AI ASSISTED VISION BASED MODEL FOR IMPROVED SAFETY

PHASE I REPORT

Submitted by

Mukunthan S	**21011102065**
R Porselvi	**21011102077**
Vikas	**21011102118**

in partial fulfilment for the award of the degree of

BACHELOR OF TECHNOLOGY
IN

COMPUTER SCIENCE AND ENGINEERING

(INTERNET OF THINGS)

SHIV NADAR
UNIVERSITY
CHENNAI

DEPARTMENT OF COMPUTER SCIENCE

AND ENGINEERING

SCHOOL OF ENGINEERING

SHIV NADAR UNIVERSITY CHENNAI

NOVEMBER 2024

SHIV NADAR UNIVERSITY CHENNAI

BONAFIDE CERTIFICATE

Certified that this report titled "**AI ASSISTED VISION BASED MODEL FOR IMPROVED SAFETY**" is the bonafide work of **Mukunthan S** (Reg. No: **21011102065**), **R Porselvi** (Reg. No: **21011102077**), **Vikas** (Reg. No: **21011102118) who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.**

SIGNATURE	**SIGNATURE**
Dr. T. Nagarajan	Dr. Surya K
Professor	**Supervisor Designation**
Head of the Department	**Supervisor**
Department of Computer Science & Engineering	Department of Computer Science & Engineering
School of Engineering	School of Engineering
Shiv Nadar University Chennai,	Shiv Nadar University Chennai,
Kalavakkam -603110.	Kalavakkam -603110.

ABSTRACT

In the context of industrial and safety-critical environments, the implementation of effective safety protocols is paramount to safeguarding the well-being of individuals. Personal Protective Equipment (PPE) plays a pivotal role in mitigating occupational hazards, and ensuring its proper utilization is of utmost importance. This research project delves into the domain of automated detection of essential safety gear, namely safety helmets, masks, and vests, leveraging sophisticated deep learning techniques. The study places a particular emphasis on two state-of-the-art object detection algorithms, Faster R-CNN and YOLOv5, with the overarching goal of developing robust models that exhibit a high degree of accuracy in identifying and precisely localizing safety equipment within images. The initiation of the project involves a meticulous process of compiling a diverse and well-annotated dataset. This dataset comprises images featuring individuals from various occupational settings donning safety helmets, masks, and vests send to mail. Subsequently, both Faster R-CNN and YOLOv5 models undergo comprehensive training using this augmented dataset, wherein the specific characteristics and performance requirements of each algorithm are meticulously considered. The choice between Faster R-CNN and YOLOv5 is a nuanced decision, influenced by factors such as real-time processing demands, detection accuracy, and computational efficiency. The outcome of this research, beyond their theoretical significance, hold substantial practical implications for the domain of occupational safety. Key Words: Personal Protective Equipment, Occupational hazards, R-CNN, YOLO, object detection algorithms, automated detection.

ACKNOWLEDGEMENT

The successful completion of any task is incomplete and meaningless without giving any due credit to the people who made it possible without which the project would not have been successful and would have existed in theory.

First and foremost, we are grateful to **Dr. T. Nagarajan**, HOD, Department of Computer Science, Shiv Nadar University Chennai, and all other faculty members of our department for their constant guidance and support, constant motivation and sincere support and gratitude for this project work. We owe a lot of thanks to our supervisor, **Dr. Surya K**, Professor, Department of Computer Science, Shiv Nadar University Chennai for igniting and constantly motivating us and guiding us in the idea of a creatively and amazingly performed Major Project in undertaking this endeavour and challenge and also for being there whenever we needed her guidance or assistance.

We would also like to take this moment to show our thanks and gratitude to one and all, who indirectly or directly have given us their hand in this challenging task. We feel happy and joyful and content in expressing our vote of thanks to all those who have helped us and guided us in presenting this project work for our Major project. Last, but never least, we thank our well-wishers and parents for always being with us, in every sense and constantly supporting us in every possible sense whenever possible.

Mukunthan S	**R Porselvi**	**Vikas**
(21011102065)	**(21011102077)**	**(21011102118)**

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CHV Colour Helmet and Vest

CLI Command Line Interface

CMID COVID-19 Mask Image Dataset

CNN Convolutional Neural Network

DNN Deep Neural Networks

FPS Frames Per Second

GPU Graphics Processing Unit

GUI Graphical User Interface

HDD Helmet Detection Dataset

HTML Hypertext Markup Language

JSON JavaScript Object Notation

MAP Mean Average Precision

MIME Multipurpose Internet Mail Extensions

OpenCV Open Source Computer Vision Library

OS Operating System

PPE Personal Protective Equipment

PYPI Python Package Index

R-CNN Region-Based Convolutional Neural Network

RPN Region Proposal Network

SMTP Simple Mail Transfer Protocol

SSD Single-Shot Detector

UNIX UNiplexed Information Computing System

YOLO You Only Look Once

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

In the domain of occupational safety, the intersection of computer vision and deep learning has emerged as a powerful solution for enhancing workplace security. This research delves into the development and comparative analysis of two prominent deep learning models, Faster R-CNN and YOLOv5, to address the imperative task of real-time detection of safety helmets, masks, and vests in industrial environments. As pivotal components of personal protective equipment, these items play a crucial role in minimizing risks and preventing injuries.

