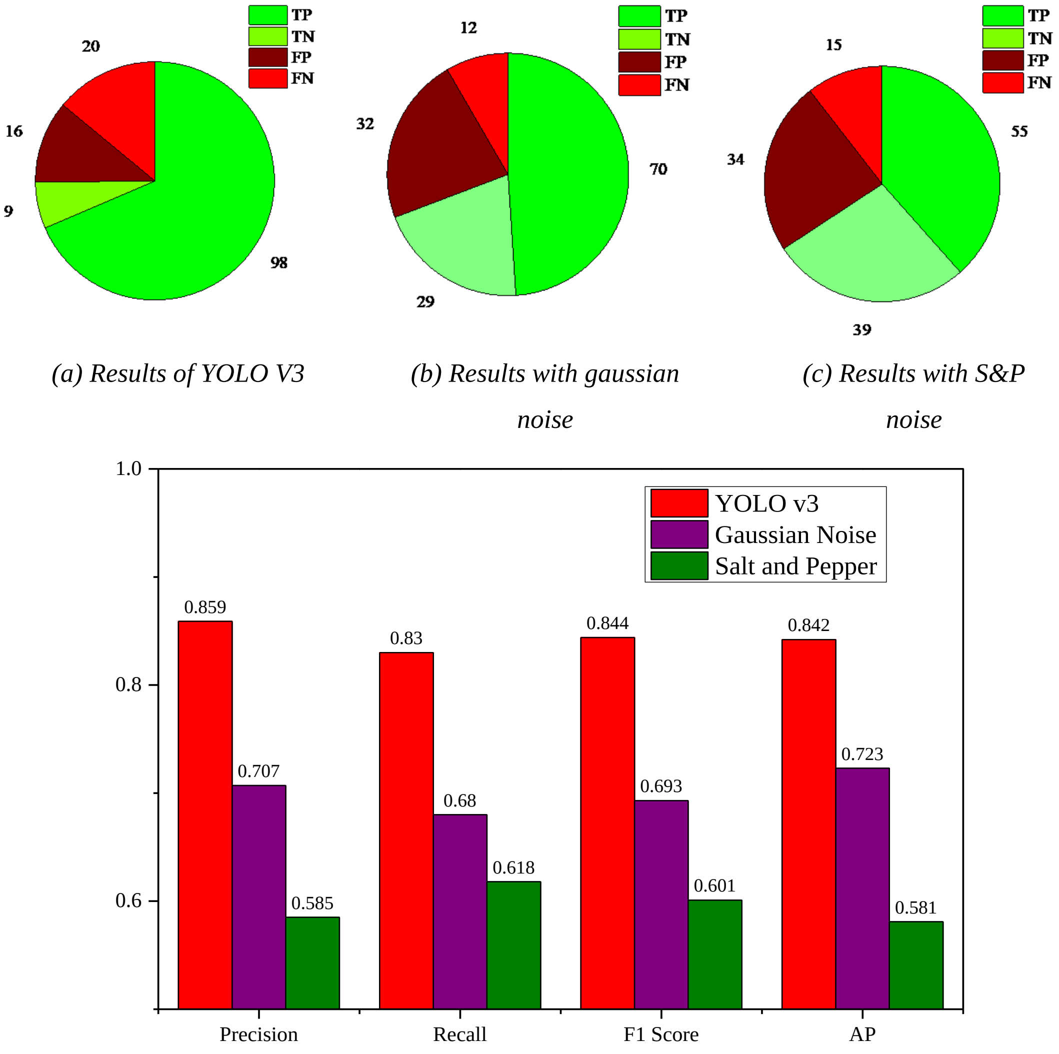
Figure 5.14 P-R curves of the model of the image with Noise

Figure 5.14 indicates after adding gaussian noise to the test image, the AP value is dropped from 0.8421 to 0.723, and after adding salt and pepper noise, the AP value is dropped from 0.8421 to 0.5813, and Figure 5.15 shows the detection results of the cracks. Table 5.2 summarizes the results of the trained model on test sets.

Table 5.2 Results of the trained model on the image with noise

Noise	**Sample**	**TP**	**FP**	**FN**	**TN**	**Precision**	**Recall**	**F1 Score**	**AP**
YOLOv3	143	98	16	20	9	0.859	0.83	0.844	0.842
Gaussian Noise	143	70	29	32	12	0.707	0.68	0.693	0.723
Salt and Pepper	143	55	39	34	15	0.585	0.618	0.601	0.581

Table 5.2 summarizes the detailed detection results of the cracks and the performance indicators. As shown from table, the number of TP and TN are decreased while the FP and FN are increased. Four performance indicators (Precision, Recall, F1 score, AP) are all decreased to varying level.



