



CSE 499

One Semester Long Single Project

A Defect Detection Method for Rail Fasteners and Fishplates Based on Deep Learning

AUTHOR

Md. Mehedi Hasan

ID: 181014077

SUPERVISOR

Muhammad Abul Hasan, PhD

Assistant Professor
Department of CSE, ULAB

A project report submitted in partial fulfillment of the requirements for the degree of Bachelor of Science
in Computer Science and Engineering

Department of Computer Science and Engineering
University of Liberal Arts Bangladesh

688, Beribadh Road, Mohammadpur, Dhaka – 1207, Bangladesh

Certificate of Approval

The project report entitled “A Defect Detection Method for Rail Fasteners and Fishplates Based on Deep Learning” is submitted to the Department of Computer Science and Engineering, University of Liberal Arts Bangladesh, Dhaka in partial fulfillment of the requirements for the degree of the Bachelor of Science.

Dated: 30th June 2022

Abul Kalam Al Azad, PhD

Associate Professor and Acting Head

Department of Computer Science and Engineering

University of Liberal Arts Bangladesh

Signature & Date

Muhammad Abul Hasan, PhD

Assistant Professor

Department of Computer Science and Engineering

University of Liberal Arts Bangladesh

Signature & Date

DEDICATION

This thesis project is dedicated to my parents who had faith, trust, love, and unconditional support throughout my life. A special feeling of gratitude to my supervisor who helped and guided me to complete this work and whose words of encouragement and push for tenacity ring in my ears.

I'm grateful to all of my teachers, friends, and relatives for their patience, support, love, and perseverance during the ups and downs to accomplish this project.

Also, I would like to dedicate this work to the sake of Almighty Allah, my Creator, and my Master.

Acknowledgment

I would like to convey my heartfelt gratitude to Dr. Muhammad Abul Hasan, my project supervisor for allowing me to collaborate with him on this thesis project. I owe him a great debt of gratitude for his direction and regular supervision, as well as for giving vital information.

The Department of Computer Science and Engineering is also a significant part since it has given me the chance to enhance my talents and grow as a learner. The essential assistance I've received from the department's whole professors and staff has had a significant positive impact on my undergraduate studies.

I would want to express my gratitude to my family members for their support and encouragement during this whole project, as well as during the last four years of my undergraduate studies. I would not have been able to complete this project without my parents' love and care, and I am grateful to them for that.

Md. Mehedi Hasan

University of Liberal Arts Bangladesh

Date: 28th June 2022

ABSTRACT

Among the components of railway tracks, Fasteners and Fishplates are more censorious and vulnerable components that need a regular inspection. Fasteners are used to impose the rail onto a fixed position of crosstie and Fishplates join two different sections of rail. Like other components of railway track Fasteners and Fishplates inevitably produce defects as a result of long-term train operations and direct revelation to the natural surroundings which have a crucial effect on the safety of train operation. Several missing fasteners and defective fishplates cause gage widening, wheel climb, and train derailments consequently. In this work, a multi-object detection method based on a deep convolutional neural network that can perform automated detection of missing fasteners and defective fishplates alongside non-defective ones is proposed. Among state of art object detection algorithms, several YOLOv5 architectures are trained on a dataset consisting of images collected from the Bangladesh Railway. Experimental tests are conducted on the ballast railway track images and videos collected from the field to demonstrate the effectiveness and robustness of the method. Experimental results from multiple contemporary state of art object detection algorithms show that YOLOv5 not only detects fasteners and fishplates defects precisely but also outperforms others through several evaluation metrics in all stages.

Keywords: Railway Fasteners and Fishplates, Defection Detection, Automatic Inspection, Deep Learning, Convolutional Neural Network, YOLOv5.

Table of Contents

1	Introduction	7-12
1.1	Introduction	7
1.2	Motivation	10
1.3	Project Timeline	11
1.4	Report Layout	12
2	Background Study	13-17
2.1	Related Works	13
2.2	Comparative Analysis & Summary	15
2.3	Scope	15
2.4	Challenges	16
3	Research Methodology	18-27
3.1	Data Collection	18
3.2	Data Annotation and Labeling	19
3.3	Preprocessing and Data Augmentation and Split	22
3.4	Applied Mechanism	23
3.5	Evaluation Metrics	27
4	Experimental Results & Discussion	29-42
4.1	Training and Validation Result	29
4.2	Comparative Analysis	38
4.3	Inference with trained models	41
5	Conclusion and Implications for Future Research	43-44
5.1	Summary of the Study & Conclusion	43
5.2	Implication for Further Study	44
	References	45-48

Chapter 1: Introduction

1.1 Introduction

Compared to other transportation services, Railway transport is a more convenient, safe, and cost-efficient mode of transportation that contributes substantially to the industrial and socio-economic development of a country. As the railway network is expanding around the world rapidly, the safe and uninterrupted operation of the railway has become a significant objective with the extensive public interest.

The pair of rails is the principal component of a railway track which bears the load of the wheelset of the train and directs the wheels forward, as shown in the figure-1.

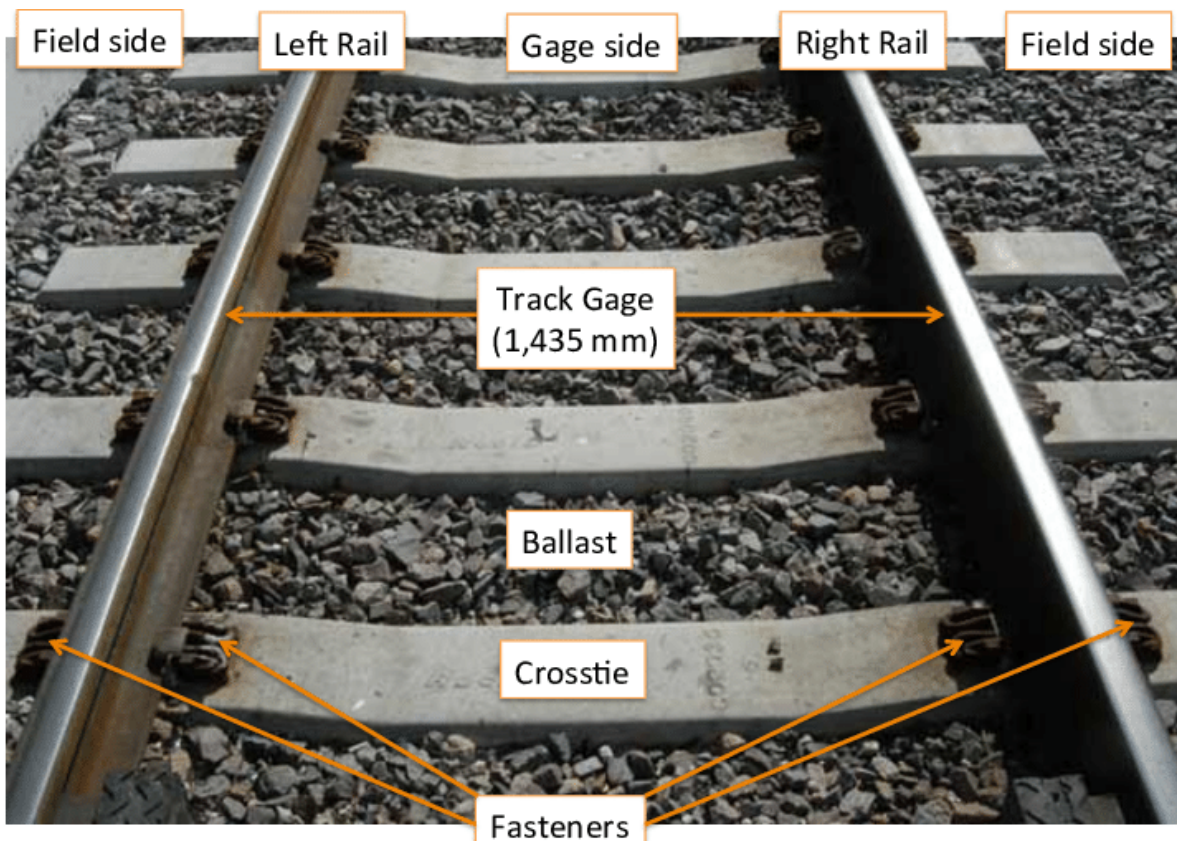


Fig.1: Rail Track components

This pair of rails is imposed onto a fixed position of the crossties using fasteners. The principal functionalities of fasteners are sealing the rails onto slab or sleepers [1], reducing friction time vibration

[2], and strictly conserving the track gauge [3]. Generally, four fasteners are utilized to fasten the pair of rails with a crosstie and that's why fasteners are the most numerous and imperative components of railway tracks compared to the usage of other components. As per the standard [4], there are 5700-6700 fasteners in a 1-kilometer railway track and this abundance makes it very tough to inspect.

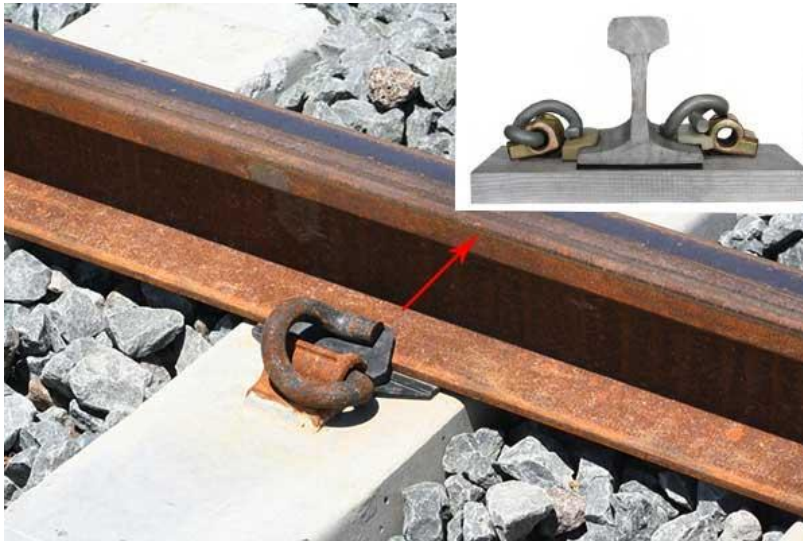


Fig. 2: Structure and Location of Fasteners

Rail Fishplates known as “Rail Joint”, is a metal plate that normally connects two adjacent rail sections using several rail bolts or spikes. This flat plate carries over the forces from one section of the rail to the adjacent one.



Fig. 3: Structure and Operating Location of Fishplates

The physical state of fasteners and fishplates in the service are damaged by the continuous contact frictions of rail-wheel, extrusion, elastic deformation, material aging, and direct natural subjection. Correspondingly, fractured or loose fasteners, missing fasteners, and missing one more bolt in fishplates are very common phenomena for railway tracks, especially which are intended for high-speed operation.



Fig. 3: Rail Track with Missing Fasteners

As this defective nature of railway components threatens the safety and smooth operation of trains, it is very crucial to inspect the railway regularly. To conduct inspections successfully regularly it is also important to study the defect detection methods based on modern computer vision technology.

Computer-Vision technologies, which normally extract perceptible features from input images of railway tracks and detect multiple types of defects in real time have been progressively popular due to efficiency and ease of accessibility.

1.2 Motivation

The railway network of the whole world is expanding over the years. The government of Bangladesh not only intended to add all the districts of the country to the national railway network but also to join our country in the international railway network such as Trans-Asian Railway. For this reason, spreading rail networks will require more advanced technology like computer vision for efficient and regular maintenance.

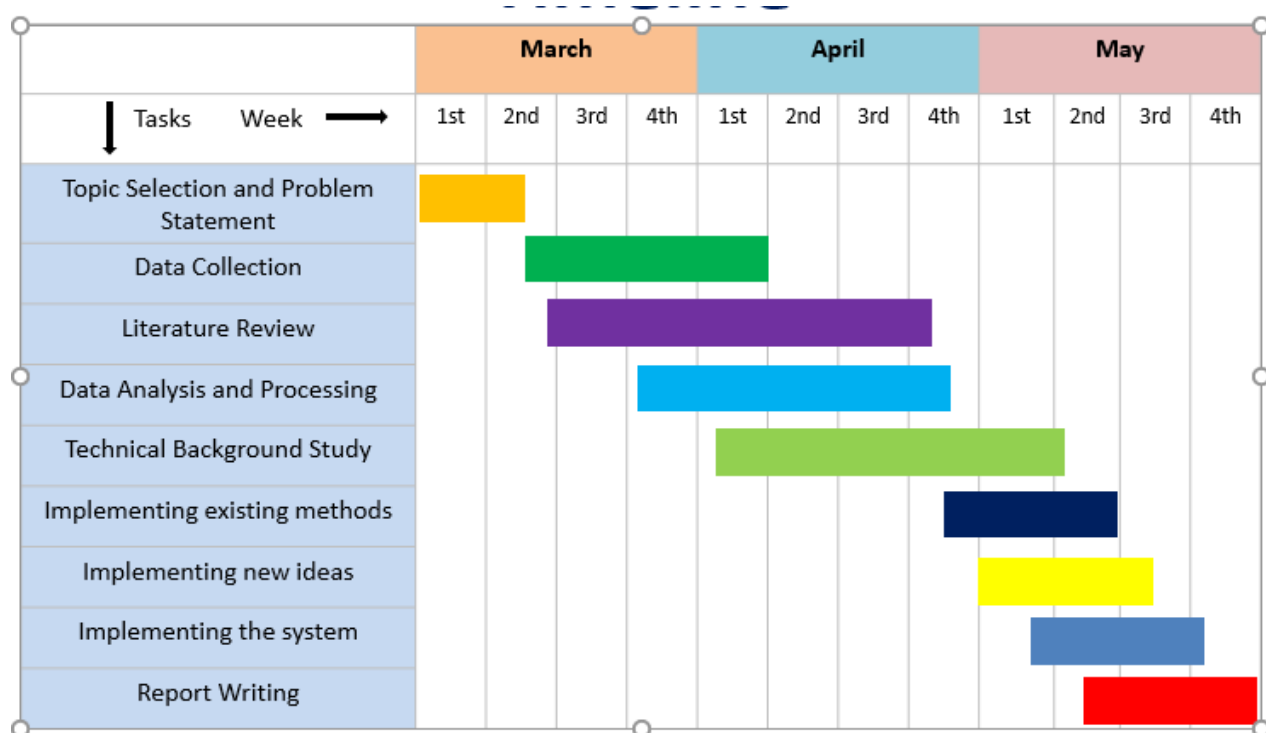
Even though railway tracks and their components need to be examined periodically to prevent incidents of train derailments and other massive accidents, the current method of inspection is not sustainable for regular inspection.

