BIOE 6306 ADVANCED ARTIFICIAL NEURAL NETWORKS

SAFETY HELMET DETECTION USING DEEP LEARNING: IMPLEMENTATION AND COMPARATIVE STUDY USING YOLOv7 AND YOLOv8

By

NITHEESH KUMAR GORLA

2193851

Table of content

ABSTRACT

Safety helmets play a key role, in ensuring the well-being of employees working in sectors like construction, power and manufacturing. However, it is common for employees to take off their helmets due to a lack of awareness and inconvenience which puts them at a risk of accidents. Incidents such as falls, electrical shocks can result in severe injuries for those without helmets. Therefore, it is essential for management to have an accurate method of detecting helmet usage. Traditional human monitoring methods are time consuming and challenging to implement when dealing with a workforce. The conventional approach to helmet identification lacks accuracy and resilience making it unsuitable for use in the industry. In this project we proposed a deep learning technique that employs You Only Look Once (YOLO) models for helmet detection with high accuracy. The aim of this project is to implement and compare the performance of two deep learning models, YOLOv7 and YOLOv8 for safety helmet detection.

INTRODUCTION

In today's fast-paced working environment there has been an increase, in construction activities, unfortunately accidents occurring at construction sites due to the absence of safety regulations have become quite common posing a threat to both lives and property. Safety helmets play a role in protecting workers lives. It's unfortunate that many construction workers do not use them due to a lack of awareness about safety measures. The existing methods of inspecting and monitoring images and video footage to ensure helmet usage are, often result in missed inspections and delays. To prevent any such incidents, it is essential to implement a real time detection system for safety helmets.

Over the years there have been remarkable advancements in computer vision and deep learning technologies. These advancements have led to the development of applications for object detection. Object detection involves both classifying objects and determining their locations accurately. Currently object detectors can be categorized into two types; networks that separate the tasks of locating objects and classifying them (like R-CNN) and networks that simultaneously predict bounding boxes and class scores (like YOLO and SSD networks).

The two-stage algorithm is considered accurate, in theory compared to the one stage approach. The Faster RCNN system is a two-step approach, for detecting targets. It consists of components, including a layer, a region candidate network, a region of interest pooling layer and a classification layer. These components work together to extract features and create feature maps. When it comes to target detection single stage detectors are faster than two stage detectors. The YOLO series algorithms are particularly impressive in object detection because they use single stage methods. In this approach the targets position is determined through regression. Its category is identified simultaneously. Only deep convolutional layers are used in the YOLO series with input images serving as the basis for the network. By applying convolution operations to these images both the targets category and position can be obtained.

Each yolo model consists of three components: Backbone, Neck and Head. The backbone of YOLO models plays a role in processing input images and extracting features that represent various patterns, edges and textures present in the image. Typically, the backbone consists of a trained convolutional neural network (CNN) architecture such as Darknet, CSPDarknet or ResNet. These architectures have been trained on large scale image classification tasks, like ImageNet. The initial training, on these classification tasks allows the main part of the model to learn a range of features that can be applied to various computer vision tasks, such as detecting objects.

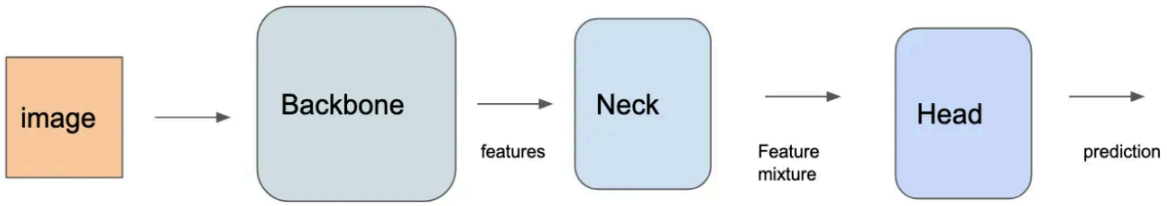


Fig1: High level architecture of yolo models

The features extracted by the backbone are then passed through layers of the YOLO model, including the middle and final layers. These layers further process the features and make predictions about bounding boxes and class probabilities for object detection.YOLO models often find a balance between accuracy and speed by selecting an architecture, for their needs based on application requirements.

DATASET

Dataset link: https://universe.roboflow.com/roboflow-universe-projects/construction-site-safety/dataset/27

I have taken this dataset from the Roboflow website (Roboflow computer vision datasets), the "Construction Site Safety dataset." The main objective of this dataset is to identify and locate workers who are wearing hats, safety vests, masks and other construction equipment. Hard hats are commonly used in construction and industrial settings to protect workers from hazards like falling objects.The dataset consists of 2,799 images, which are divided into three sets: 2,603, for training purposes 114 for validation purposes and 82 for testing purposes. Each image, in the dataset has been annotated using Roboflows annotation tool with bounding boxes that indicate where the targets are located. These annotations serve as ground truth data to train and evaluate object detection models.

The dataset contains total 10 target labels:

- Hardhat - Mask - NO-Hardhat
- NO-Mask - NO-Safety Vest - Person
- Safety Cone - Safety Vest - machinery - vehicle

Roboflow is a computer vision platform that facilitates faster and more accurate development of computer vision models by offering data collection methods, pre-processing techniques and model training capabilities. With Roboflow users can upload their custom datasets create annotations through drawing tools adjust image orientations or sizes as needed modify image contrast levels or apply data augmentation techniques and it also supports model training as well.

ARCHITECTURE OVERVIEW

The model architecture of YOLOv7

YOLOv7 is the most accurate real time object detection model, in computer vision tasks. In July 2022 a research paper titled "YOLOv7; Trainable bag of freebies sets new state of the art for real time object detectors" was published by Chien Yao Wang, Alexe Bochkovskiy and Hong Yuan Mark Liao.

YOLOv7 is developed by the same authors of Scaled YOLOv4 and YOLO R. In the YOLOv7 they introduced major changes by including

- E ELAN (Extended Efficient Layer Aggregation Network) - E ELAN refers to the block in the backbone of YOLOv7. The architecture of E ELAN allows YOLOv7 to enhance its learning ability continuously through techniques like "expand, shuffle, merge cardinality " without disrupting the gradient path.