(a) Results of YOLO V3

(b) Results with gaussian noise

(c) Results with S&P noise

(d) Results comparison of YOLO and image with different noise

Figure 5.15 Detection Results of YOLO v3 for the image with Noise

Figure 5.15 demonstrates the detection results of original image and image with noises. It clearly shows that compared with the noiseless data set, the model has lower performance on image sets with noise. Both noises decrease the detection accuracy of the model while the image salt and pepper noise have less performance than image with gaussian noise. Figure 5.16 and 5.17 demonstrates the detection results of gaussian noise and salt and pepper noise, respectively.

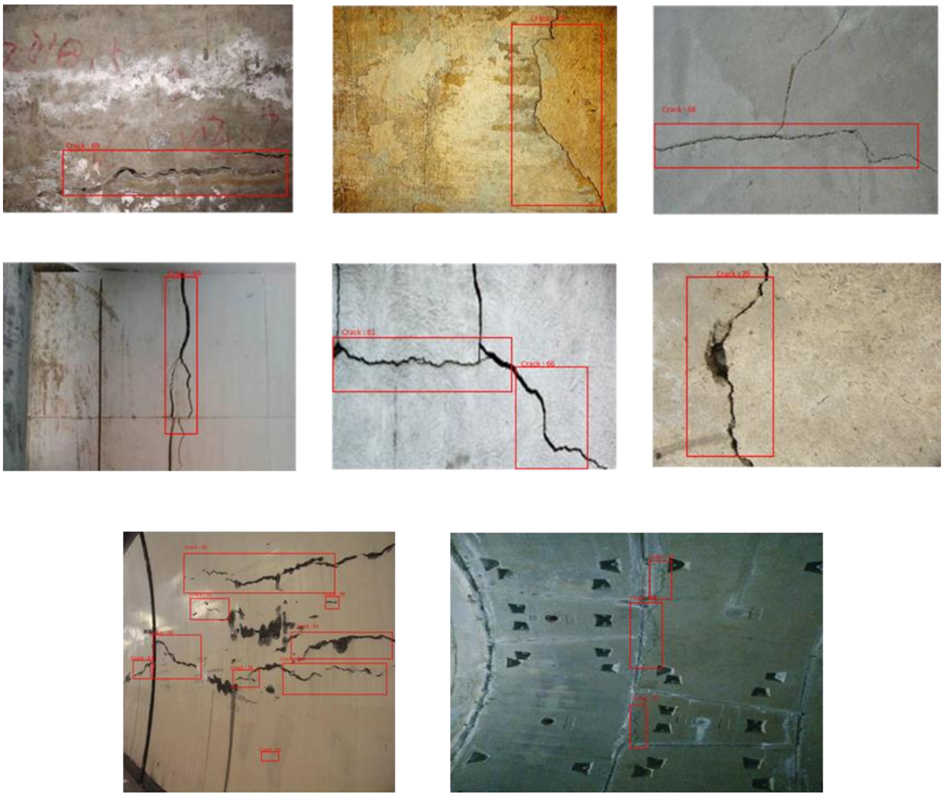


Figure 5.16 Detection results of image with Gaussian Noise

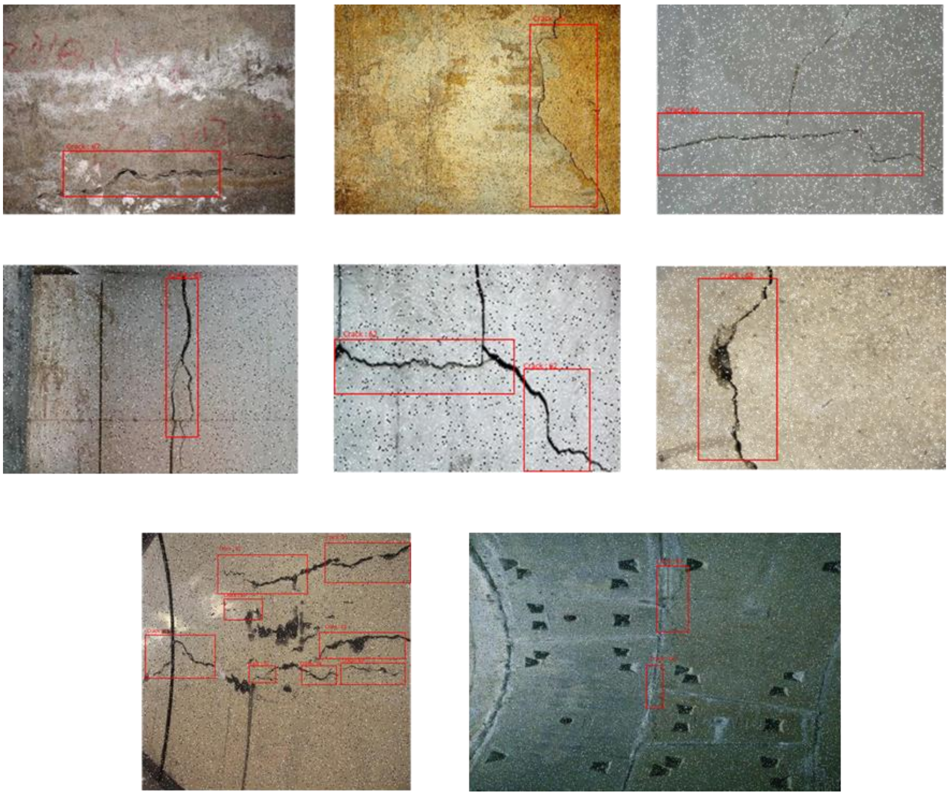


Figure 5.17 Detection results of the image with Salt and Pepper Noise

After adding noise to the test set, the performance index indicated that both noises could decrease the performance of the model. Moreover, according to the AP index, compare with the original data set, the detection accuracy is reduced approximately 31% while the tested image and PR curve graph also indicate that YOLOv3 algorithm has poor robustness when dealing with noise, the precision of detection under noise disturbance has higher variation and shows it is affected by the noise, which could not resist the noise interference to a certain range and obtain an unsatisfactory result of cracks detection.

5.5 Conclusion

This chapter mainly focus on the fundamental principles of YOLOv3 algorithm and its application in the detection of cracks of tunnels. The reason for choose YOLOv3 is that it is detection algorithm based on the one-stage, which could focus detection speed and could meet the real-time monitoring requirement. Relative to the previous object detection algorithm such as YOLOv2 and SSD, YOLOv3 integrated with RPN and ResNet in network, which provide the algorithm has ability to build deeper network without gradient disappear hence improve the detection accuracy on small object, to achieve higher precision of cracks detection.

Chapter 6: Conclusions and Future Work

6.1 Conclusions

This thesis focuses on the application of deep neural networks in underground projects, and has two major applications. One TBM performance prediction by using neural networks and another is using neural networks to extract the feature of underground space cracks, identify and locate the cracks on the TBM lining and the tunnels. Through the research, the following conclusions are obtained:

A literature review has been done on TBM performance prediction by using the neural networks and proposed a method of prediction for TBM performance and rock mass parameter by input multiple feature data into a neural network based on the LSTM model. By adjust the specification of the neural network model, the number of neurons, the timesteps, the learning rate, the random dropout rate and use optimization function to improve the prediction accuracy of the model. In consideration of the lithology, surrounding rock type and groundwater activity of the excavation scope, through the machine learning of TBM data and rock data, the equipment operating parameters of the TBM stable operating section are predicted. Based on the existing TBM operation data, the improved LSTM machine learning method based on RNN can effectively predict the TBM tunneling parameters. Overall, the proposed model is extendable and can be suited for other relatively engineering aspects.