Currently, railway tracks are inspected mostly by manual inspection systems, conducted by trained and professional human operators as shown in figure 4. These railroad personnel either walk on the tracks or sit on hi-rail vehicles and move at a very slow speed observing the track components with human eyes. This traditional inspection system suffers from various disadvantages like a high rate of missed detection, moderate detection efficiency, inferior real-time performance, time consumption and requires skillful operators. Apart from these, this inspection system is subjective and doesn't produce any visual record which could be audited later. All these problems can be solved if we can deploy computer-vision technology with deep convolutional neural networks to detect defects of various rail track components.



Fig.4: Manual Inspection with Human Operators

1.3 Project Timeline



1.4 Report Layout

Chapter 2: In chapter 2, I will provide my literature review and project background. I will explain why I selected this thesis project and why I want to research on this problem. The scope this thesis project will be also described here.

Chapter 3: In this chapter, I will introduce the methodology and steps that I have followed to research this problem. I will provide concise theoretical description about the methods or models with which I've experimented with the dataset.

Chapter 4: In this chapter, I will show and discuss various experimental outcomes with the help of diagrams. I will also introduce a comparative analysis of the performances of the models that I've experimented with.

Chapter 5: In the last chapter I will discuss the outcomes of this entire thesis project and conclude the observations and findings. I will also discuss the future study.

Chapter 2: Background Study

2.1 Related Works

In traditional computer technology, the defect detection procedure for railway components utilizes shallow learning classification algorithms to classify the condition of railway components like fasteners, rail surfaces, etc. AdaBoost Classifier [8, 9], Probabilistic Graphical Models (PGM)[10], Support Vector Machine (SVM)[11, 12, 13], multilayer perceptron classifier[14, 15], etc. are the classification algorithms that are used by authors in their works for defect detection.

In this procedure, features are extracted from the components in the image using artificially outlined features and inputted into the classification model. Authors used different types of shallow features in their research works, such as edge feature[16], Haar-like feature[17, 18], direction field feature[19], HOG feature[20], Dense-SIFT feature[21] etc.

In their work [22] Xavier, Vishal, and Rama bootstrapped difficult samples and carefully aligned training data, also reducing intra-class variation to improve the classification accuracy. They used HOG as a feature extractor and SVM as the classifier model for detecting railway fasteners.

Nevertheless, detection methods like this suffer from low accuracy as features are extracted for object area instead of object detection. Thus, extracted object features are vulnerable to the impact of unnecessary background information and classifiers can't identify objects like fasteners in unusual states precisely.

In the last decade, due to the advancement of deep learning technology, success in the image processing field, and huge increment in the speed and availability of GPUs, many impactful works are done by scholars applying deep neural networks for the detection of multiple rail track components.

Li et al. [23] proposed a method consisting of three steps to classify the state of fasteners. Firstly, the fastener area is localized using the saliency model and then proposed sub-images of fasteners are segmented semantically using PSPNet. Ultimately, the fastener's state is classified using vector geometry measurements.

ZIWEN, MANGUI, WANG [24] used a deep extractor adding conditional random fields with a convolutional neural network to extract low-level abstract features and preceding information on the rail surface. They proposed a convolutional neural network consisting of two branches to generate feature maps of high levels. They also introduced CRF-RNN by formulating dense conditional random fields with pairwise Gaussian potentials to achieve high accuracy in the inspection.

Ma et al. [25] improved the accuracy of classification using the CNN by cropping out the sub-images of the tie area which were not important for the identification of the fastener state from the image of the fastener area.

Praneeth, Johnny, and Florian [26] adopted various steps of image processing to reduce the fastener localization error before using ResNet-50 and a custom convolution neural network to detect rail track fasteners.

Gibert et al. [27] proposed a method with a combination of customized convolutional network and SVM to generate the abstract features from fasteners firstly, and then classify fastener types accordingly.

Dechen, Qiang, and Jianwei[28] proposed a method consisting of ResNet and Adversarial Network (GAN) to diagnose fastener faults for automated inspection. Firstly, they tracked the distribution of missing fastener data using the GAN generating additional real faulty fastener samples. ResNet is utilized then to detect and recognize the belonging class.

Tiange, Fangfang and Tsui [29] showed that the enhanced YOLOv2 model performed better than multiple one-stage contemporary object detectors to detect rail track components like fasteners, bolts, rails, etc.

They proposed this method as a fast and accurate method for the detection of rail track components.

Ya-Wen, Chen-Chiung, and Wei-Hsin[30] used YOLOv3 to detect the defects of railway fasteners in real time. They collected data containing fastener images from a 20 km railway track. They proposed this method as an efficient way of railway track inspection in real time.

Model sizes and the number of parameters are reduced to increase detection speed by Tangbo Jialin, and Jianwei [31] in their work. They proposed an optimized version of YOLOv4 to detect rail surface defects by placing MobileNetv3 as the backbone of YOLOv4. They also applied separable convolution on the PANet layer. In the result, they showed that they could reduce the number of parameters to 78.04% and achieve detection speed up to 10.36 fps.

Multiple CNN models are utilized by Danyang, Liming, and Shubin Zheng[32] to detect the defects of fasteners and surfaces of railway tracks. They proposed an optimized backbone for the YOLOv5 model by using a Ghost bottleneck [33] replacing the original CSP bottleneck in the architecture which essentially reduces the model parameters and increases the processing speed of the network. Fasteners and surface of rail are localized by this modified version YOLOv5. In the next step, they used Mask R-CNN to detect and segment the surface defects and ResNet to classify the fastener's state separately.

2.2 Comparative Analysis and Summary

In the aforementioned related works and as far as I've explored on the web, there are some similar research works that have been conducted before. For instance, many authors have proposed methods based on deep learning and one-stage object detectors to detect rail track components like fasteners. However, I didn't find any conducted research that worked with railway track fishplate defect detection with the help of deep learning technology. Defect detection of both rail track fasteners and fishplates has not appeared in previous works. If we see the region of areas from where data are collected, it is obvious that most of the datasets are created from Chinese railways. No dataset is generated from Bangladesh railway based on what detection method is proposed. As we know that variety, abstractness, noises, etc. differ based on the region from which the dataset is generated. The physical shape defect's nature of operational fasteners and fishplates in a region is different from the other regions, especially in other countries. For this reason, the detection problem of Bangladesh railway track components like fasteners and fishplates is a new problem to solve.

Apart from the dataset, no researcher experimented with different versions of modern state of art object detection models. For example, there are different versions of YOLOv5 that differ in size, accuracy, number of parameters, detection speed, etc. As the inspection system is conducted in real time with a vehicle of minimum velocity, the detection speed of the proposed method is a concern.

2.3 Scope of the problem

The problem of Defection detection of railway track components like fasteners and fishplates has a wide scope currently and will be extended in the future. Railway networks around the globe are expanding every year and many neighboring countries are being connected through rail paths as this transportation medium is more safe, comfortable, and cheap. For example, the Bangladesh government has already taken the initiative to join the country in Trans-Asian Railway and spread the railway network in all 64 districts. When the railway network is much expanded, it requires huge caliber and resources for maintenance purposes in a regular manner. In this case, only the application of advanced computer vision technology can help with high efficiency.



Fig. 5: Automated Inspection

Although the railway network is not expanded, automated inspection with the help of advanced computer vision technology can ensure smooth and safe operation of trains, reduce wastage of resources, and reduce the number of incidents of train derailments, and so on. Many first-world countries have already implemented automatic inspection systems based on computer vision and ensured flawless operation of railways. Therefore, it is very obvious that the extent of this defection detection problem is huge.

2.4 Challenges

While working with image data in machine learning or deep learning algorithms, is quite challenging because of the data size and the resources that it requires for processing. Apart from the common resources like CPU and RAM, deep learning algorithms require GPU or TPU for huge mathematical calculations in the background. There is a huge advancement in the GPU production industry in recent years, still, GPUs with basic capacity are not enough for deep learning model training. GPUs with higher capacity are very expensive and also not available everywhere.

Google Colab, Caggle, Amazon, etc. provide free virtual runtimes with GPUs or TPUs, still, there are many issues such as they provide a limited amount of runtime with limited resources, the requirement of stable internet, need to be active in the foreground while training, etc. Paid versions of these virtual runtimes also come with many issues like unavailability in every country, resource limitations, etc. Without GPU or TPU it is not possible to conduct good research based on deep learning with image data. The resources or information needed to conduct a technical background study are not available on the internet in a learner-friendly manner. Troubleshooting any problem while experimenting requires much time and effort for this reason.

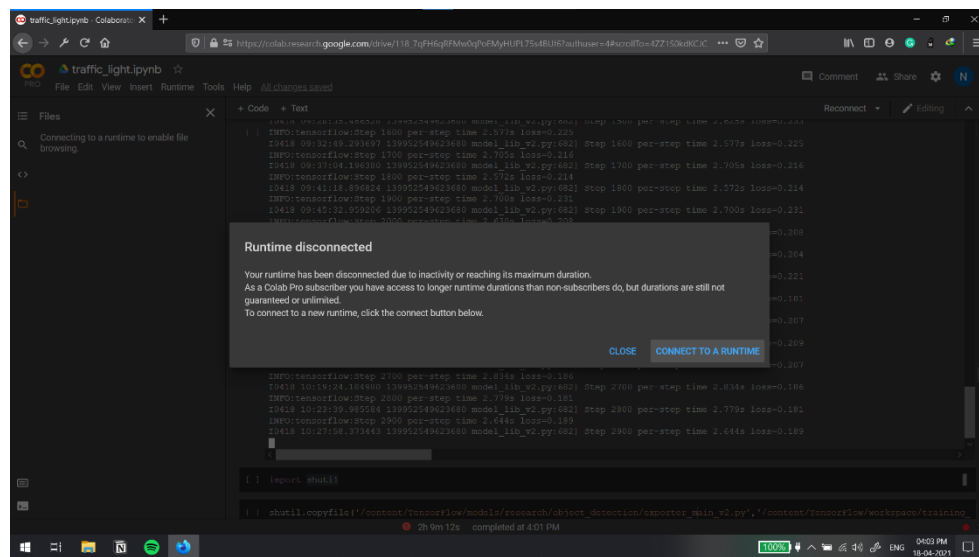


Fig. 6: Google Colab Runtime Limitation

To teach a model of deep learning about a defect requires a good number of images with similar defects. However, in the data collecting field, that number of images may not be available and it is very tough to

collect the minimum number of images with defects. Huge differences are created in the number of intra-class instances as a consequence which is a barrier to achieving good accuracy for all classes. As the defect detection problem of a public transportation system has a safety concern, the data annotation process requires personnel who has professional knowledge about rail track components and their physical structure.

Chapter 3: Research Methodology

3.1 Data Collection

The Akhaura-Sylhet section of Bangladesh Railway was chosen for collecting image and video data. On this 177 kilo-meter single line, meter gauge track 8 passenger trains with (double trip) and multiple freight trains are operated daily. Incidents of train derailments in this section, in greater numbers than in other sections and reports [5] by Bangladeshi media indicated that there could be number of defects of track components in this section.

RGB images were collected from different spots of the section using smartphone cameras with a resolution of 4000 x 3000 pixels. Images were captured from two different angles using gimbals. One angle emulates image capturing from the locomotive front-upper side which captures the frame containing both rail, crossties, and fasteners (upper view). Another angle emulates image capturing from a near-side position of rail which captures the frame containing the side surface of rail, fishplates, and fasteners (where available). Emulation of the image capturing process from two different angles is shown below:



Fig.7: Image Capturing Process from Side Angle

The total number of collected images is 2826 including images from both angles and the total number of videos collected is 15 of length 55 minutes in total.