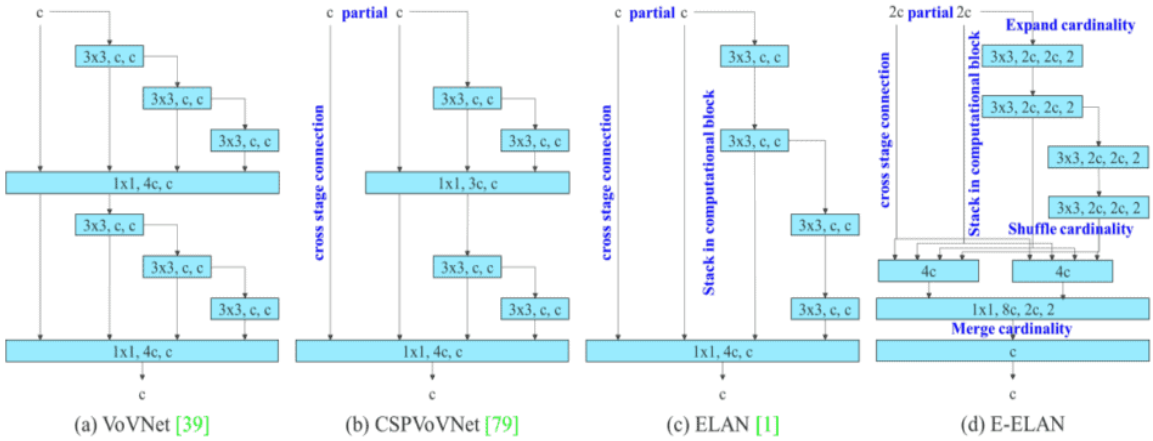


Fig2: E-ELAN Network

- Model Scaling for Concatenation based Models -Model Scaling, in YOLOv7 involves adjusting attributes of the model to create variants that cater to application requirements effectively.

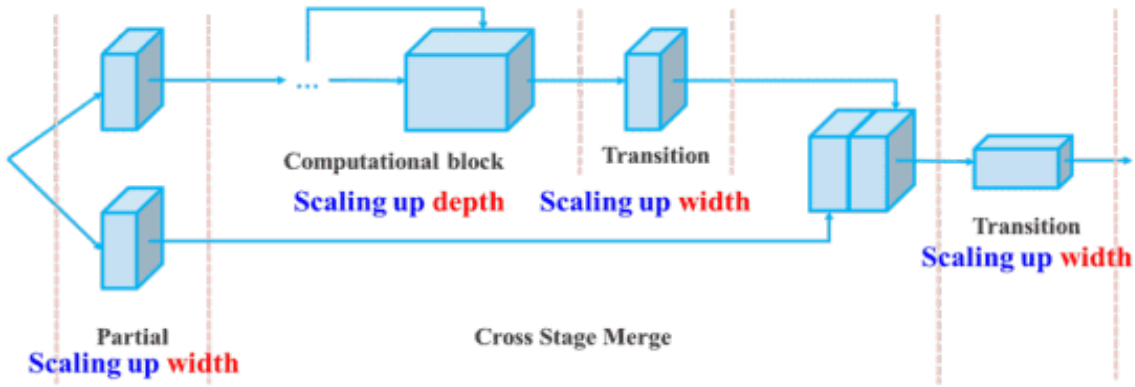


Fig3: YOLOv7 compound scaling

- Trainable Bag of Freebies- Bag of Freebies (BoF) are methods employed to improve a model's performance without increasing training costs.

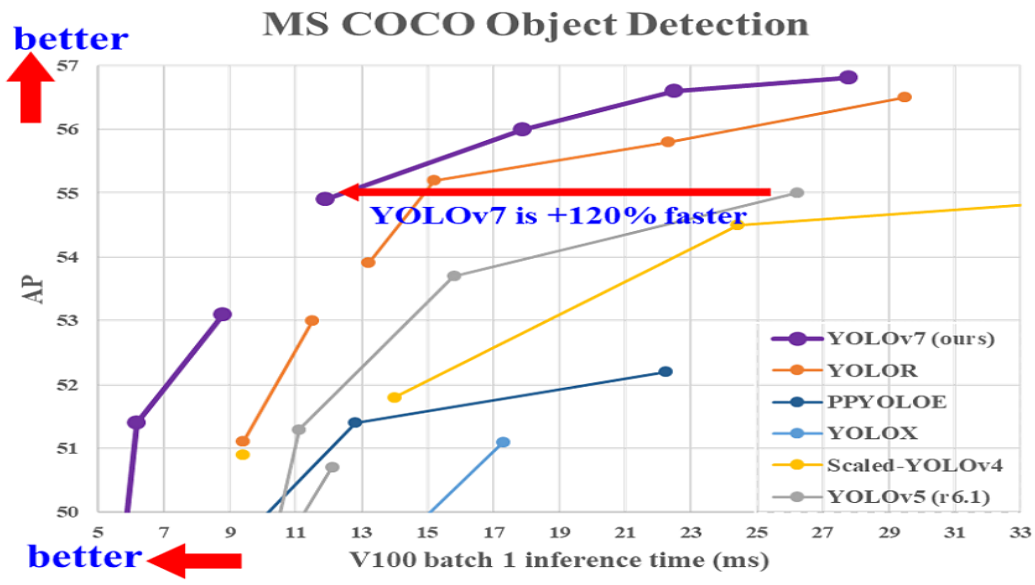


Fig4: YOLOv7 Exp Results

Based on the graph (Fig3) it is evident that when YOLOv7 applied on MS COCO Object Detection dataset it stands out in terms of both speed and accuracy compared to other YOLO object detectors. YOLOv7 surpasses all real time object detectors in terms of speed and accuracy. It achieves a range of FPS (Frames Per Second) from 5, to 165 and an mAP (Average Precision), between 51.4% and 56.8%. YOLOv7 has managed to reduce the number of parameters and computation by 40% while still improving performance, which's quite an accomplishment.