Moreover, based on the research of image recognition and combining with the deep learning algorithm, this thesis conducts model training based on large amount of image sets, determines whether there are cracks in the image and labels the position of cracks in the complex image. It can provide a novel technical theory for high precision intelligent detection of complex tunnel defects which has good research and development potential. Furthermore, based on the different requirements of purpose for deep learning detection algorithms, this thesis adopts two different deep learning frameworks. Pytorch is used for the implementation of the Faster R-CNN algorithm, while the YOLOv3 algorithm use darknet-53 which is a relatively lightweight Keras based deep learning framework. Both frameworks have their own advantages and disadvantages, therefore, for specific object detection algorithms, it is necessary to adopt the corresponding deep learning framework platform for these algorithms to achieve their maximum potential, and obtain better performance on detecting the cracks in the underground space. Compared with the conventional crack detection method based on machine learning, the method adopted in this thesis is more efficient and accurate. The final results of the comparative test indicate the accurate identification and real-time localization of the underground crack, which can meet the real-time detection requirements.

The Faster R-CNN algorithm is applied to TBM lining defects detection in underground space and its specific parameters are adjusted to improve the detection performance. Faster R-CNN is an algorithm based on the Fast R-CNN algorithm with adjusted and optimized network structure. From structure aspect, Faster R-CNN belongs to the two-stage algorithm, and compare with the previous two-stage algorithm, Faster R-CNN has integrated feature extraction, proposal extraction, bounding box regression and classification into one network, which significantly improves the comprehensive performance, and improves the detection accuracy and speed of cracks in underground space. The research mainly combined with the characteristic of tunnel cracks itself which is relatively thin and long, improves the Faster R-CNN algorithm and optimized the specific parameter. The results of test indicate that the improved algorithm has better performance on crack detection. Moreover, due to the environmental condition of the underground site, the collected image of tunnels may exist noise interference and out of clarity. Therefore, the noise is added to the image of the test set before detection in order to test the robustness of the algorithm. The detection results show that the Faster R-CNN algorithm is robust when facing images with noise.

The YOLOv3 algorithm is applied to crack detection in underground space. YOLO is a typical representative algorithm of the one stage method, it outputs the category of the object and its corresponding position based on the one stage idea, therefore its detection speed is faster than those algorithms based on the regional proposal method which could satisfy the real time monitoring requirement. Compared with YOLO v2, YOLO v3 has more detection units. It improves the network architecture and replaces the loss function in v2 (from SoftMax loss to logistic loss), as a result, YOLOv3 has better detection performance on small objects while maintaining the same detection speed. Moreover, Gaussian noise and salt and pepper noise are added to all cracks test images before detection to test the robustness of the YOLOv3 algorithm. The results show that compared with the two stage Faster R-CNN algorithm, the YOLOv3 algorithm based on one stage is less robust to the image of the crack containing noise.