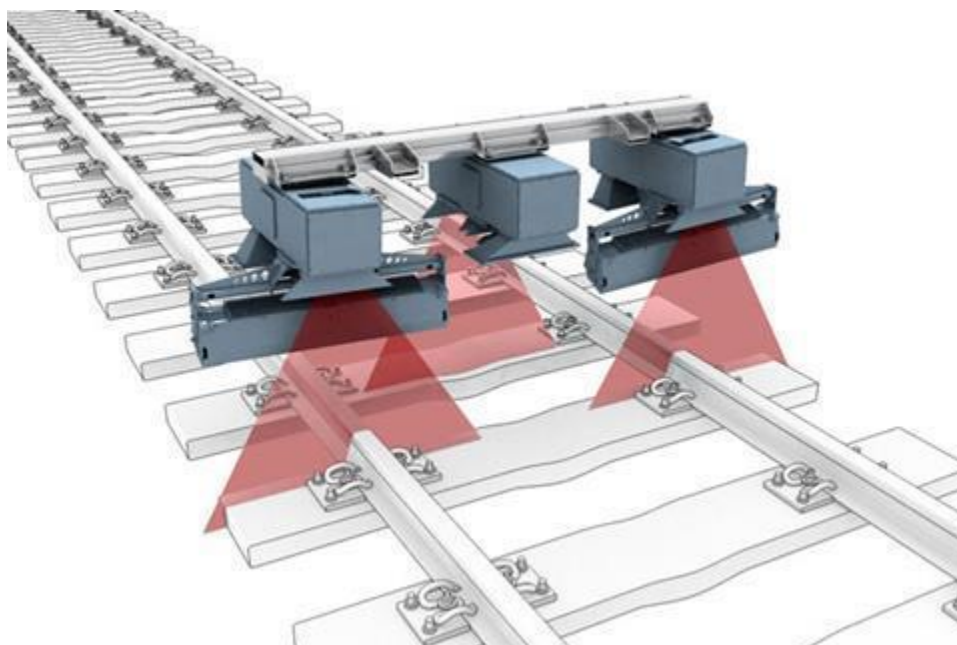


Fig.8: Image Capturing Process from Upper Angle

3.2 Data Annotation and Labeling

After eliminating several images, 2445 images were selected for annotation. As this research involved detecting defects of fasteners and fishplates, four distinct classes/labels/targets were created. The four classes are shown below with images.

1. Fastener: It's the class of all normal/fault-less rail track fasteners. A fastener image could be from any angle. This class has the most number of instances in the dataset. In an image, there are normally multiple instances of this class.

2. Missing Fastener: It's the class of all instances where the clipper is missing. This class represents the defective state of a fastener. This class has the second highest number of instances in the dataset. In an image, there are normally multiple instances of this class.



Fig.9 &10: Fasteners and Missing Fasteners

3. Non-Defective Fishplates: All the instances of fishplate where all the volts (4 volts in general) are available are labeled using this class. In the track, the number of fishplates is very fewer than fasteners and that's why the number of instances of this class is not so high. In an image, there is normally a single fishplate instance at most.



Fig. 11: Non-defective Fishplate

4. Defective Fishplate: All the instances of fishplate where at least a single bolt is missing are labeled using this class. The class represents the defective state of a fishplate. In the track, the number of defective fishplates is very fewer than other types of instances and that's why the number of instances of this class is the lowest in the dataset. In an image, there is normally a single defective fishplate instance at most.



Fig.12 Defective Fishplate

Data labeling is done by using “LabelImg” software. Suppose an image is marked with a bounding box of an instance in the front end. In that case, a label file is generated in the background with the same name of the image except for an extension that contains the label, coordinates of the instance's center point, height, and width sequentially. All these values are normalized using the resolution of the image. These label files are very important for training as they contain the learning values of a model.

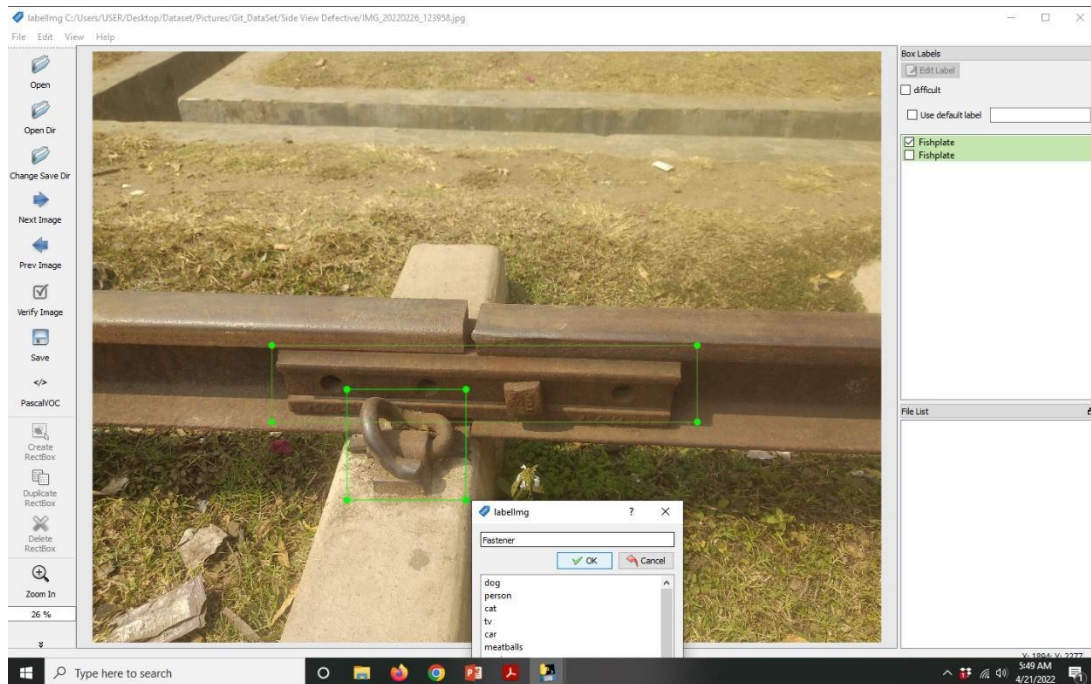


Fig. 13: Labeling Using LabelImg Tool

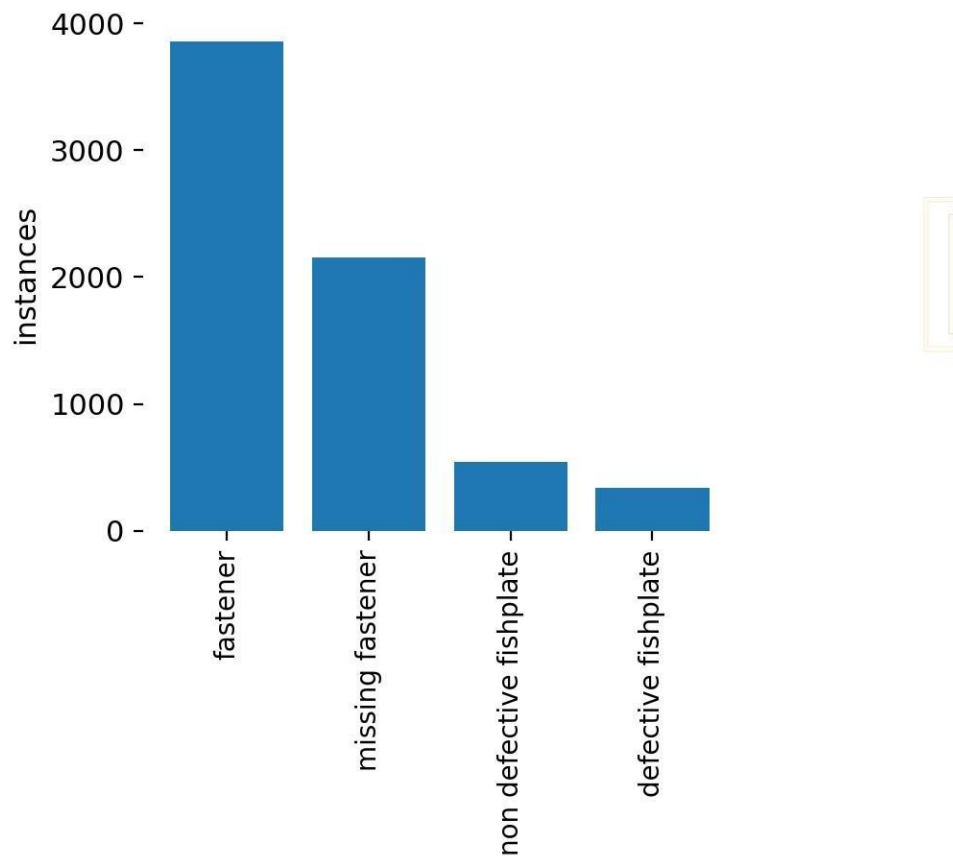


Fig 35: Number of Instances of different classes

3.3 Preprocessing and Data Augmentation and Split

All the images are resized to 640 x 640 pixels using image processing software. Image resizing is necessary because all the collected data may not be in the same resolution and manageable resources may not be able to take the load of a dataset containing images of very high resolution while training. It's stated that 640 x 640 resolution is good enough for the learning process of a model and detection with good accuracy.

After preprocessing, the data augmentation process is done using “roboflow” following these steps:

1. Flip: Vertical and Horizontal

2. Rotate 90 degrees: Counterclockwise and Upside Down
3. Rotation: Between -10 degrees and +10 degree
4. Saturation: Between -40% and +40%
5. Brightness: Between -25% and +25%
6. Exposure: Between -10% and +10%
7. Blur: Up to 1.25 pixel
8. Noise: Up to 5% pixel

These steps are conducted so that the model can perform well with unseen data in the real environment.

After the augmentation process, the total number of images is increased to 4890. For the dataset generation, 80% of data were kept as training data, 18% as validation, and 2% as testing data.

3.4 Applied Mechanism

In this thesis project, the YOLOv5 model family is used as a single-stage object detector to localize and classify the fasteners and fishplates in the original image. The overview of the network framework is shown below in Figure:

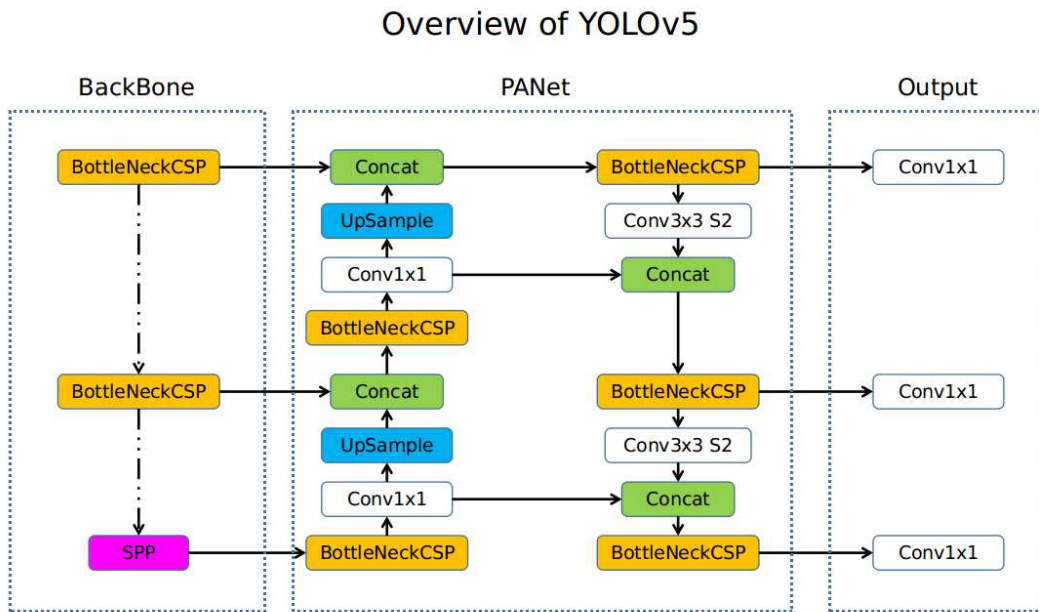


Fig. 14: YOLOv5 Overview

YOLO (You Only Look Once) is the name of a series of deep convolutional neural networks which perform both object localization and classification in the image within a single stage. Like the other contemporary single-stage object detector, it consists of three important parts which perform distinct functionalities. These are considered as 1. Backbone 2. Neck 3. Head.

The main purpose of the model backbone is to extract important object features from the input image. It is a Convolutional Neural Network. Cross Stage Partial Network (CSP) [6] is introduced as the backbone of YOLOv5 intended to extract rich informative and fine-grained features from input images. A Focus component is added to the Backbone to perform input image slicing. This Focus component also downsample information by including floating point operation (FLOPs) from the image and retaining a more complete image for feature extraction in the next block.

The main role of Model Neck is to produce feature pyramids. Feature pyramids assist models to perform better logical reasoning on the task of object scaling.

When an object with a different scale and size appears in the input image it helps the model to recognize the object according to the previous learnings. Thus, Feature pyramids provide advantages to performing accurately on unseen data. YOLOv5 uses the modern Path Aggregation network (PANet) [7] as the model neck to produce feature pyramids. PANet intended to solve the issues in difficult propagation in terms of low-level features in Feature Pyramid Networks[34]

In the end, the ultimate job of detection is performed by the model Head. Anchor boxes are applied to the generated feature pyramids and produce the final output vector. The output vector consists of bounding boxes of predicted object's position, objectness scores, and the class probabilities. The configuration of the Head part of YOLOv5 is the same as in YOLOv3 and YOLOv4.

YOLOv5 is more optimized than YOLOv4 and YOLOv5 for loss function, network configuration, and data enhancement. Adaptive image scaling, Adaptive anchor box calculation, and Mosaic data enhancement techniques are utilized for the data enhancement.