The model architecture of YOLOv8

YOLOv8 is the one of family of yolo models that was recently released in Jan 10th 2023 by ultralytics. Yolov8 performs better when compared to other yolo versions in terms of speed and accuracy by introducing some new features in the architecture.

- Yolov8 can be used for 3 different computer vision tasks.
    1. Object detection
    2. Image Classification
    3. Segmentation
- Models Available in YOLOv8 are

YOLOv8n	YOLOv8s	YOLOv8m	YOLOv8l	YOLOv8x

YOLOv8 Nano is the fastest and smallest, while YOLOv8 Extra Large (YOLOv8x) is the

most accurate yet the slowest among them.

The YOLOv8 architecture is based on a network, which can be split into two main components: the backbone and the head. The backbone consists of 53 layers and is a modified version of the CSPDarknet53 architecture. It incorporates stage partial connections to enhance the exchange of information, between different layers. On the other hand, the head of YOLOv8 consists of convolutional layers followed by fully connected layers. These specific layers are responsible for making predictions, about bounding boxes, object scores and class probabilities for objects identified in an image.

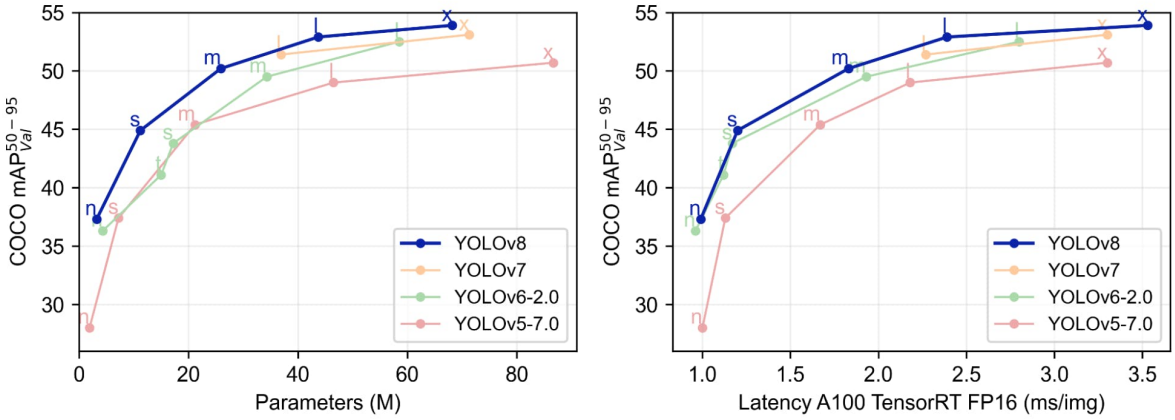


Fig4: YOLOv8 Exp Results

Based on the graph (Fig4) when YOLOv8 is trained and tested on MS COCO Object Detection dataset it performs better compared to other YOLO models. The YOLOv8m model was able to achieve a mAP of 50.2%, on the COCO dataset while the larger YOLOv8x model achieved an accuracy of 53.9% despite having, than twice the number of parameters.

Key Features of YOLOv8 includes:

- Improved Accuracy
- Enhanced Speed
- Multiple Backbones
- Adaptive Training
- Advanced-Data Augmentation
- Customizable Architecture

MODEL TRAINING & EVALUATION

For this project I've opted to use the Google Colab notebook as my platform, for running custom YOLO models, YOLOv7 and YOLOv8. To carry out all the tests I'm using the Tesla T4 graphics processing unit (GPU) with 16GB of GPU memory, which is seamlessly

connected through Google Colab. Additionally the platform provides 12GB of random access memory (RAM) enabling training process.

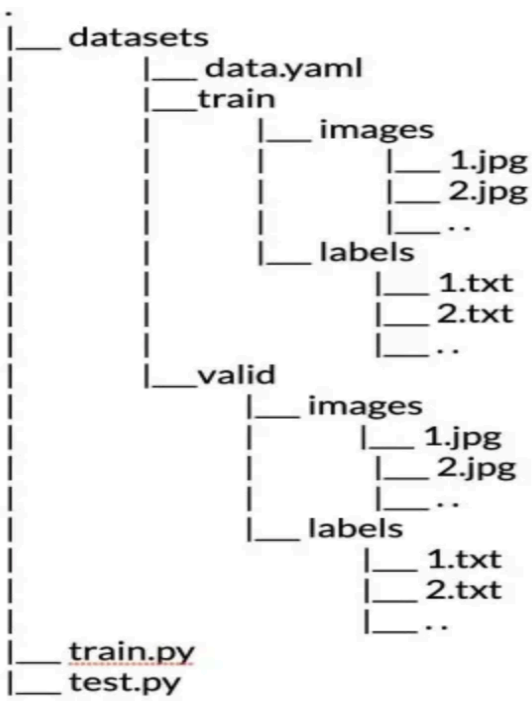


Fig5: Folder structure used by training yolo models.

The data.yaml file contains the path of the training, validation and testing datasets along with the target label classes.

To conduct the tests, I've divided the image dataset into three sets; a training set with 2,603 images a validation set with 114 images and a testing set with 82 images. Two different tests were performed to assess and compare the performances of YOLOv7 and YOLOv8 models. The primary metric used to evaluate these models' performance is Average Precision (mAP). This metric is commonly employed in studies focused on safety helmet detection to illustrate precision and recall trends across training models. The other evaluation metrics used for performance are precision (Pr), recall (Re) and F1 score. These metrics can be expressed mathematically as

$$Pr = \frac{TP}{TP + FP} \qquad Re = \frac{TP}{TP + FN}$$

$$AP = \int_0^1 Pr(Re)\ )\ dRe \qquad mAP = \frac{1}{n}\sum_{k=1}^{k=n} AP_k$$

Where TP is the True Positive, FP is the False Positive, FN is the False Negative, n=number of classes, AP is the average precision

The mean Average Precision (mAP) is obtained by comparing the detected bounding box with the ground truth bounding box. Precision measures the ratio of true positives to all positive predictions, while recall measures the ratio of true positives, to all actual objects.