6.2 Application of Deep Learning in TBM and Tunnel Inspection

The objective of this research is to apply deep learning to TBM and tunnel inspection. For the construction of tunnels by TBM, the application of deep learning based on the prediction system could divide into different stages. In the initial stage, the operating data of TBM is collected by the sensors and send to the analysis center, the data is analysed by the trained model based on the existing TBM operating model. Based on the current data the model will output the prediction of TBM for the operator to assist the decision. In the next stage, the TBM will integrate the prediction system that currently collected data could use for real time training of the model without exchange data to the analysis center. The ultimate objective of the prediction system is, the TBM subsystem including prediction and AI assisted decision making method could be developed to assist self-adjustment for TBM operating parameters which make TBM could operating automatically to make underground tunnelling excavation economical and prevent potential hazards.

For tunnel inspection, the application of computer vision-based inspection can be divided into different stages. For the initial stage, the conventional inspection is performed during the maintenance period by the manual inspection. A mobile device integrated with the vision device collected the image of the lining during the maintenance period and send it to the data analysis center to detect and mark the cracks in the image. With further technology development, in the next stage, the device integrated with the image processing unit that could detect the cracks of linings without sending the data to the analysis center. The image processing unit integrated on the devices could automatically analysis the video, detect and mark the location of the cracks while the operator could use a cellular data exchange system to remotely control and monitoring the device with streaming video transmission. The ultimate development of the device is, the Unmanned Aerial Vehicle integrates with the vision and image processing unit that could automatically inspect the tunnel according to the preset routine and detect, mark the cracks without any manual operation the tunnel remains operated. This inspection method not only ensures the safety of the operator but also reduces the maintenance time of the tunnel, which makes the utilization rate of the tunnel increase greatly.

6.3 Limitation and Future Work

However, apply the research into the practice is the following objects could improve in future research:

For the construction of tunnel by TBM, the LSTM solve the existing problem such as gradient explosion and vanish in RNN by introducing the gate system, however, the model is using existing collected data to train the model while different geological condition and different TBM model has a significant influence of TBM performance. Moreover, the current model could not assist the operator to decide during the TBM operating period. Therefore, in the future, one main task is using the currently collected data to instant train and validate the model. And another is to develop a function in the model that could be based on prediction results of data to guide and warn the extraordinary status of the operator during the excavation period.

For TBM lining defects detection, there are two kinds of neural network algorithm for crack detection of tunnels are used in this thesis, the first method is two stage Faster R-CNN which based on region proposal. Compare with the previous two stage method, the detection speed is improved by maximizing the parameter sharing the position sensitive fraction graph is proposed to solve the balance problem between object position invariance and position variability. However, the algorithm still takes a long time to process image cracks in the tunnel and detection speed is too slow to meet the requirement of real time crack detection. Therefore, it is considered to further optimize its network structure to improve its detection speed in order to satisfy the real time detection requirement.

The second method used in this thesis is the YOLO algorithm, which is an end-to-end detection algorithm based on the single-stage method. Therefore, compared with the two-stage algorithm, it has significant advantage in the detection speed. However, for smaller objects, the detection accuracy is relatively lower. The algorithm is optimizing by changing the specific parameters in the algorithm to achieve higher accuracy of crack detection. however, the amount of computation is also increased, which leads to the detection speed decreasing. Therefore, it is necessary to improve the detection accuracy as much as possible while satisfying the detection speed, and achieve a balance between the detection speed and the detection accuracy.

Compared with the conventional machine learning image classification, the algorithm based on CNN has higher accuracy and detection efficiency. Currently, it is only used for the detection of cracks. In the future, the algorithm can be optimized to evaluate the damage level of the detected tunnel crack defects (such as light crack, heavy crack, crack zone with seepage, etc.). Moreover, by adding different training sets, the model is except detecting other different defects such as peeling and seepage.

References

ADOKO, A. C., GOKCEOGLU, C. & YAGIZ, S. 2017. Bayesian prediction of TBM penetration rate in rock mass. *Engineering Geology,* 226**,** 245-256.

AFRADI, A., EBRAHIMABADI, A. & HALLAJIAN, T. 2019. Prediction of the Penetration Rate and Number of Consumed Disc Cutters of Tunnel Boring Machines (TBMs) Using Artificial Neural Network (ANN) and Support Vector Machine (SVM)—Case Study: Beheshtabad Water Conveyance Tunnel in Iran. *Asian Journal of Water, Environment and Pollution,* 16**,** 49-57.

ALVAREZ GRIMA, M. 2000. *Neuro-Fuzzy Modeling in Engineering Geology*.