Both YOLOv5 and YOLOv4 use mosaic data enhancement to

In the middle/hidden layers of YOLOv5 Leaky ReLU(Rectified Linear Unit) is used and in the final detection layer, the Sigmoid activation function is used. Both Stochastic Gradient Descent (SGD) and

ADAM optimizers are available to experiment with in YOLOv5. Binary Cross Entropy with Logits Loss function is used in the YOLOv5 method as loss function.

In YOLO v5 the Leaky ReLU activation function is used In middle/hidden layers and the sigmoid activation function is used in the final detection layer

There are several architectures in the YOLOv5 family that basically differs in their parameter size called P5 models. They are:

- YOLOv5n(nano)
- YOLOv5s(small)
- YOLOv5m (medium)
- YOLOv5l(large)
- YOLOv5x(extra large)

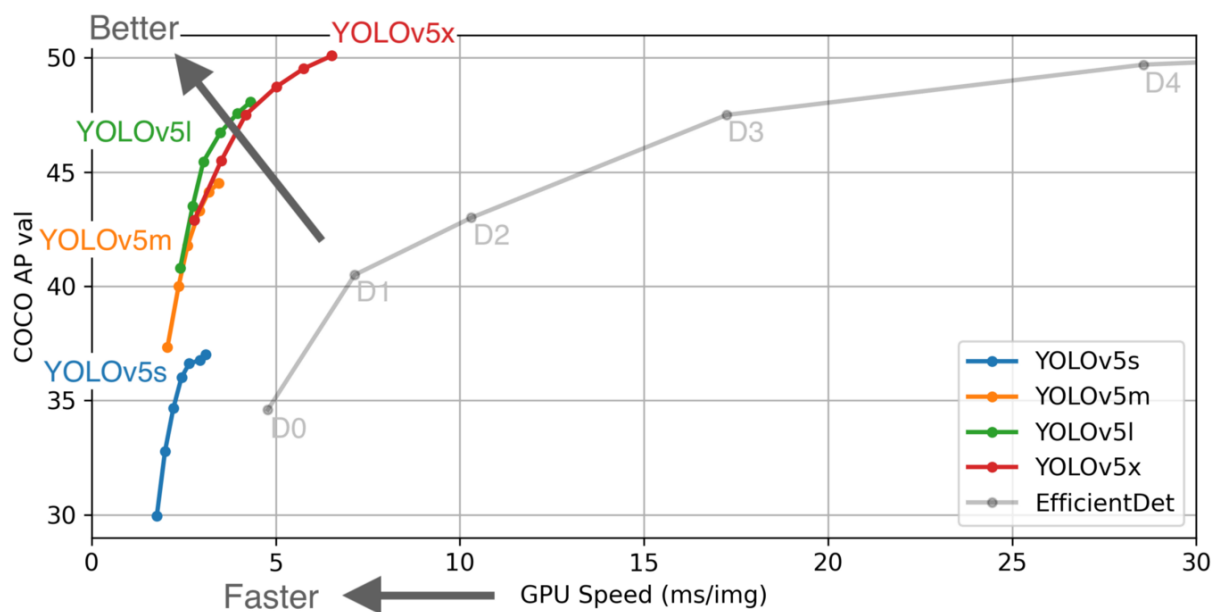
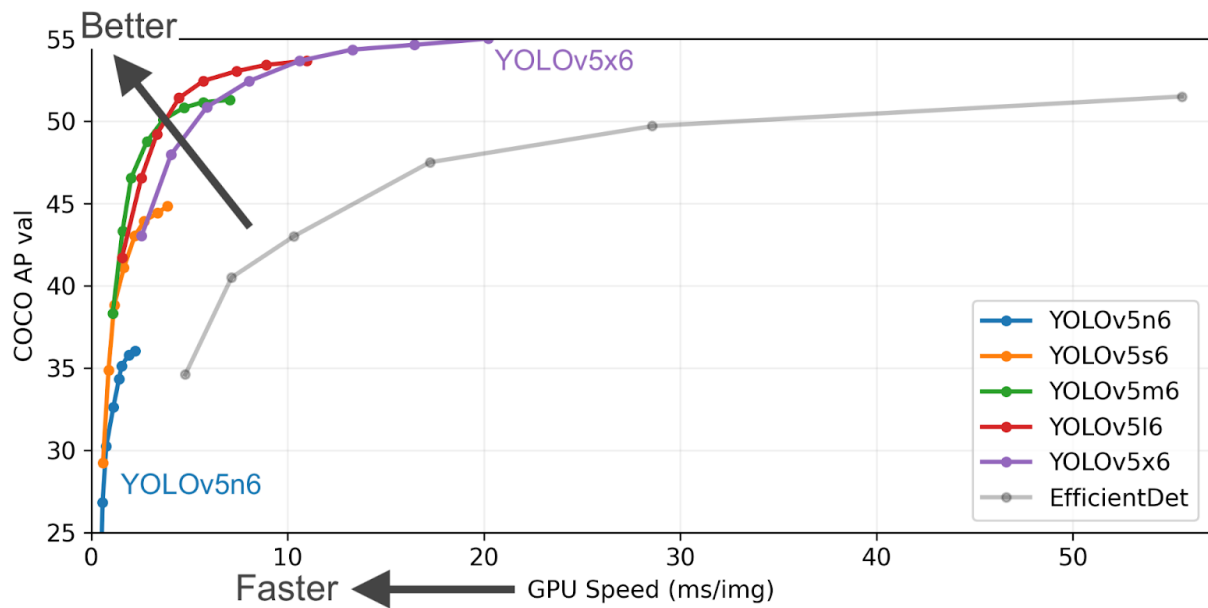


Fig. 15. Comparison of YOLOv5 model P5 versions

Additional series of models are also released named P6 models. They are:

- YOLOv5n6(nano)
- YOLOv5s6(small)
- YOLOv5m6 (medium)

- YOLOv5l6(large)
- YOLOv5x6(extra large)



Pretrained Checkpoints								
Model	size (pixels)	mAP ^{val} 0.5:0.95	mAP ^{val} 0.5	Speed CPU b1 (ms)	Speed V100 b1 (ms)	Speed V100 b32 (ms)	params (M)	FLOPs @640 (B)
YOLOv5n	640	28.0	45.7	45	6.3	0.6	1.9	4.5
YOLOv5s	640	37.4	56.8	98	6.4	0.9	7.2	16.5
YOLOv5m	640	45.4	64.1	224	8.2	1.7	21.2	49.0
YOLOv5l	640	49.0	67.3	430	10.1	2.7	46.5	109.1
YOLOv5x	640	50.7	68.9	766	12.1	4.8	86.7	205.7
YOLOv5n6	1280	36.0	54.4	153	8.1	2.1	3.2	4.6
YOLOv5s6	1280	44.8	63.7	385	8.2	3.6	12.6	16.8
YOLOv5m6	1280	51.3	69.3	887	11.1	6.8	35.7	50.0
YOLOv5l6	1280	53.7	71.3	1784	15.8	10.5	76.8	111.4
YOLOv5x6	1280	55.0	72.7	3136	26.2	19.4	140.7	209.8
+ TTA	1536	55.8	72.7	-	-	-	-	-

Fig. 15. Comparison of YOLOv5 model P5 and P6 versions

3.5 Evaluation Metrics

For measurement of the performance of object detectors, multiple criteria can be adopted such as frame per second (fps), Precision, Recall, F1 score, etc. However, the most common evaluation metric is Mean Average Precision (mAP). Firstly, Intersection over Union (IoU) is calculated which is the ratio of the area of overlap (between the predicted bounding box and truth bounding box) and the area of union. Primarily a threshold value is fixed to decide whether detection is true positive or false positive.

When the IoU score is greater than the threshold value, the prediction is classified as True Positive, and when the Iou score is less than the threshold value the prediction is classified as False Positive.

$$\begin{aligned} \textit{Precision} &= \frac{\textit{True Positive}}{\textit{True Positive} + \textit{False Positive}} \\ &= \frac{\textit{True Positive}}{\textit{All Observations}} \end{aligned} \quad (1)$$

$$\begin{aligned} \textit{Recall} &= \frac{\textit{True Positive}}{\textit{True Positive} + \textit{False Negative}} \\ &= \frac{\textit{True Positive}}{\textit{All Ground Truth}} \end{aligned} \quad (2)$$

Precision represents the percentage of accurate prediction for all observations. The recall represents the percentage of accurate predictions with respect to the original ground truth.

Average Precision is calculated for all the existing classes separately using the above equation.

Consequently, the mean average precision of all classes is calculated which is known as mean average precision (mAP). This metric is used to compare the performance of different object detector methods.

Chapter 4: Experimental Results and Discussion

4.1 Training and Validation Result

Evaluation metrics during training and validation of different architectures of YOLOv5, EfficientDet D-01, and FasterRCNN-ResNet152 are shown below with multiple figures:

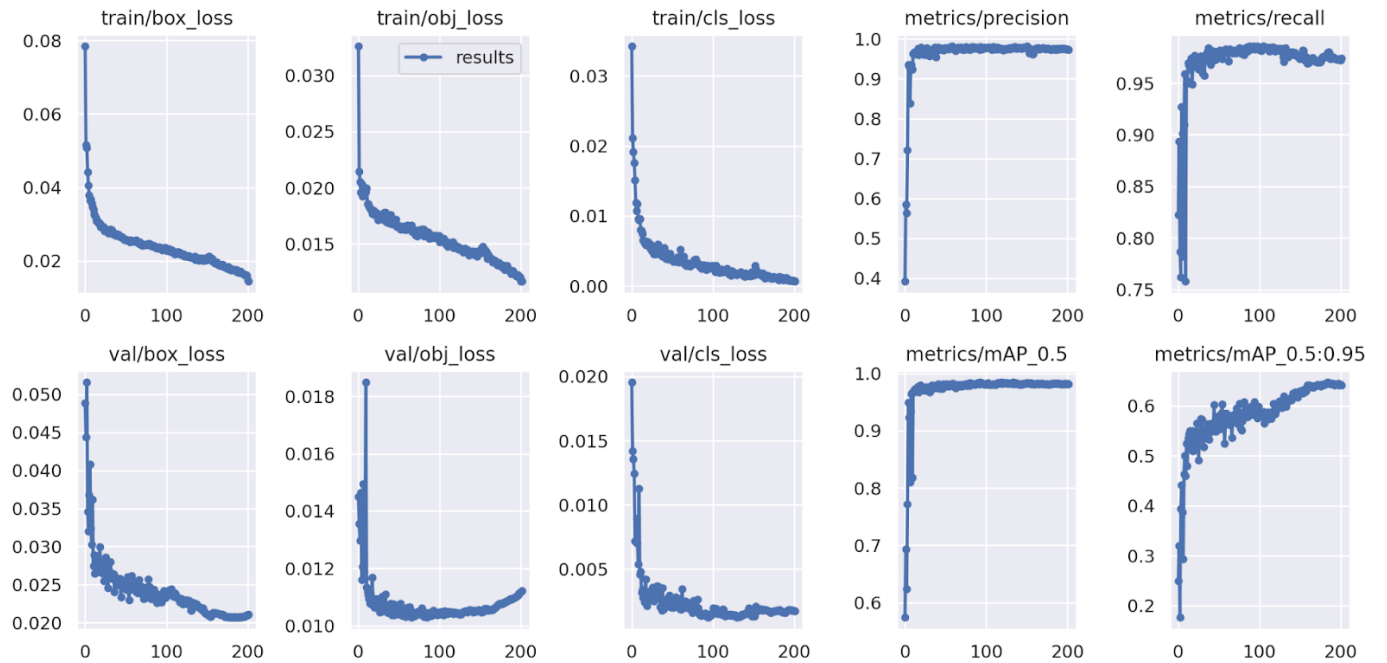


Fig. 16: YOLOv5 - Extra Large Results

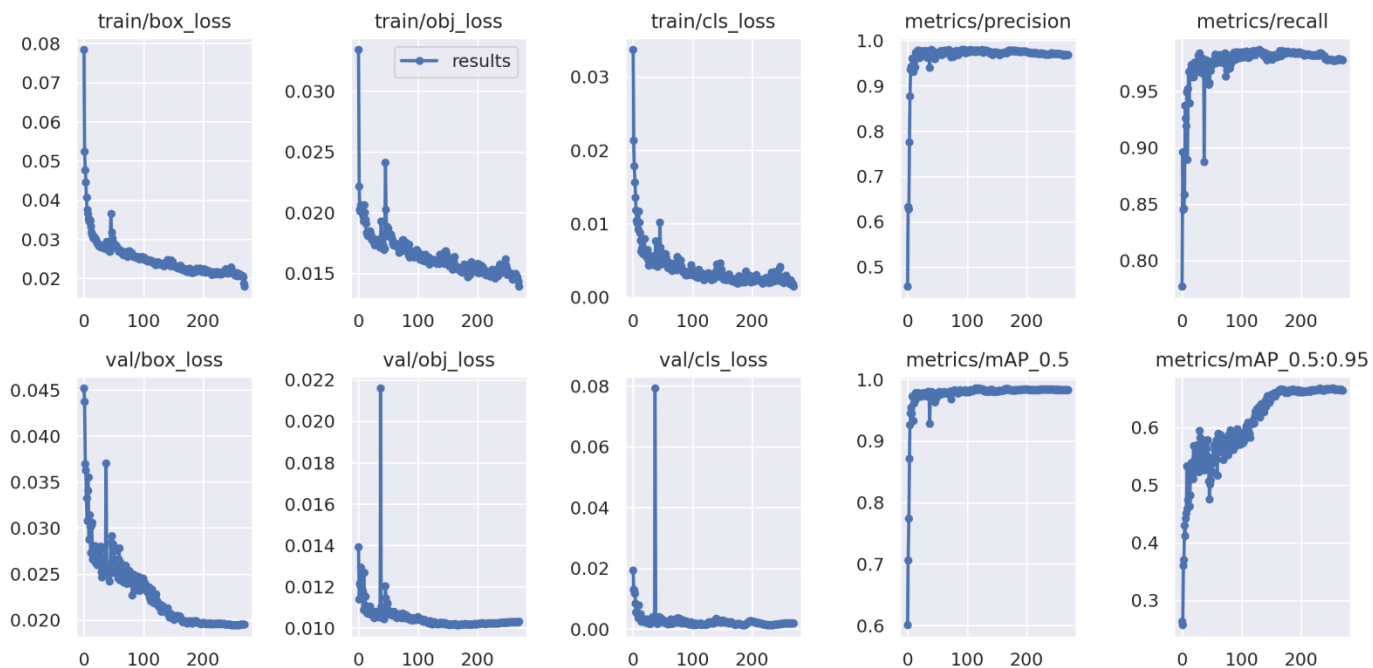


Fig. 17: YOLOv5 - Large Results

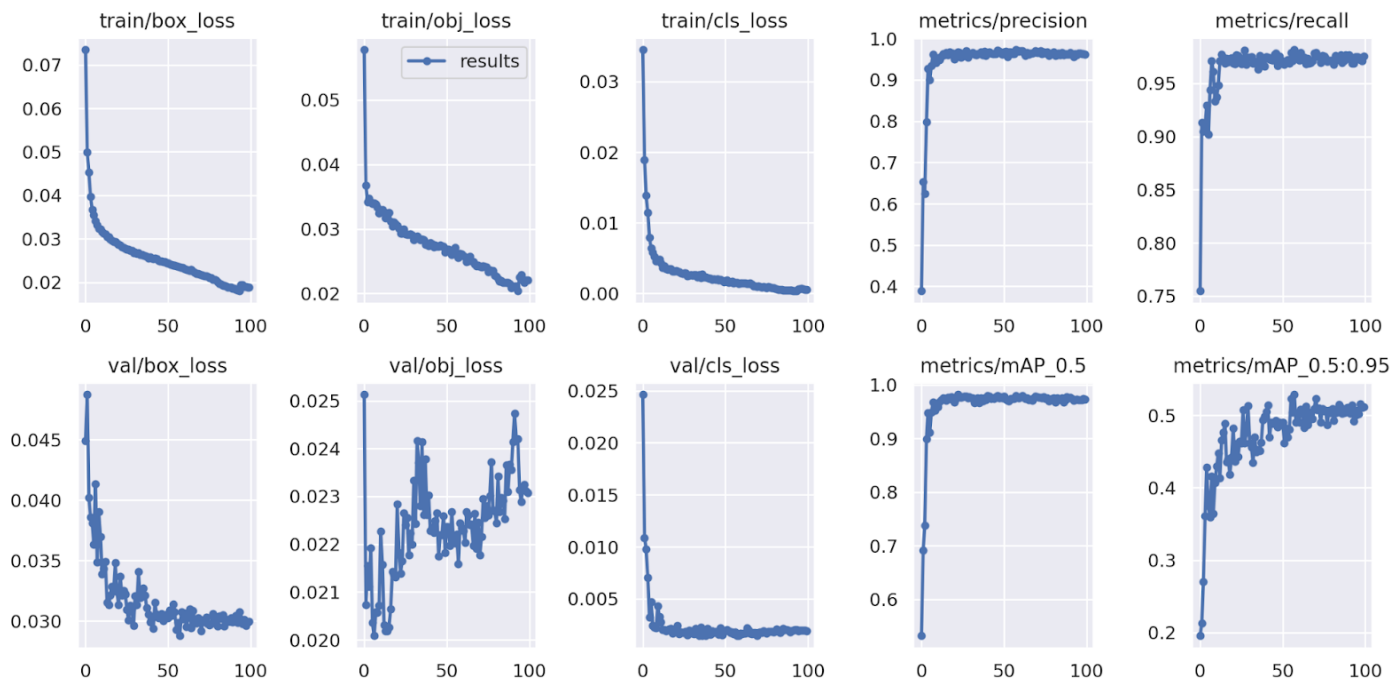


Fig. 18: YOLOv5 - Small Results

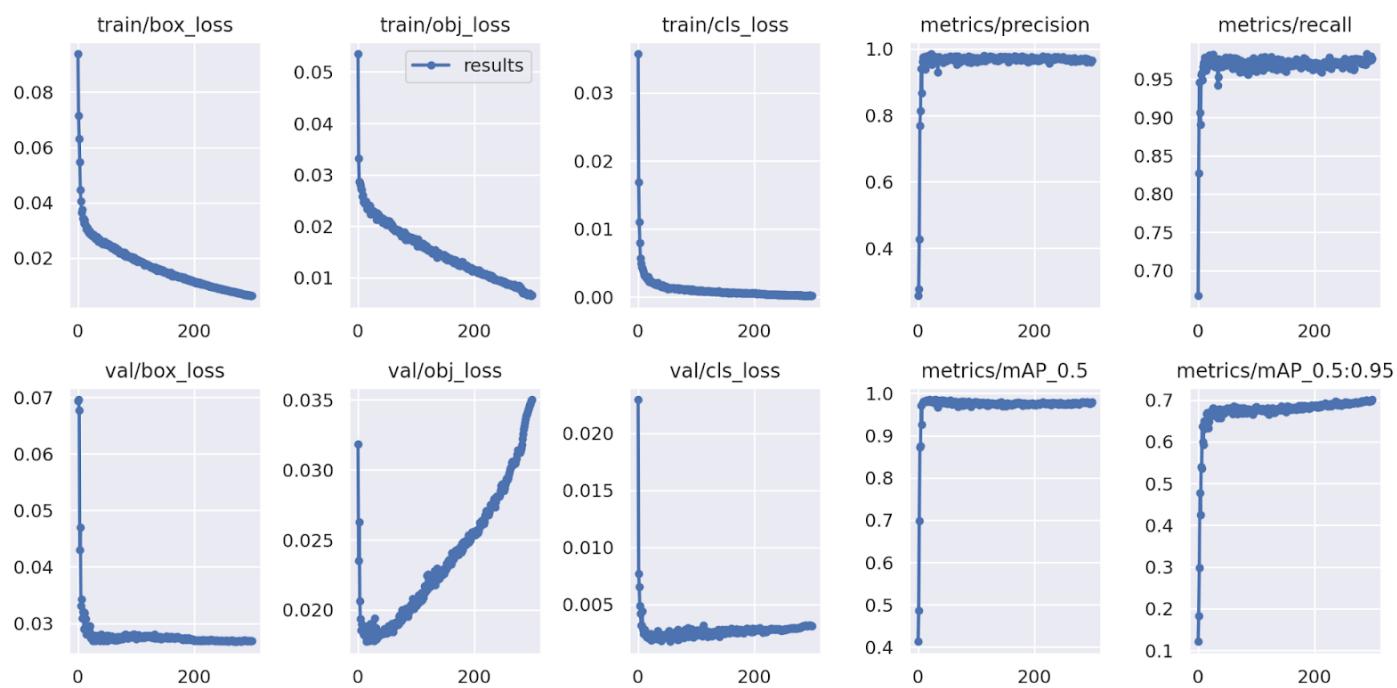


Fig. 19: YOLOv5 – Results in Extra Large with Plain Dataset

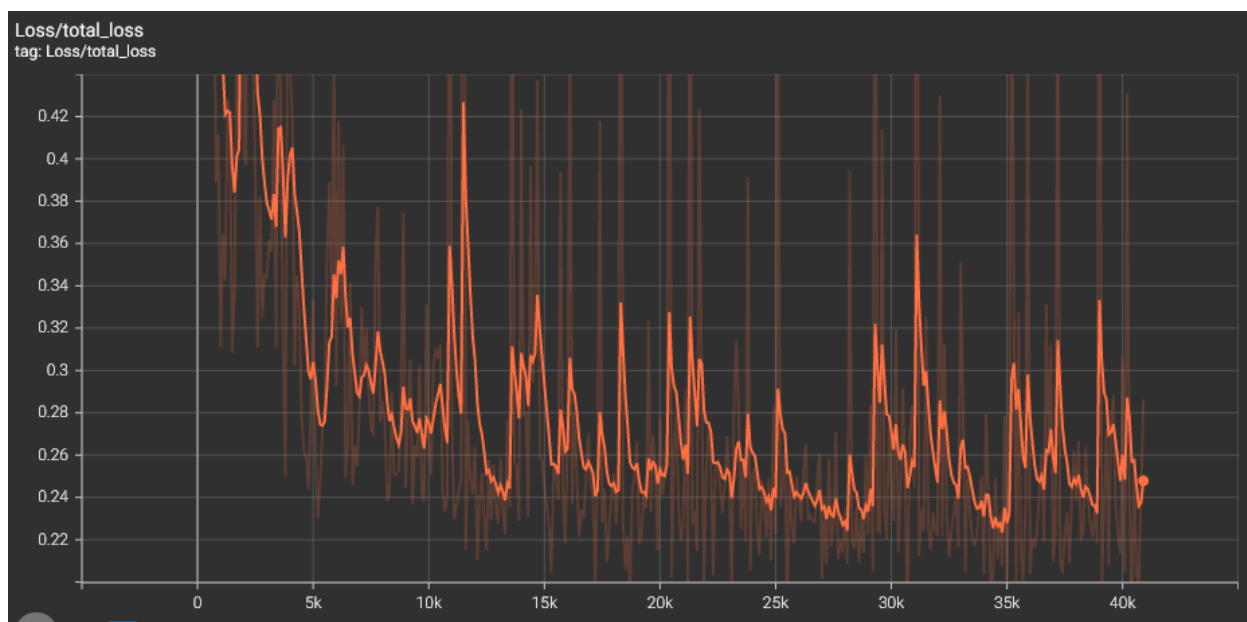


Fig. 20: Results of EfficientDet-D01

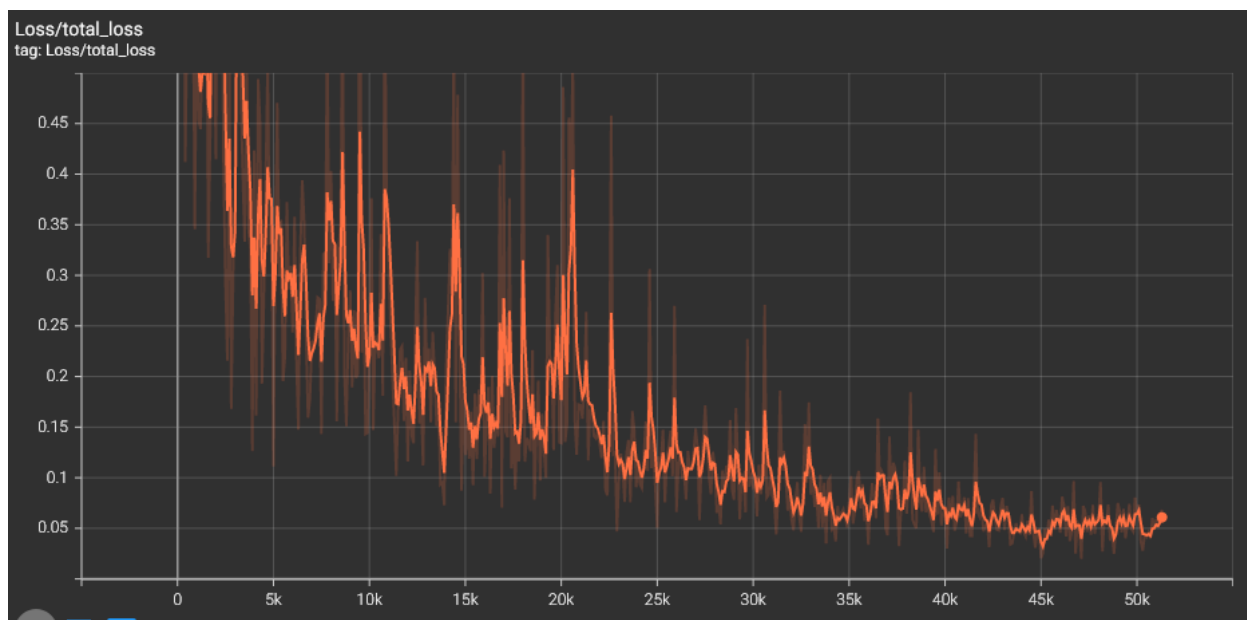


Fig. 21: Results of FasterRCNN-ResNet152

F1 curve of different architectures of YOLOv5 are shown below with multiple figures:

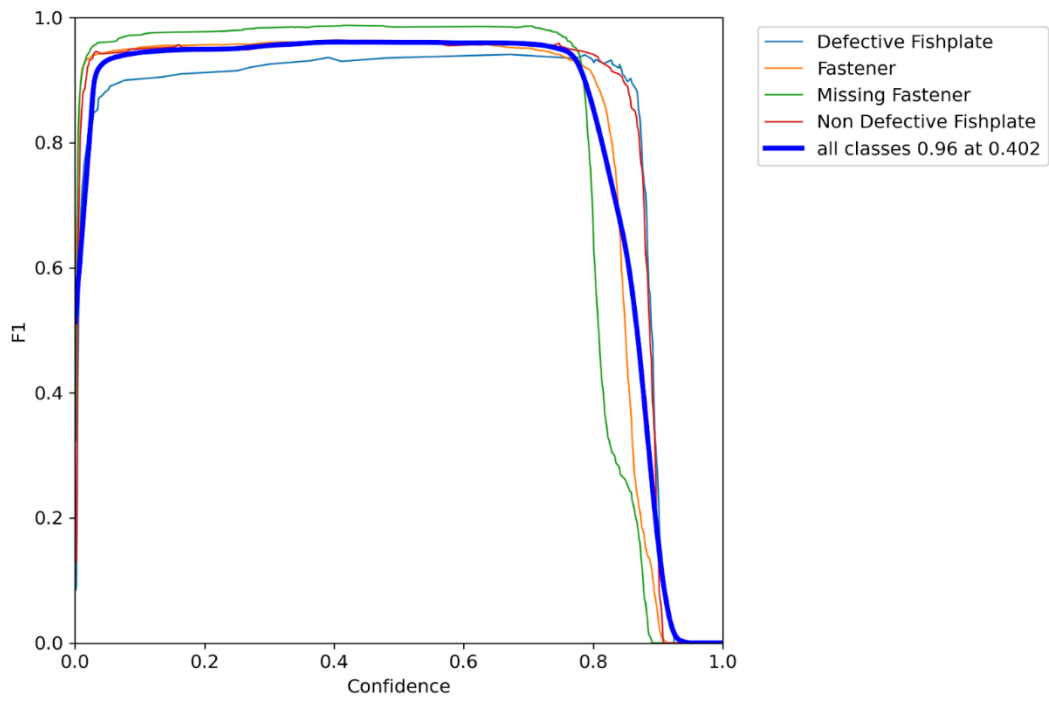


Fig. 22: F1 curve of YOLOv5- Extra Large

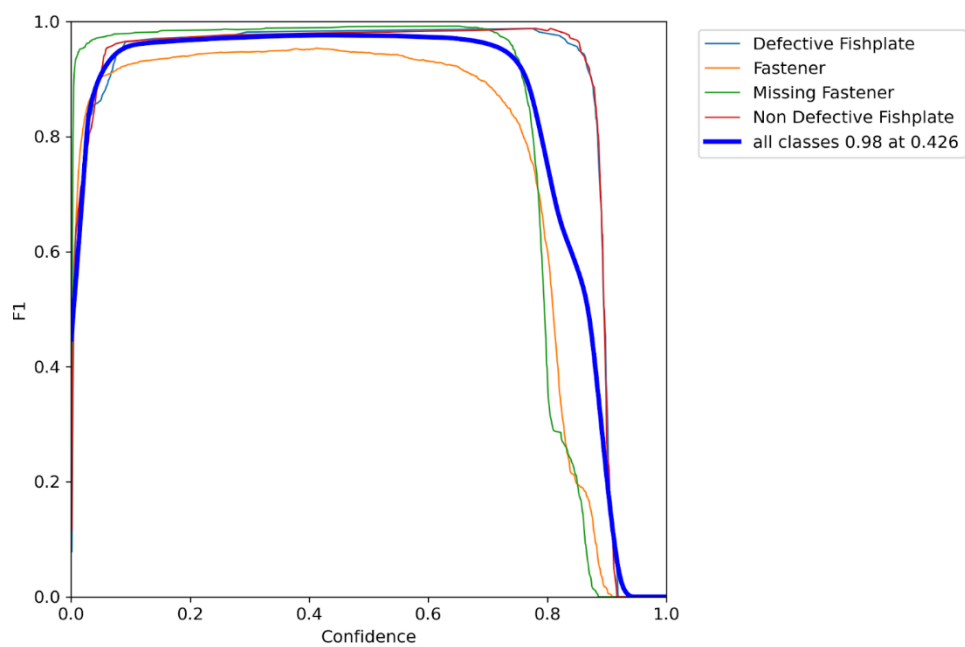


Fig. 23: F1 curve of YOLOv5- Large

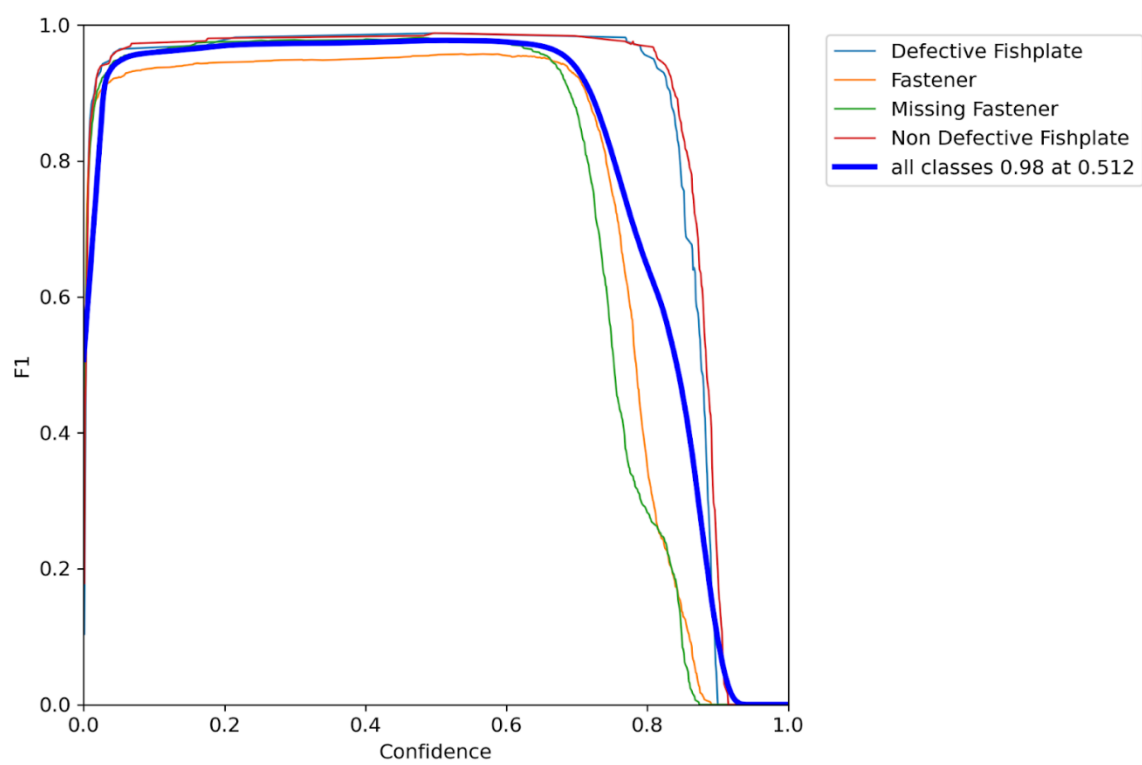


Fig. 24: F1 curve of YOLOv5- Small

Confusion Matrix of Different Architectures of YOLOv5 family is shown below with multiple figures.