RESULTS AND DISCUSSION

Model	Epochs	Training Time
YOLOv7	10	0.478 Hrs
YOLOv8	50	1.411 Hrs

TABLE I: Speed comparison of YOLOv7 and YOLOv8

While training these YOLO models in google colab, the compute resources provided by google colab are running out of computation when I tried to run for a greater number of epochs(That is because these YOLO models are heavily computation models). For that reason, I have trained YOLOv7 for only 10 epochs, each epoch running for 2.5 to 3 mins and for running 10 epochs it took 0.478 Hrs for training. On the other hand, YOLOv8 model has taken 1.411 hrs for training 50 epochs, each epoch has taken about 1.6 mins. From this we can conclude that YOLOv8 is performing better and fast in terms of speed when compared with YOLOv7.

For this project IoU (Intersection over Union) is set to a threshold of 0.5. If the IoU score is, above 0.5 we consider it as a good prediction. And the output will be considered true. However, if the IoU score is, below 0.5 the output will be considered false. IoU is a metric used for understanding how the predicted bounding box aligns with the actual bounding box in the given dataset.

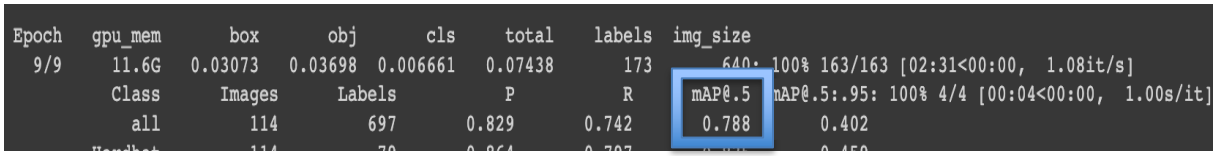


Fig. 6: mAP of YOLOv7 at epoch number 10

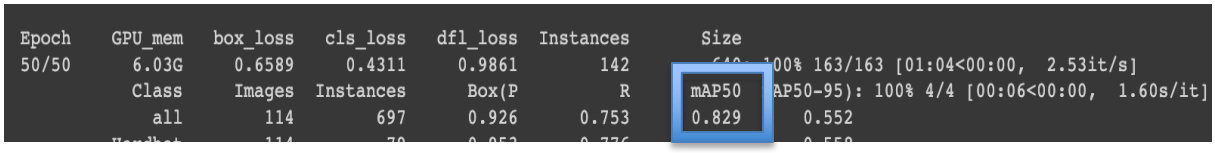


Fig. 7: mAP of YOLOv8 at epoch number 50

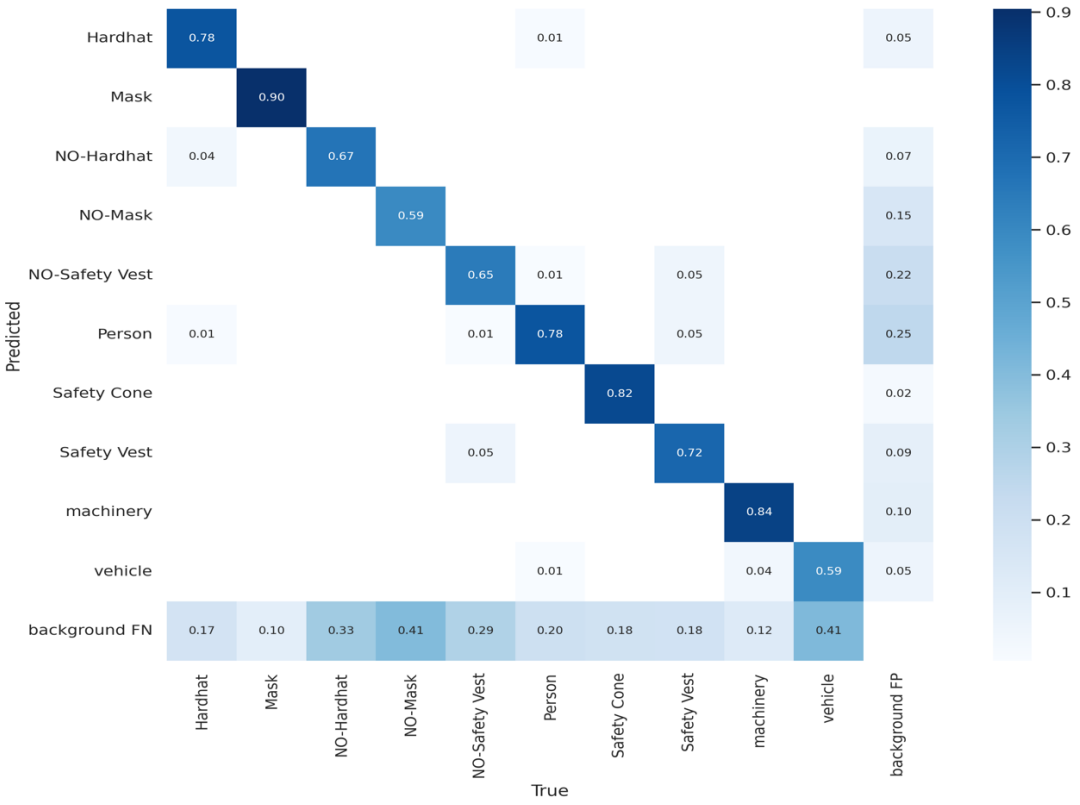


Fig. 8: Confusion matrix of the YOLOv7 model

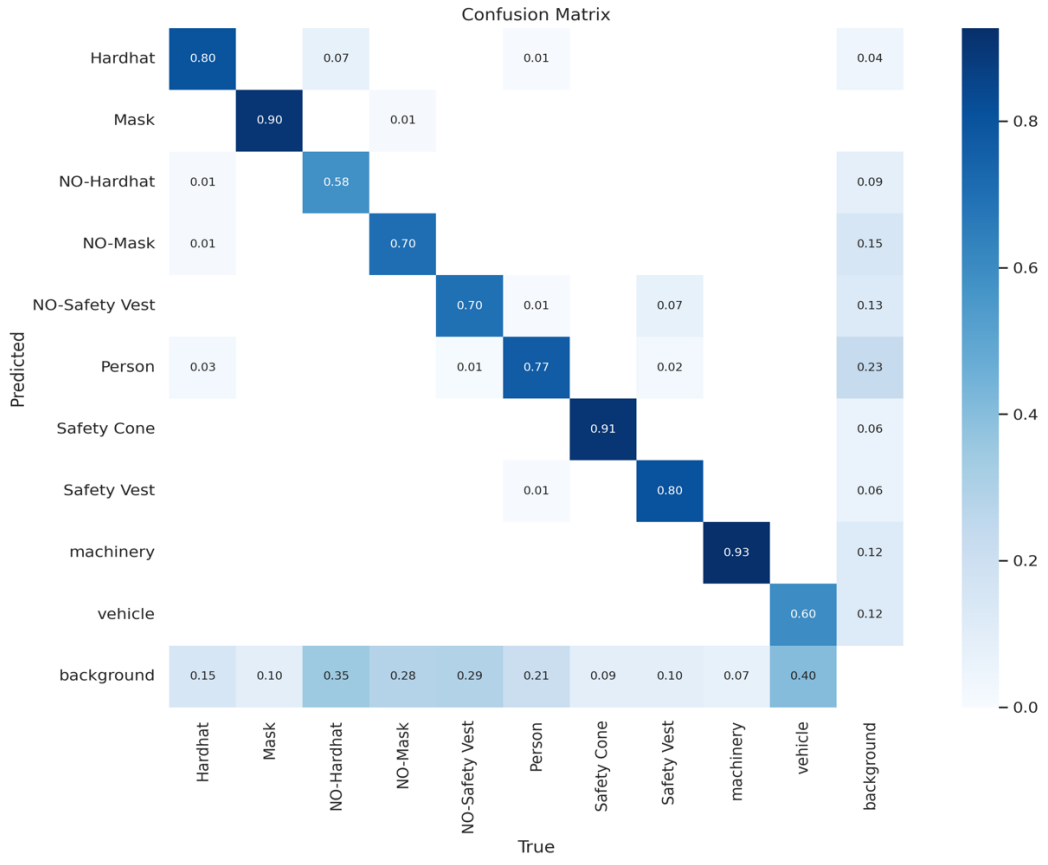


Fig. 9 : Confusion matrix of the YOLOv8 model

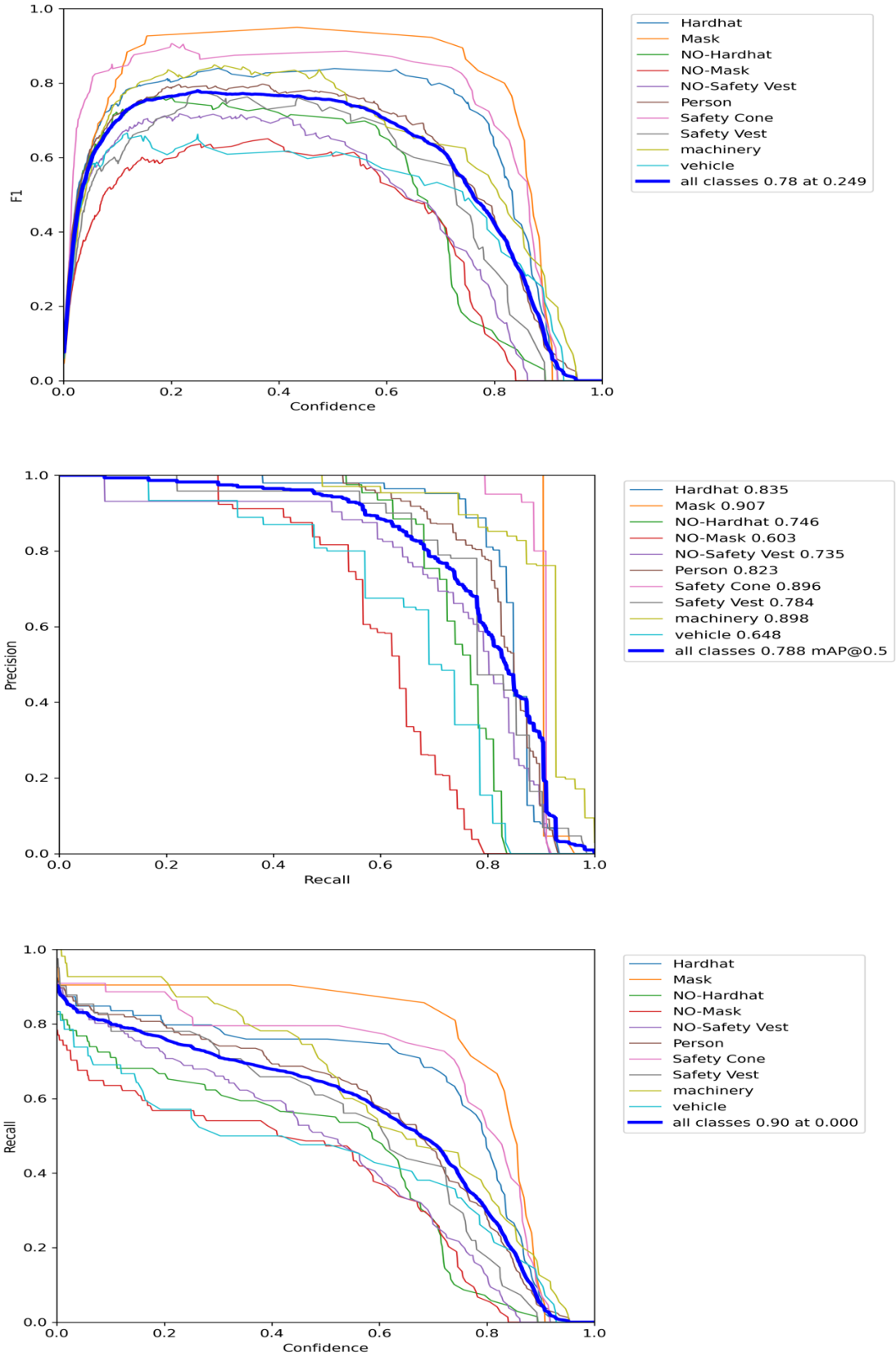


Fig. 10: Detection performance metrics of the YOLOv7 model

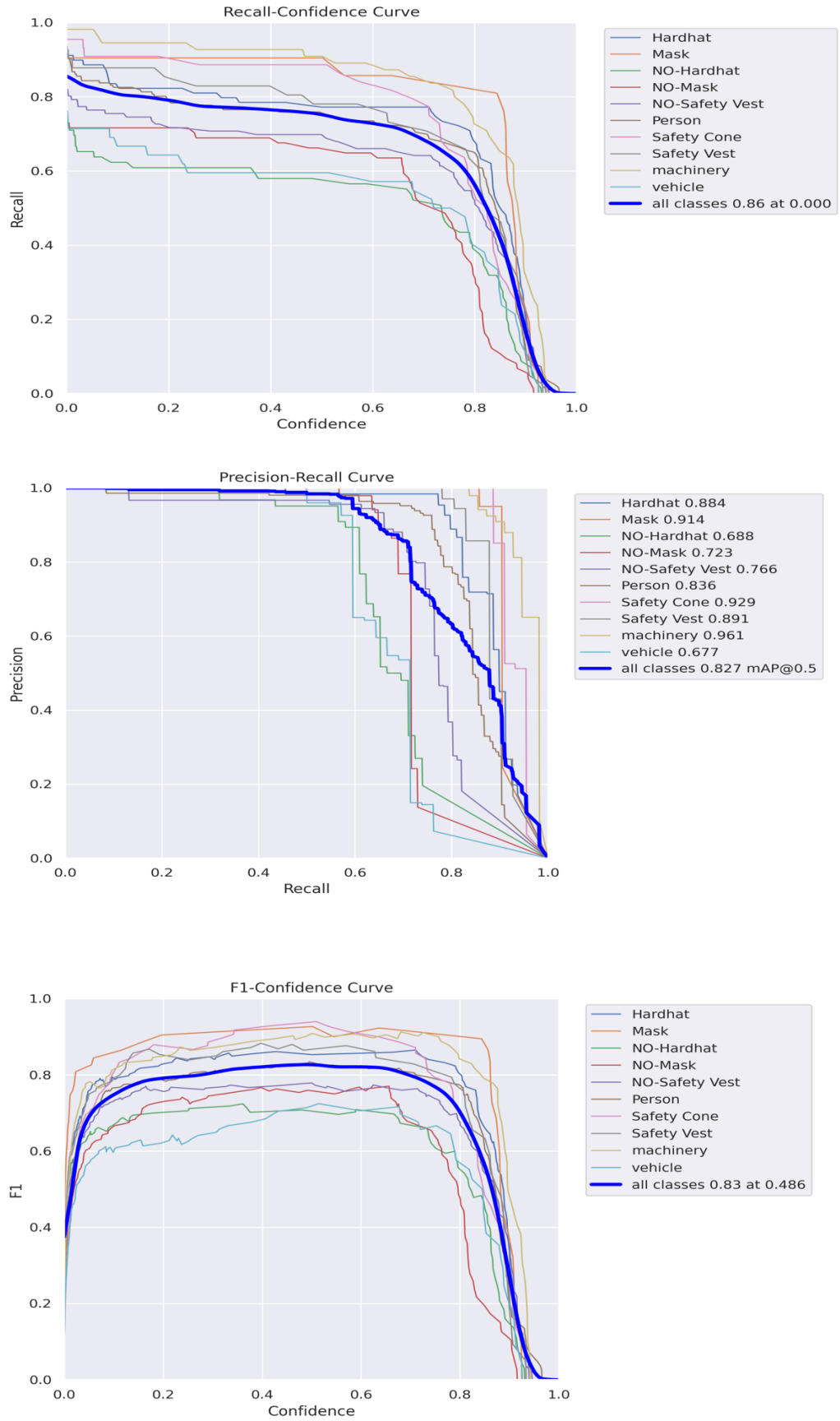
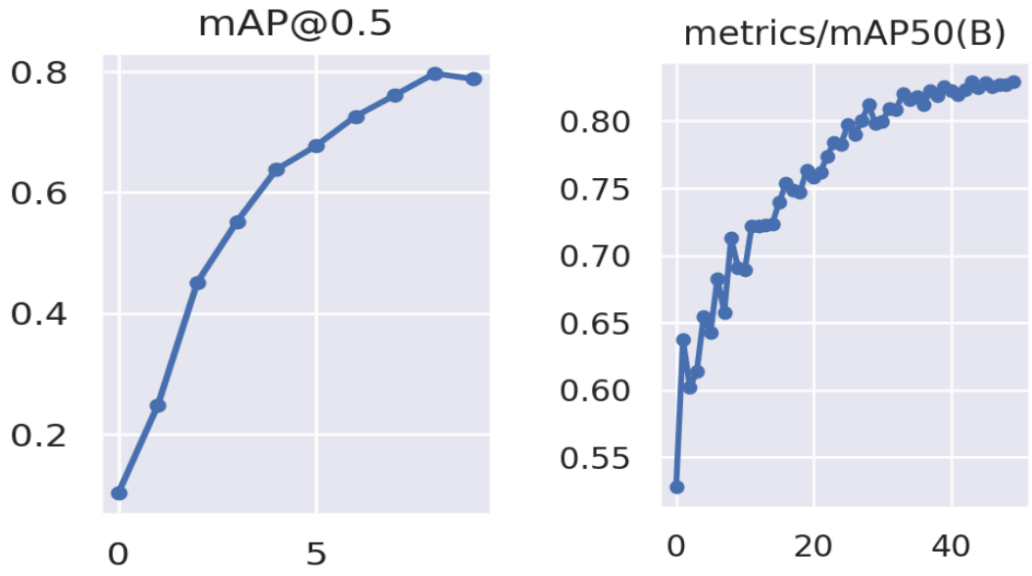