ALVAREZ GRIMA, M., BRUINES, P. A. & VERHOEF, P. N. W. 2000. Modeling tunnel boring machine performance by neuro-fuzzy methods. *Tunnelling and Underground Space Technology,* 15**,** 259-269.

ARMAGHANI, D. J., MOHAMAD, E. T., NARAYANASAMY, M. S., NARITA, N. & YAGIZ, S. 2017. Development of hybrid intelligent models for predicting TBM penetration rate in hard rock condition. *Tunnelling and Underground Space Technology,* 63**,** 29-43.

ATTARD, L., DEBONO, C. J., VALENTINO, G. & DI CASTRO, M. 2018. Vision-based change detection for inspection of tunnel liners. *Automation in Construction,* 91**,** 142-154.

BARTON, N. 1999. TBM performance estimation in rock using Q(TBM). *Tunnels and Tunnelling International,* 31**,** 30-34.

BENARDOS, A. G. & KALIAMPAKOS, D. C. 2004. Modelling TBM performance with artificial neural networks. *Tunnelling and Underground Space Technology,* 19**,** 597-605.

BENATO, A. & ORESTE, P. 2015. Prediction of penetration per revolution in TBM tunneling as a function of intact rock and rock mass characteristics. *International Journal of Rock Mechanics and Mining Sciences,* 74**,** 119-127.

BENGIO, Y. & DELALLEAU, O. 2011. *On the Expressive Power of Deep Architectures*.

BRULAND, A. 2000. Hard Rock Tunnel Boring.

CHEN, Z., ZHANG, Y., LI, J., LI, X. & JING, L. 2020. Diagnosing tunnel collapse sections based on TBM tunneling big data and deep learning: A case study on the Yinsong Project, China. *Tunnelling and Underground Space Technology***,** 103700.

COLLOBERT, R. & WESTON, J. 2008. A unified architecture for natural language processing: deep neural networks with multitask learning. *Proceedings of the 25th international conference on Machine learning.* Helsinki, Finland: Association for Computing Machinery.

CORTES, C. & VAPNIK, V. 1995. Support-vector networks. *Machine Learning,* 20**,** 273-297.

CRCHI. Available: http://en.crchi.com// [Accessed].

CREG. Available: http://en.crectbm.com/ [Accessed].

DALAL, N. & TRIGGS, B. Histograms of oriented gradients for human detection. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 20-25 June 2005 2005. 886-893 vol. 1.

DANANGTODAY. 2017. Available: https://baodanang.vn/english/society/201710/investigation-into-cracks-in-existing-hai-van-tunnel-to-be-conducted-2575643/ [Accessed].

DAVOUDI, R., MILLER, G. R. & KUTZ, J. N. 2018. Data-driven vision-based inspection for reinforced concrete beams and slabs: Quantitative damage and load estimation. *Automation in Construction,* 96**,** 292-309.

DENG, Z., SUN, H., ZHOU, S., ZHAO, J., LEI, L. & ZOU, H. 2018. Multi-scale object detection in remote sensing imagery with convolutional neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing,* 145**,** 3-22.

EFTEKHARI, M., BAGHBANAN, A. & BAYATI, M. 2010. Predicting Penetration Rate Of A Tunnel Boring Machine Using Artificial Neural Network. *ISRM International Symposium - 6th Asian Rock Mechanics Symposium.* New Delhi, India: International Society for Rock Mechanics and Rock Engineering.

FARMER, I. W. & GLOSSOP, N. H. 1980. Mechanics of disc cutter penetration.

FATTAHI, H. & BABANOURI, N. 2017. Applying Optimized Support Vector Regression Models for Prediction of Tunnel Boring Machine Performance. *Geotechnical and Geological Engineering,* 35**,** 2205-2217.

FREUND, Y. & SCHAPIRE, R. A Short Introduction to Boosting. 1999.

GALL, V. E., MARWAN, A., SMARSLIK, M., OBEL, M., MARK, P. & MESCHKE, G. 2018. A holistic approach for the investigation of lining response to mechanized tunneling induced construction loadings. *Underground Space,* 3**,** 45-60.

GAO, B., WANG, R., LIN, C., GUO, X., LIU, B. & ZHANG, W. 2020. TBM penetration rate prediction based on the long short-term memory neural network. *Underground Space*.

GAO, X., SHI, M., SONG, X., ZHANG, C. & ZHANG, H. 2019. Recurrent neural networks for real-time prediction of TBM operating parameters. *Automation in Construction,* 98**,** 225-235.

GHASEMI, E., YAGIZ, S. & ATAEI, M. 2014. Predicting penetration rate of hard rock tunnel boring machine using fuzzy logic. *Bulletin of Engineering Geology and the Environment,* 73**,** 23-35.

GIRSHICK, R. B. 2015. Fast R-CNN. *2015 IEEE International Conference on Computer Vision (ICCV)***,** 1440-1448.

GIRSHICK, R. B., DONAHUE, J., DARRELL, T. & MALIK, J. 2014. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *2014 IEEE Conference on Computer Vision and Pattern Recognition***,** 580-587.