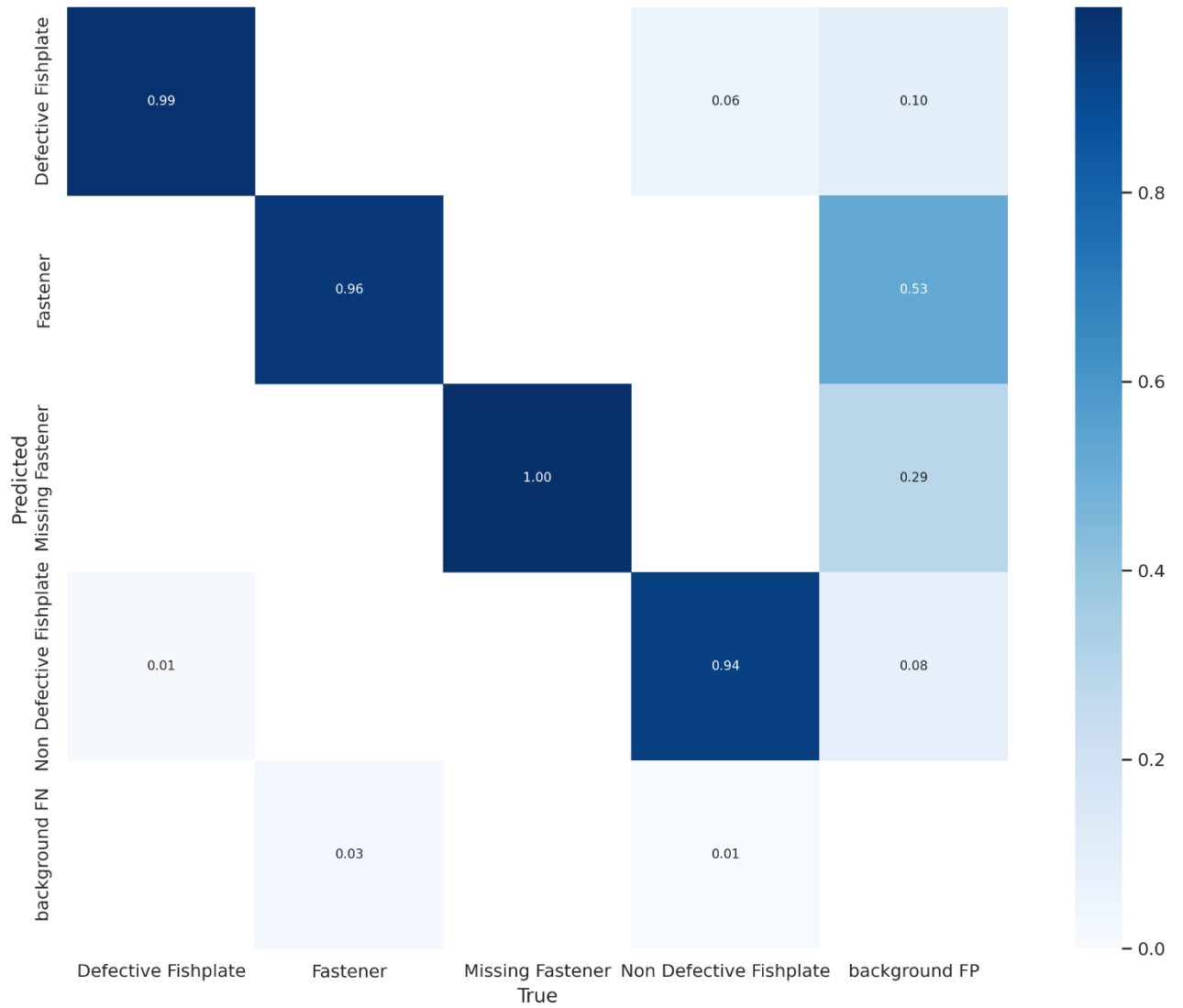


Fig. 25: Confusion Matrix of YOLOv5 – Extra Large

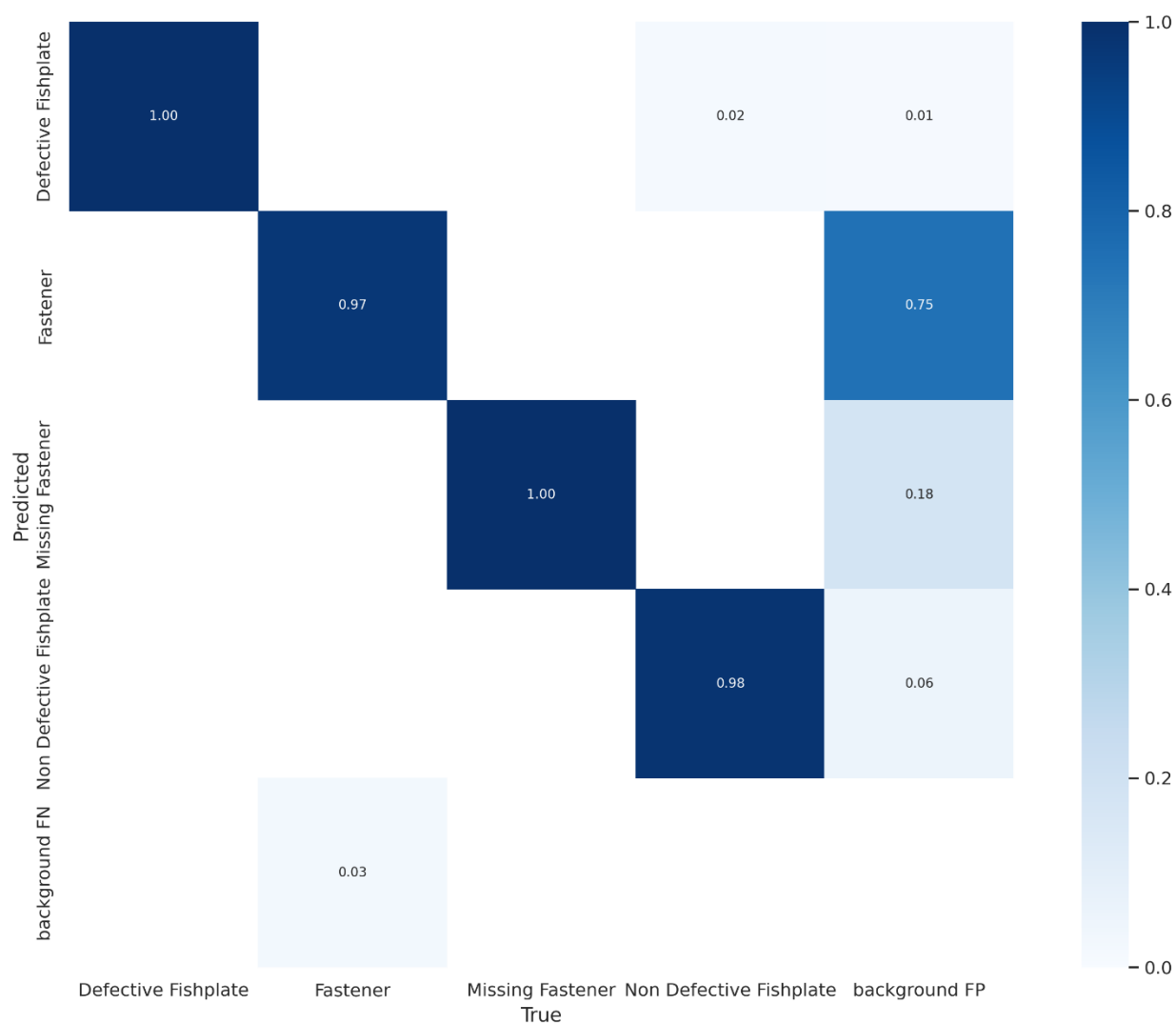


Fig. 26: Confusion Matrix of YOLOv5 -Large

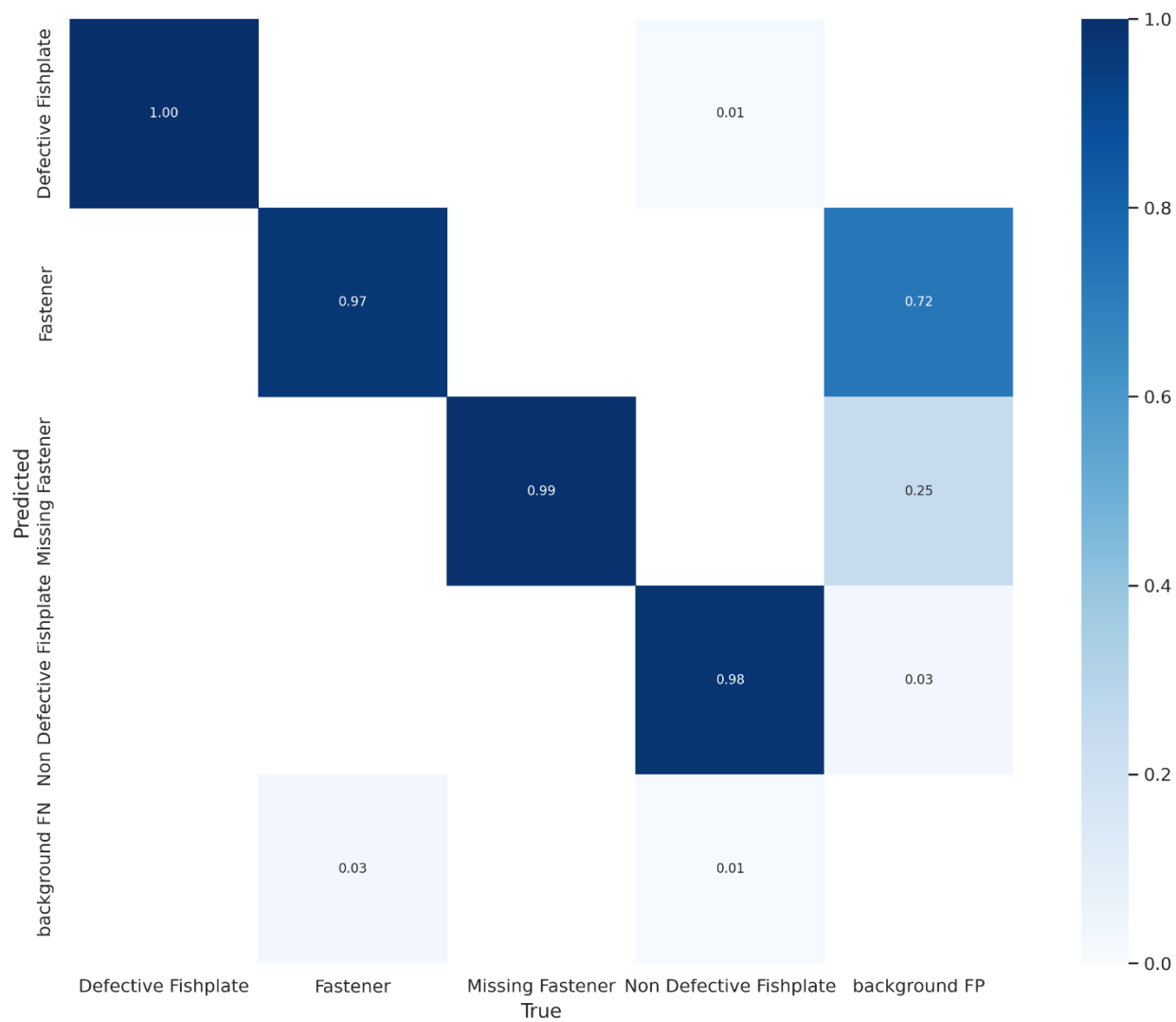


Fig. 27: Confusion Matrix of YOLOv5 –Small

4.2 Comparative Analysis

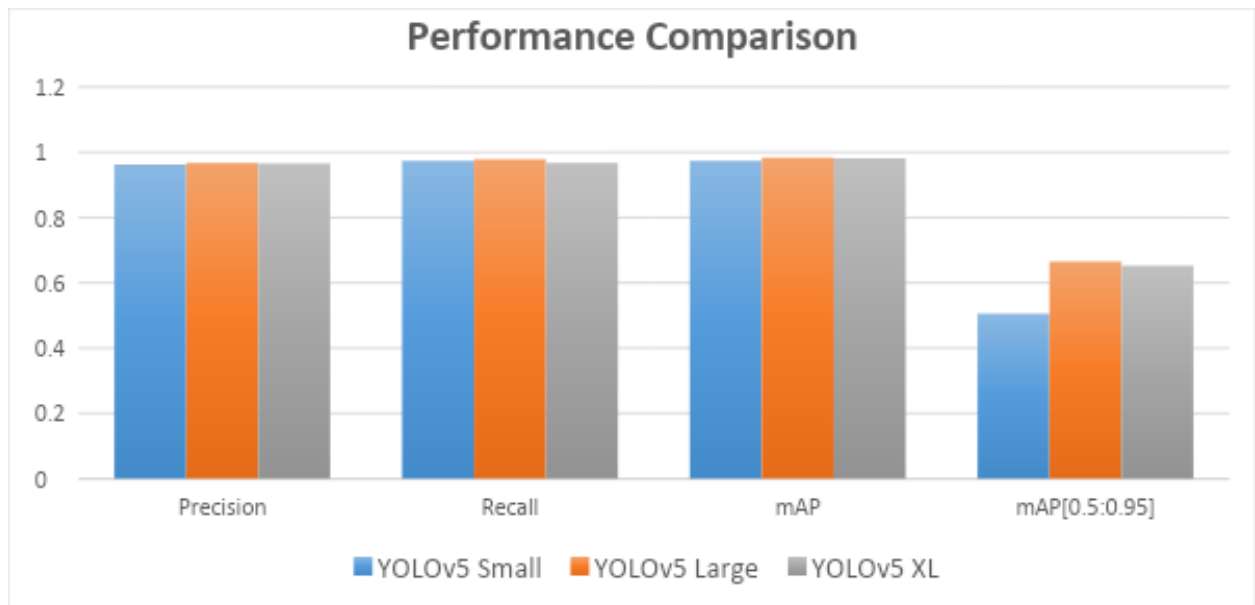


Fig. 28: Evaluation Metrics Score Comparison of different YOLOv5 architecture

We can clearly observe in the figure that the score of mAP at 0.5 IoU threshold for all the YOLOv5 small, large, and extra-large models are very identical. But in terms of mAP[0.5:0.95], the YOLOv5 extra-large and large model performed better than the small model.



Fig. 29: Loss score Comparison of different YOLOv5 architecture

We can see that after training certain epochs overall loss score for YOLOv5 extra-large is less than YOLOv5 small and large version. But surprisingly class loss of YOLOv5 small is less than YOLOv5 large and extra-large

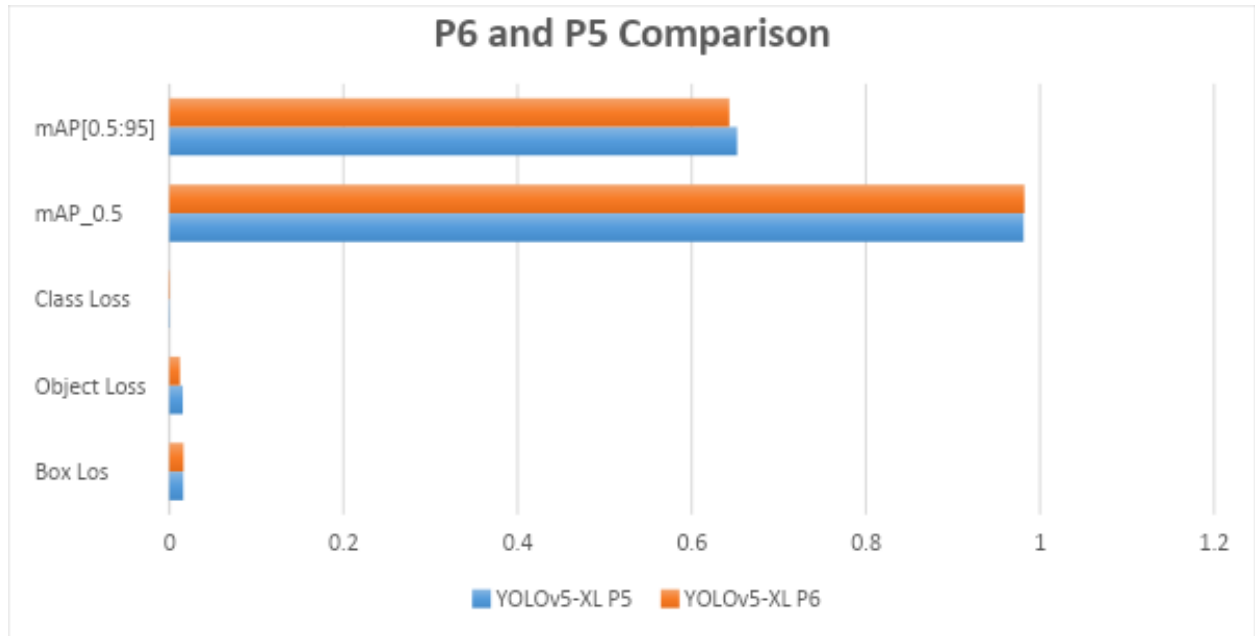


Fig. 30: Performance Comparison of P5 and P6 YOLOv5 architecture

From the figure, we can depict that both P5 and P6 models performed identically and differentiate margin is very low, even though the P6 version of a YOLOv5 model is heavier and slower than the P5 version

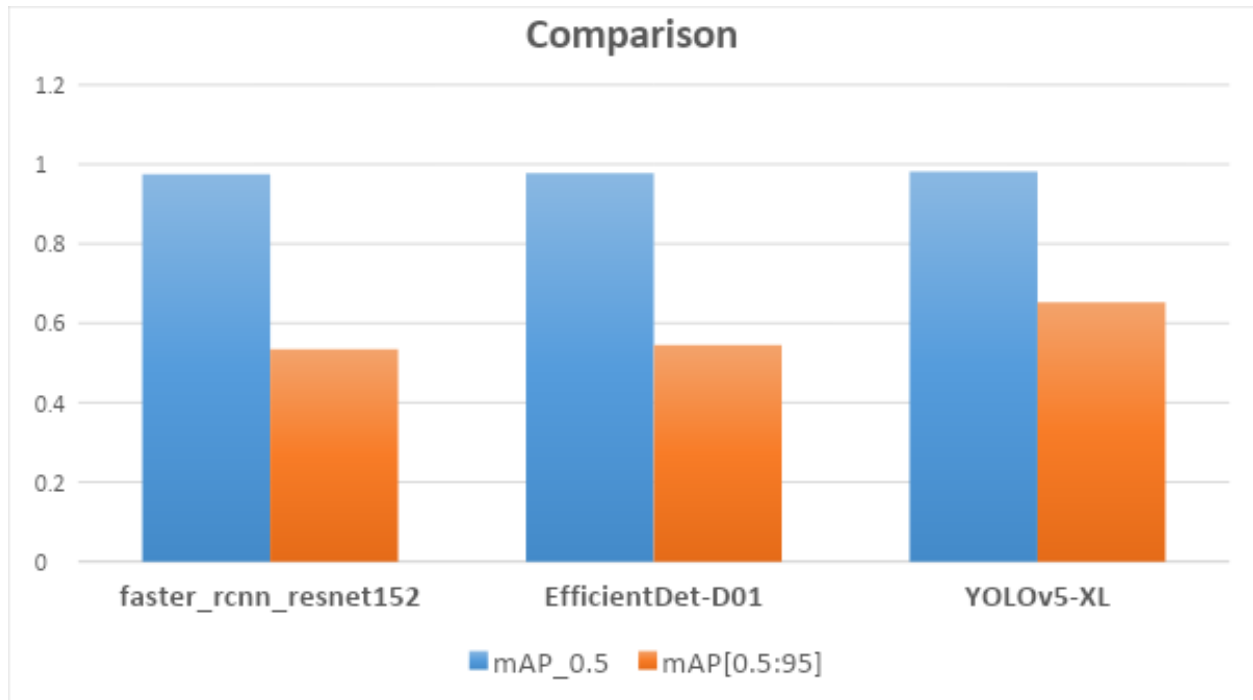


Fig. 31: Performance Comparison of YOLOv5 XL, EfficientDet-D01, FasterRCNN-ResNet152