Fig. 11: Detection performance metrics of the YOLOv8 model



(a) YOLOv7 (b) YOLOv8

Fig. 12: mAP obtained from YOLOv7 and YOLOv8

The mAP obtained from YOLOv7 and YOLOv8 models are show in fig: 6 and fig:7 respectively, and their graphical representation is shown in fig 12. Among these two models YOLOv8 has been achieved the mAP of 82.9% and YOLOv7 achieved mAP of 78.8%. By comparing the accuracy of these two models YOLOv8 is performing better than YOLOv7 model. The confusion matrix of the YOLOv7 and YOLOv8 models are shown in figures 8 and 9 respectively, and detection performance metrics are shown in figures 10 and 11 respectively. After comparing the confusion matrices of these two models it is evident that YOLOv8 performs overall when it comes to detecting safety helmets.

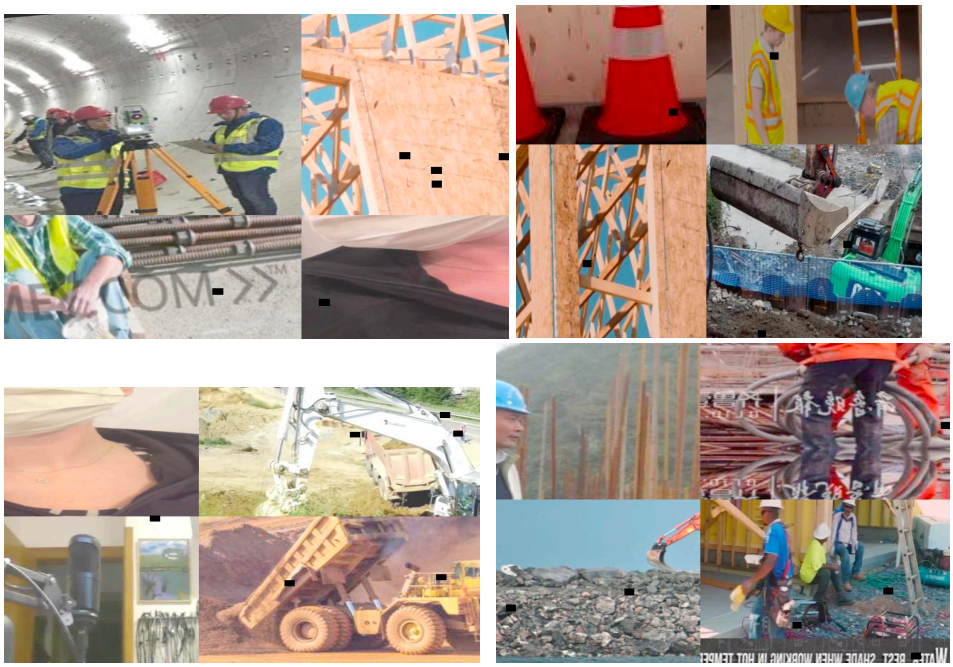
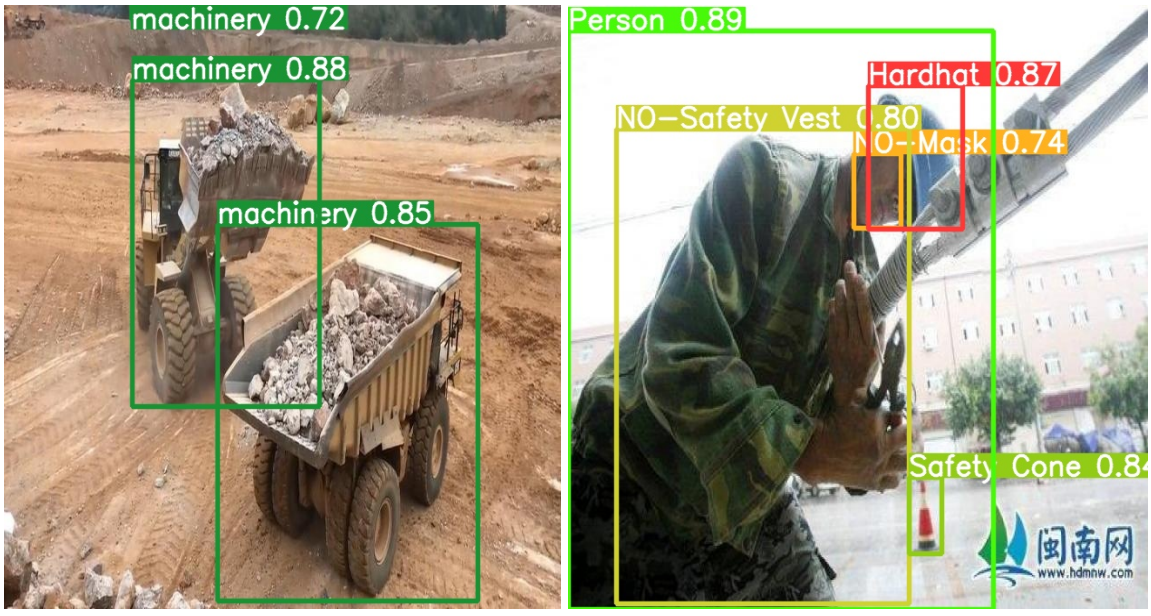