GITHUB. 2020. *GitHub Ranking* [Online]. Available: https://github.com/ [Accessed].

GONG, Q. M., JIAO, Y. Y. & ZHAO, J. 2006. Numerical modelling of the effects of joint spacing on rock fragmentation by TBM cutters. *Tunnelling and Underground Space Technology,* 21**,** 46-55.

GONG, Q. M. & ZHAO, J. 2007. Influence of rock brittleness on TBM penetration rate in Singapore granite. *Tunnelling and Underground Space Technology,* 22**,** 317-324.

GONG, Q. M. & ZHAO, J. 2009. Development of a rock mass characteristics model for TBM penetration rate prediction. *International Journal of Rock Mechanics and Mining Sciences,* 46**,** 8-18.

GOODFELLOW, I., BENGIO, Y. & COURVILLE, A. 2016. *Deep Learning*, The MIT Press.

GOODFELLOW, I. J., POUGET-ABADIE, J., MIRZA, M., XU, B., WARDE-FARLEY, D., OZAIR, S., COURVILLE, A. & BENGIO, Y. 2014. Generative adversarial nets. *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2.* Montreal, Canada: MIT Press.

GRAHAM, P. C. 1976. Rock exploration for machine manufacturers. In: Z.T. Bieniawski (Ed.). *Exploration for Rock Engineering***,** 173–180.

GRAVES, A. 2012. *Supervised Sequence Labelling with Recurrent Neural Networks*.

GRAVES, A., MOHAMED, A.-R. & HINTON, G. E. 2013. Speech recognition with deep recurrent neural networks. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing***,** 6645-6649.

GUO, T., XU, Z., YAO, X., CHEN, H., ABERER, K. & FUNAYA, K. 2016. *Robust Online Time Series Prediction with Recurrent Neural Networks*.

HAN, Y. & YE, J. C. 2017. Framing U-Net via Deep Convolutional Framelets: Application to Sparse-View CT. *IEEE Transactions on Medical Imaging,* 37.

HASSANPOUR, J., ROSTAMI, J. & ZHAO, J. 2011. A new hard rock TBM performance prediction model for project planning. *Tunnelling and Underground Space Technology,* 26**,** 595-603.

HE, K., ZHANG, X., REN, S. & SUN, J. 2015. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 37**,** 1904-1916.

HE, K., ZHANG, X., REN, S. & SUN, J. 2016. Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)***,** 770-778.

HERRENKNECHT. Available: https://www.herrenknecht.com/ [Accessed].

HINTON, G., DENG, L., YU, D., DAHL, G. E., MOHAMED, A., JAITLY, N., SENIOR, A., VANHOUCKE, V., NGUYEN, P., SAINATH, T. N. & KINGSBURY, B. 2012. Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups. *IEEE Signal Processing Magazine,* 29**,** 82-97.

HINTON, G. E., OSINDERO, S. & TEH, Y.-W. 2006. A fast learning algorithm for deep belief nets. *Neural Comput.,* 18**,** 1527–1554.

HOCHREITER, S. & SCHMIDHUBER, J. 1997. Long Short-Term Memory. *Neural Computation,* 9**,** 1735-1780.

HOSANG, J., BENENSON, R. & SCHIELE, B. 2017. Learning Non-maximum Suppression. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)***,** 6469-6477.

HUGHES, H. M. 1986. The relative cuttability of coal-measures stone. *Mining Science and Technology,* 3**,** 95-109.

ITA 2000. *Recommendations and Guidelines for Tunnel Boring Machines (TBMs)*.

JIANG, Y., ZHANG, X. & TANIGUCHI, T. 2019. Quantitative condition inspection and assessment of tunnel lining. *Automation in Construction,* 102**,** 258-269.

KATHURIA, A. 2018. What's new in YOLO v3?

KHADEMI HAMIDI, J., SHAHRIAR, K., REZAI, B. & ROSTAMI, J. 2010. Performance prediction of hard rock TBM using Rock Mass Rating (RMR) system. *Tunnelling and Underground Space Technology,* 25**,** 333-345.

KOOPIALIPOOR, M., TOOTOONCHI, H., JAHED ARMAGHANI, D., TONNIZAM MOHAMAD, E. & HEDAYAT, A. 2019. Application of deep neural networks in predicting the penetration rate of tunnel boring machines. *Bulletin of Engineering Geology and the Environment,* 78**,** 6347-6360.

KRIZHEVSKY, A., SUTSKEVER, I. & HINTON, G. E. 2012. ImageNet classification with deep convolutional neural networks. *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1.* Lake Tahoe, Nevada: Curran Associates Inc.

LAUGHTON, C. 1998. *Evaluation and prediction of tunnel boring machine performance in variable rock masses.*

LECUN, Y., BENGIO, Y. & HINTON, G. 2015. Deep learning. *Nature,* 521**,** 436-444.