It is very obvious from the figure that in terms of mAP[0.5:0.95], YOLOv5 outperformed EfficientDet-D01 and FasterRCNN-ResNet152

4.3 Inference with Trained Model

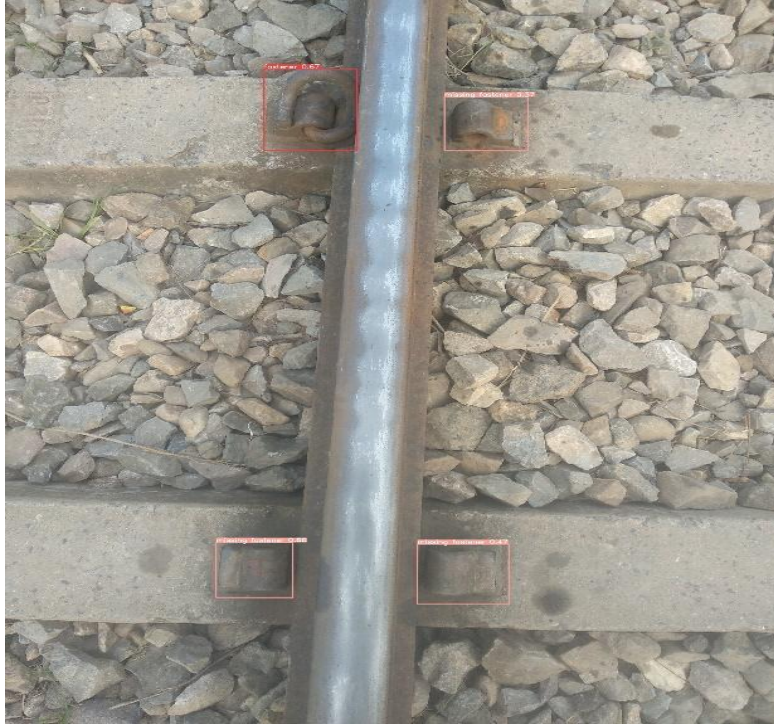


Fig. 32: Inferencing an image having a single line

Though all the training and validation data consist of double rail from the upper view angle, the model performed accurately in an image consisting single rail from the upper view angle.



Fig. 33: Inferencing an image with double rail and front viewing angle

Though all the training and validation data are from the upper view angle, the model performed accurately in an image consisting of double rail from another view angle.



Fig. 34. Inferencing an image from the side angle

The model performed accurately on the image that is captured from a side angle and consists of multiple instances of different sizes of a class.

Chapter 5: Conclusion and Implication for Future Research

5.1 Summary of the Study and Conclusions

- Comparing the experimental results of evaluation metrics like mAP, mAP@[0.5:0.95], for trained models YOLOv5(multiple architectures), EfficientDet, and FasterRCNN-ResNet152, It is obvious that YOLOv5 Large and Extra Large models outperformed both EfficientDet and FasterRCNN-ResNet152. However, scores of mAP@[0.5:0.95] for YOLOv5 Small, EfficientDet, and FasterRCNN-ResNet152 are very close to each other.
- Among the architectures of different sizes in the YOLOv5 framework, architectures consisting of more parameters performed better in my custom dataset.
- Though accuracy is better for longer architectures, there is a trade-off between accuracy and detection speed.
- Results of the experimented models like YOLOv5(XL), YOLOv5(Large), YOLOv5(Small) are obtained better with starting learning rate = 0.01, ending learning rate = 0.0001, momentum = 0.937, SGD optimizer, batch size = 16.
- Though the score of evaluation metrics with the Model that is trained with the plain dataset is better than the Models that are trained with the augmented dataset, a model trained with the augmented dataset performed better in the inference with unseen data.
- YOLOv5 (Large) and YOLOv5(XL) performed almost identically in the dataset through YOLOv5(XL) takes much more time in inference time.
- Detection Speeds during training are validation are obtained as follows YOLOv5(small) > YOLOv5 (large) > YOLOv5 (Extra Large) > EfficientDet > FasterRCNN-152
- Thus, concerning the detection speed, YOLOv5(Large) is a better performer model in the dataset.

5.2 Implication for Further Study

- Though the YOLOv5 models are performing inference in real time with great accuracy, the model can be optimized by changing the configuration in the YOLOv5 model backbone; CSPNet.
- The techniques or methods to object tracking and counting with YOLOv5 can be a subject of further study which will be required to implement the inspection system in the real world.
- Overall, the procedure to implement the automated inspection system with the proposed defect detection method can be studied further.

References

- [1] S. Morales-Ivorra, J. I. Real, C. Hernández, and L. Montalbán, "Derailment risk and dynamics of railway vehicles in curved tracks: Analysis of the effect of failed fasteners," *J. Mod. Transp.*, vol. 24, no. 1, pp. 3847, 2016, doi: 10.1007/s40534-015-0093-z.
- [2] X. Xiao, X. Jin, and Z. Wen, "Effect of disabled fastening systems and ballast on vehicle derailment," *J. Vib. Acoust.*, vol. 129, no. 2, pp. 217229, 2007, doi: 10.1115/1.2424978.
- [3] T. Tong, Y. Luo, and L. Li, "Research on the influence of rail fastener stiffness upon dynamic performance of vehicle-track system," in *Proc. 7th Int. Conf. Measuring Technol. Mechatronics Automat. (ICMTMA)*, 2015, pp. 923925, doi: 10.1109/icmtma.2015.227.
- [4] Roberto Sañudo Ortega, Joao Pombo, Stefano Ricci, Marina Miranda, The importance of sleepers spacing in railways, *Construction and Building Materials*, Volume 300, 2021, 124326, ISSN 0950-0618, <https://doi.org/10.1016/j.conbuildmat.2021.124326>
- [5] “সিলেটের রেলপথ যেন ‘মৃত্যুফাঁদ’”, Wais Khasru, Manab Zamin Online, 07/02021, <https://mzamin.com/article.php?mzamin=261849>
- [6] Wang, Chien-Yao & Liao, Hong-yuan & Wu, Yuen-Hua & Chen, Ping-Yang & Hsieh, Jun-Wei & Yeh, I-Hau. (2020). CSPNet: A New Backbone that can Enhance Learning Capability of CNN. 1571-1580. 10.1109/CVPRW50498.2020.00203.
- [7] Liu, Shu & Qi, Lu & Qin, Haifang & Shi, Jianping & Jia, Jiaya. (2018). Path Aggregation Network for Instance Segmentation.

- [8] C. Aytekin, Y. Rezaei Tabar, S. Dogru, and I. Ulusoy, "Railway fastener inspection by real-time machine vision," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 45, no. 7, pp. 1101–1107, 2015
- [9] H. Trinh, N. Haas, Y. Li, C. Otto, and S. Pankanti, "Enhanced rail component detection and consolidation for rail track inspection," in *Proceedings of the 2012 IEEE Workshop on the Applications of Computer Vision*, pp. 289–295, Breckenridge, CO, USA, January 2012.
- [10] X. Wei, Z. Yang, Y. Liu et al., "Railway track fastener defect detection based on image processing and deep learning techniques: a comparative study," *Engineering Applications of Artificial Intelligence*, vol. 80, pp. 61–81, 2019
- [11] E. Resendiz, J. Hart, and N. Ahuja, "Automated visual inspection of railroad tracks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 751–760, 2013.
- [12] X. Gibert, V. M. Patel, and R. Chellappa, "Robust fastener detection for autonomous visual railway track inspection," in *Proceedings of the 2015 IEEE Winter Conference on Applications of Computer Vision*, pp. 694–701, IEEE, Waikoloa, HI, USA, January 2015
- [13] L. Liu, F. Zhou, and Y. He, "Automated status inspection of fastening bolts on freight trains using a machine vision approach," *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, vol. 230, no. 7, pp. 1629–1641, 2016
- [14] P. D. Ruvo, A. Distanto, E. Stella, and F. Marino, "A GPUbased vision system for real time detection of fastening elements in railway inspection," in *Proceedings of the 2009 16th IEEE International Conference on Image Processing*, pp. 2309–2312, Cairo, Egypt, February 2010.
- [15] F. Marino, A. Distanto, P. L. Mazzeo, and E. Stella, "A realtime visual inspection system for railway maintenance: automatic hexagonal-headed bolts detection," *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, vol. 37, no. 7, pp. 418–428, 2007.

- [16] P. Dollar and C. L. Zitnick, "Fast edge detection using structured forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 8, pp. 1558–1570, 2015
- [17] Y. Xia, F. Xie, and Z. Jiang, "Broken railway fastener detection based on adaboost algorithm," in *Proceedings of the 2010 International Conference on Optoelectronics and Image Processing*, pp. 313–316, Haikou, China, November 2010.
- [18] H. Feng, Z. Jiang, F. Xie, P. Yang, J. Shi, and L. Chen, "Automatic fastener classification and defect detection in vision-based railway inspection systems," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 4, pp. 877–888, 2014.
- [19] J. Yang, W. Tao, M. Liu, Y. Zhang, H. Zhang, and H. Zhao, "An efficient direction field-based method for the detection of fasteners on high-speed railways," *Sensors*, vol. 11, no. 8, pp. 7364–7381, 2011.
- [20] E. Resendiz, J. Hart, and N. Ahuja, "Automated visual inspection of railroad tracks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 751–760, 2013.
- [21] X. Wei, Z. Yang, Y. Liu et al., "Railway track fastener defect detection based on image processing and deep learning techniques: a comparative study," *Engineering Applications of Artificial Intelligence*, vol. 80, pp. 61–81, 2019.
- [22] Xavier Gibert; Vishal M. Patel; Rama Chellappa, "Robust Fastener Detection for Autonomous Visual Railway Track Inspection" 2015 IEEE Winter Conference on Applications of Computer Vision, Jan. 2015
- [23] L. Li, R. Sun, S. Zhao et al., "Semantic-Segmentation-Based rail fastener state recognition algorithm," *Mathematical Problems in Engineering*, vol. 2021, Article ID 8956164, 15 pages, 2021.
- [24] ZIWEN ZHANG , MANGUI LIANG, AND ZHE WANG, "A Deep Extractor for Visual Rail Surface Inspection", *IEEE Access* PP(99):1-1, January 2021

- [25] A. Ma, Z. Lv, X. Chen et al., “Pandrol track fastener defect detection based on local convolutional neural networks” Proceedings of Institution of Mechanical Engineers Part I-Journal of Systems and Control Engineering, 2020
- [26] Chandran, Praneeth. (2021). An Investigation of Railway Fastener Detection Using Image Processing and Augmented Deep Learning. Sustainability. 13. 10.3390/su132112051.
- [27] X. Gibert, V. M. Patel, and R. Chellappa, “Deep multitask learning for railway track inspection,” IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 1, pp. 153–164, 2017
- [28] Dechen Yao, Qiang Sun, Jianwei Yang, Hengchang Liu, Jiao Zhang³, “Railway Fastener Fault Diagnosis Based on Generative Adversarial Network and Residual Network Model” SN - 1070-9622,, Shock and Vibration PB - Hindawi, <https://doi.org/10.1155/2020/8823050>
- [29] Wang, Tiange & Yang, Fangfang & Tsui, Kwok-Leung. (2020). Real-Time Detection of Railway Track Component via One-Stage Deep Learning Networks. Sensors. 20. 4325. 10.3390/s20154325.
- [30] Lin, Wendy & Hsieh, Chen-Chiung & Huang, Wei-Hsin & Hsieh, Sun-Lin & Hung, Wei-Hung. (2019). Railway Track Fasteners Fault Detection using Deep Learning. 187-190. 10.1109/ECICE47484.2019.8942769.
- [31] Tangbo Bai , Jialin Gao, Jianwei Yang , “A Study on Railway Surface Defects Detection Based on Machine Vision” Entropy (Basel). 2021 Nov
- [32] Danyang Zheng , Liming Li, Shubin Zheng, “A Defect Detection Method for Rail Surface and Fasteners Based on Deep Convolutional Neural Network” Computational Intelligence and Neuroscience, Volume 2021, Article ID 2565500, 15 pages, 2021
- [33] K. Han, Y. Wang, Q. Tian et al., “GhostNet: more features from cheap operations,” in Proceedings of the 2020 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1580–1589, Seattle, WA, USA, June 2020.

[34] Lin, Tsung-Yi & Dollar, Piotr & Girshick, Ross & He, Kaiming & Hariharan, Bharath & Belongie, Serge. (2017). Feature Pyramid Networks for Object Detection. 936-944. 10.1109/CVPR.2017.106.