Fig. 13 Sample images from Training set

Fig. 14 Testing results of YOLOv7



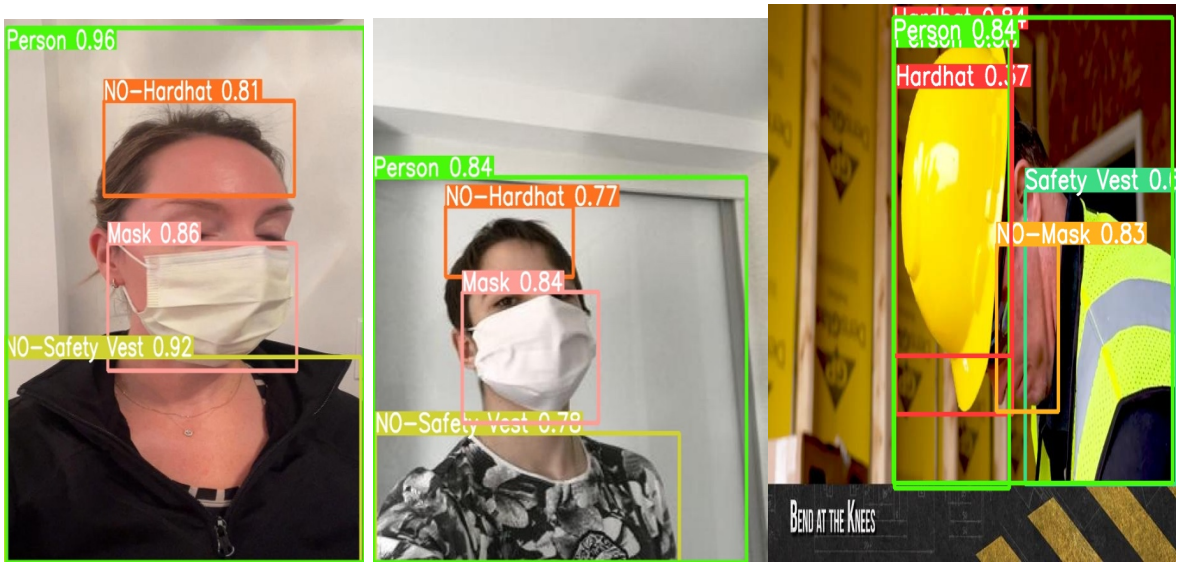


Fig. 15 Testing results of YOLOv8

YOLOv7 Tested on video results:
https://drive.google.com/file/d/136hc11QzNy7PGGltR3LsK520DmP9S1h0/view?usp=sharing

YOLOv8 Tested on video results:
https://drive.google.com/file/d/1BqTP14wyQlbfgeyoOrMttqFkbozhc177/view?usp=sharing

CONCLUSION

In this project I have evaluated the performance of YOLOv7 and YOLOv8 models in safety helmet detection. Both YOLOv7 and YOLOv8 are performing similar in terms of accuracy and YOLOv8 performs better in terms of speed. When we consider both speed and accuracy in safety helmet detection YOLOv8 is performing better than YOLOv7 model. The choice of epoch value and batch size has a considerable impact on the results obtained from these models.

FUTURE WORK

Future work for this project includes, when we have much time, we can develop an end-to-end safety helmet detection project by using these YOLOv7 and YOLOv8 models and deploy in real-time environments like construction sites, power stations and manufacturing plans. By Detecting whether the workers are wearing helmets or not in real-time helps in preventing the accidents and we can ensure that employees are following the safety regulations by using the necessary protective equipment. We can even create an immediate alert kind of system that

alerts both the worker and supervisor whenever the worker not wearing the helmet, safety vest. This allows for quick corrective actions as well. Over the time the system can generate more data on helmet compliance rates and then we can easily identify the patterns of helmet usage and suggest the areas for improvement and can reduce the downtime and keeping the productivity levels consistent.

REFERENCES

[1] "Mt-yolov6 pytorch object detection model." [Online]. Available: https://models.roboflow.com/object- detection/mt- yolov6
[2] "Yolov7 pytorch object detection model." [Online]. Available: https:/models.roboflow.com/object- detection/yolov7
[3] "Hard hat dataset," 2020. [Online]. Available: https://makeml.app/datasets/hard- hat-workers
[4] G. Munkhjargal, "Safety helmet wearing dataset," 2021. [Online]. Available: https://data.mendeley.com/datasets/9rcv8mm682/4
[5] "Yolov5 pytorch object detection model." [Online]. Available: https: //models.roboflow.com/object- detection/yolov5
[6] N. Kwak and D. Kim, "A study on detecting the safety helmet wearing using yolov5-s model and transfer learning," International Journal of Advanced Culture Technology, vol. 10, no. 1, pp. 302–309, 2022.
[7] A. Agrawal, J. Gans, and A. Goldfarb, Prediction machines: the simple economics of artificial intelligence. Harvard Business Press, 2018.
[8] A. Goldfarb, J. Gans, and A. Agrawal, The Economics of Artificial Intelligence: An Agenda. University of Chicago Press, 2019.
[9] J. Shaw, F. Rudzicz, T. Jamieson, A. Goldfarb et al., "Artificial intelli- gence and the implementation challenge," Journal of medical Internet research, vol. 21, no. 7, p. e13659, 2019.
[10] S. Dargan, M. Kumar, M. R. Ayyagari, and G. Kumar, "A survey of deep learning and its applications: a new paradigm to machine learning," Archives of Computational Methods in Engineering, vol. 27, no. 4, pp. 1071–1092, 2020.
[11] S. Gu, L. Ding, Y. Yang, and X. Chen, "A new deep learning method based on alexnet model and ssd model for tennis ball recognition," in 2017 IEEE 10th International Workshop on Computational Intelligence and Applications (IWCIA). IEEE, 2017, pp. 159–164.
[12] B. A. Lease, W. Wong, L. Gopal, and C. W. Chiong, "Pixel-level weed classification using evolutionary selection of local binary pattern in a stochastic optimised ensemble," SN Computer Science, vol. 1, no. 6, pp. 1–13, 2020.
[13] C. Zhen, X. Jingbo, B. Jun, and X. Min, "Feature extraction algorithm based on evolutionary depth learning," Computer Science Image, vol. 42, no. 11, p. 11, 2015.
[14] Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu, "Object detection with deep learning: A review," IEEE Transactions on Neural Networks and Learning Systems, vol. 30, no. 11, pp. 3212–3232, 2019.
[15] A. Dundar, J. Jin, B. Martini, and E. Culurciello, "Embedded streaming deep neural networks accelerator with applications," IEEE transactions on neural networks and learning systems, vol. 28, no. 7, pp. 1572–1583, 2016.