LECUN, Y., BOTTOU, L., BENGIO, Y. & HAFFNER, P. 1998. Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE,* 86**,** 2278-2324.

LEE, J. H., LEE, J., PARK, J. W. & MOON, Y. S. Efficient Algorithms for Automatic Detection of Cracks on a Concrete Bridge. 2008.

LI, S., LIU, B., XU, X., NIE, L., LIU, Z., SONG, J., SUN, H., CHEN, L. & FAN, K. 2017. An overview of ahead geological prospecting in tunneling. *Tunnelling and Underground Space Technology,* 63**,** 69-94.

LI, Y.-H., XU, S.-D. & LIU, J.-P. 2015. A new convergence monitoring system for tunnel or drift based on draw-wire displacement sensors. *Tunnelling and Underground Space Technology,* 49**,** 92-97.

LIN, T.-Y., DOLLáR, P., GIRSHICK, R. B., HE, K., HARIHARAN, B. & BELONGIE, S. J. 2017. Feature Pyramid Networks for Object Detection. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)***,** 936-944.

LIPTON, Z. C. 2015. A Critical Review of Recurrent Neural Networks for Sequence Learning. *ArXiv,* abs/1506.00019.

LIU, Q., LIU, J., PAN, Y., KONG, X. & HONG, K. 2017. A case study of TBM performance prediction using a Chinese rock mass classification system – Hydropower Classification (HC) method. *Tunnelling and Underground Space Technology,* 65**,** 140-154.

LIU, W., ANGUELOV, D., ERHAN, D., SZEGEDY, C., REED, S., FU, C.-Y. & BERG, A. SSD: Single Shot MultiBox Detector. ECCV, 2016.

LOWE, D. G. 1999. Object Recognition from Local Scale-Invariant Features. *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2.* IEEE Computer Society.

MAGGIORI, E., TARABALKA, Y., CHARPIAT, G. & ALLIEZ, P. 2017. Convolutional Neural Networks for Large-Scale Remote-Sensing Image Classification. *IEEE Transactions on Geoscience and Remote Sensing,* 55**,** 645-657.

MAHDEVARI, S., SHAHRIAR, K., YAGIZ, S. & AKBARPOUR SHIRAZI, M. 2014. A support vector regression model for predicting tunnel boring machine penetration rates. *International Journal of Rock Mechanics and Mining Sciences,* 72**,** 214-229.

MATSUOKA, T. & MATSUSHIMA, K. 2016. Crack detection using spectral clustering based on crack features. *2016 IEEE Region 10 Conference (TENCON)***,** 2575-2578.

MINSKY, M. & PAPERT, S. 1969. *Perceptrons; an introduction to computational geometry,* Cambridge, Mass.,, MIT Press.

MOHAMMADI, S. D., TORABI-KAVEH, M. & BAYATI, M. 2015. Prediction of TBM penetration rate using intact and mass rock properties (case study: Zagros long tunnel, Iran). *Arabian Journal of Geosciences,* 8**,** 3893-3904.

MUDULI, P. R. & PATI, U. C. 2013. A novel technique for wall crack detection using image fusion. *2013 International Conference on Computer Communication and Informatics***,** 1-6.

NELSON, P. 1983. *Tunnel boring machine performance in sedimentary rock.*

OLIVEIRA, H. & CORREIA, P. L. 2013. Automatic Road Crack Detection and Characterization. *IEEE Transactions on Intelligent Transportation Systems,* 14**,** 155-168.

OULLETTE, R., BROWNE, M. & HIRASAWA, K. 2004. *Genetic algorithm optimization of a convolutional neural network for autonomous crack detection*.

PAPAGEORGIOU, C., OREN, M. & POGGIO, T. 1998. A general framework for object detection. *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)***,** 555-562.

QUINLAN, J. R. 1986. Induction of decision trees. *Machine Learning,* 1**,** 81-106.

RAILPROJECTVICTORIA. Available: https://railprojects.vic.gov.au/ [Accessed].

RAVINDRAN, N., SHERYL, O. A., SAMRAJ, A. & MAHESWARI, N. Stable and crit gesticulation recognition in children and pregnant women by Naïve Bayes classification. 2013 International Conference on Current Trends in Information Technology (CTIT), 11-12 Dec. 2013 2013. 259-264.

REDMON, J., DIVVALA, S., GIRSHICK, R. B. & FARHADI, A. 2016. You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)***,** 779-788.

REDMON, J. & FARHADI, A. 2017. YOLO9000: Better, Faster, Stronger. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)***,** 6517-6525.

REDMON, J. & FARHADI, A. 2018. YOLOv3: An Incremental Improvement. *ArXiv,* abs/1804.02767.

REN, S., HE, K., GIRSHICK, R. & SUN, J. 2017. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 39**,** 1137-1149.

ROBBINS.

RONNEBERGER, O., FISCHER, P. & BROX, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. MICCAI, 2015.

ROSENBLATT, F. 1958. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review,* 65**,** 386-408.

ROSTAMI, J. 1997. *DEVELOPMENT OF A FORCE ESTIMATION MODEL FOR ROCK FRAGMENTATION WITH DISC CUTTERS THROUGH THEORETICAL MODELING AND PHYSICAL MEASUREMENT OF CRUSHED ZONE PRESSURE.*

ROSTAMI, J. 2016. Performance prediction of hard rock Tunnel Boring Machines (TBMs) in difficult ground. *Tunnelling and Underground Space Technology,* 57**,** 173-182.

ROSTAMI, J. & OZDEMIR, L. 1993. New model for performance production of hard rock TBMs. *Proceedings - Rapid Excavation and Tunneling Conference***,** 793-809.

SANG, J., WU, Z., GUO, P., HU, H., XIANG, H., ZHANG, Q. & CAI, B. 2018. An improved YOLOv2 for vehicle detection. *Sensors,* 18**,** 4272.

SAPIGNI, M., BERTI, M., BETHAZ, E., BUSILLO, A. & CARDONE, G. 2002. TBM performance estimation using rock mass classifications. *International Journal of Rock Mechanics and Mining Sciences*, 39**,** 771-788.

SCHMIDHUBER, J. 1992. Learning to control fast-weight memories: An alternative to dynamic recurrent networks. *Neural Computation,* 4**,** 131-139.

SERMANET, P., EIGEN, D., ZHANG, X., MATHIEU, M., FERGUS, R. & LECUN, Y. 2014. OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. *CoRR,* abs/1312.6229.

SHAO, C., LI, X. & SU, H. 2013. Performance Prediction of Hard Rock TBM Based on Extreme Learning Machine. *Proceedings of the 6th International Conference on Intelligent Robotics and Applications - Volume 8103.* Busan, South Korea: Springer-Verlag.

SUNDERMEYER, M., SCHLüTER, R. & NEY, H. LSTM Neural Networks for Language Modeling. INTERSPEECH, 2012.

TAO, H., JING-CHENG, W. & LANG-WEN, Z. 2015. Prediction of hard rock TBM penetration rate using random forests. *The 27th Chinese Control and Decision Conference (2015 CCDC)***,** 3716-3720.

TIN KAM, H. Random decision forests. Proceedings of 3rd International Conference on Document Analysis and Recognition, 14-16 Aug. 1995 1995. 278-282 vol.1.

UIJLINGS, J., SANDE, K., GEVERS, T. & SMEULDERS, A. 2013. Selective Search for Object Recognition. *International Journal of Computer Vision,* 104**,** 154-171.

WANG, L., KANG, Y., CAI, Z., ZHANG, Q., ZHAO, Y., ZHAO, H. & SU, P. 2012. The energy method to predict disc cutter wear extent for hard rock TBMs. *Tunnelling and Underground Space Technology,* 28**,** 183-191.

WEI, M., WANG, Z., WANG, X., PENG, J. & SONG, Y. 2020. Prediction of TBM penetration rate based on Monte Carlo-BP neural network. *Neural Computing and Applications*.

YAGIZ, S. & KARAHAN, H. 2011. Prediction of hard rock TBM penetration rate using particle swarm optimization. *International Journal of Rock Mechanics and Mining Sciences,* 48**,** 427-433.

YAGIZ, S. & KARAHAN, H. 2015. Application of various optimization techniques and comparison of their performances for predicting TBM penetration rate in rock mass. *International Journal of Rock Mechanics and Mining Sciences,* 80**,** 308-315.

YIN, W., KANN, K., YU, M. & SCHüTZE, H. 2017. Comparative Study of CNN and RNN for Natural Language Processing. *ArXiv,* abs/1702.01923.

ZHANG, Q., LIU, Z. & TAN, J. 2019. Prediction of geological conditions for a tunnel boring machine using big operational data. *Automation in Construction,* 100**,** 73-83.

ZHAO, J. 2006. The challenge of TBM excavation in rock. *Tunnels & tunnelling international.***,** 27-29.

ZHAO, J., GONG, Q. M. & EISENSTEN, Z. 2007a. Tunnelling through a frequently changing and mixed ground: A case history in Singapore. *Tunnelling and Underground Space Technology,* 22**,** 388-400.

ZHAO, Z., GONG, Q. M., ZHANG, Y. & ZHAO, J. 2007b. Prediction model of tunnel boring machine performance by ensemble neural networks. *Geomechanics and Geoengineering,* 2.

ZHOU, L., ZHU, Z., DONG, Y., YING, P. & WANG, M. 2019. Study of the fracture behavior of mode I and mixed mode I/II cracks in tunnel under impact loads. *Tunnelling and Underground Space Technology,* 84**,** 11-21.

ZITNICK, C. L. & DOLLáR, P. Edge Boxes: Locating Object Proposals from Edges. *In:* FLEET, D., PAJDLA, T., SCHIELE, B. & TUYTELAARS, T., eds. Computer Vision – ECCV 2014, 2014// 2014 Cham. Springer International Publishing, 391-405.