The study aims to offer a nuanced understanding of the performance characteristics of each model, encompassing metrics such as precision, through an exploration of architecture components, region proposal mechanisms, and feature extraction techniques, the research aims to unravel the nuances of Faster R-CNN. Simultaneously, it scrutinizes the one-stage detection approach, anchor box clustering, and feature pyramid network of YOLOv5. Sent to mail by evaluating detection accuracy, speed, and resource utilization, the study aspires to provide valuable insights for practitioners seeking to implement robust safety gear detection systems.

This comparative analysis, rooted in real-world scenarios and diverse datasets, endeavours to contribute to the ongoing discourse on leveraging deep learning for workplace safety, offering recommendations for optimal model selection based on specific application requirements and laying the groundwork for future advancements in the field.

1.2 PROBLEM STATEMENT

In the construction industry, ensuring the safety of workers is paramount, yet traditional methods of monitoring safety compliance can be time-consuming and prone to human error. Recognizing this challenge, this project aims to leverage deep learning models for the automated detection of safety equipment such as helmets, masks, and vests in a construction site environment. The primary objective is to develop a robust and accurate system that can analyze real-time video feeds or images captured within the construction site and promptly identify instances where workers are not wearing the necessary safety gear. By automating this process, the project seeks to enhance safety protocols, reduce the risk of accidents, and streamline compliance monitoring procedures.

However, several challenges need to be addressed to achieve the project's objectives effectively. Firstly, the construction environment poses unique challenges such as varying lighting conditions, occlusions caused by equipment or debris, and the presence of multiple workers engaged in diverse tasks simultaneously. These factors can significantly impact the performance of the deep learning models and necessitate the development of sophisticated algorithms capable of handling such complexities. Additionally, the project must ensure the privacy and ethical considerations of workers by implementing mechanisms to anonymize or obfuscate personal identifiers while still accurately detecting safety violations. Overcoming these challenges will be crucial in developing a reliable and scalable solution that contributes to the overall safety and well-being of construction site workers.

1.3 OBJECTIVE

Develop a deep learning-based machine learning model capable of accurately detecting helmet, safety vest, and mask worn by workers in a construction site environment. Enhance safety measures in construction sites by automating the monitoring of safety equipment adherence among workers using advanced computer vision techniques. Implement real-time detection capabilities to swiftly identify instances of non-compliance with safety regulations, such as workers not wearing helmets, safety vests, or masks. Integrate the model with surveillance systems within the construction site to enable continuous monitoring of worker safety without human intervention.

Establish a reliable system for capturing images of non-compliant workers and generating automated alerts to supervisors via email, including the respective photographs for immediate action. Ensure the scalability and robustness of the model to handle varying environmental conditions, lighting conditions, and worker movements within the construction site. Optimize the model's performance to achieve high accuracy and minimal false positives, thereby reducing the chances of unnecessary alerts and ensuring efficient resource allocation. Conduct thorough testing and validation of the model using diverse datasets encompassing different scenarios and variations in safety equipment positioning and appearance. Provide an intuitive user interface for supervisors to view alerts, review detected images, and take necessary actions to enforce safety protocols among workers effectively.

CHAPTER 2

LITERATURE REVIEW

2.1 LITERATURE REVIEW

Fast Personal Protective Equipment Detection for Real Construction Sites Using Deep Learning Approaches *Zijian wang, Yimin Wu, Lichao Yang, Arjun Thirunavukarasu (2021)*

The existing deep learning-based Personal Protective Equipment (PPE) detectors can only detect limited types of PPE and their performance needs to be improved, particularly for their deployment on real construction sites. This paper introduces an approach to train and evaluate eight deep learning detectors, for real application purposes, based on You Only Look Once (YOLO) architectures for six classes, including helmets with four colours, person, and vest. Meanwhile, a dedicated high-quality dataset, CHV, consisting of 1330 images, is constructed by considering real construction site background, different gestures, varied angles and distances, and multi PPE classes. The comparison result among the eight models shows that YOLO v5x has the best map (86.55%), and YOLO v5s has the fastest speed (52 FPS) on GPU. The detection accuracy of helmet classes on blurred faces decreases by 7%, while there is no effect on other person and vest classes. And the proposed detectors trained on the CHV dataset have a superior performance compared to other deep learning approaches on the same datasets.

The novel multiclass CHV dataset is open for public use.

Deep learning for site safety: Real-time detection of personal protective equipment *Nipun D. Nath, Amir H. Behzadan, Stephanie G. Pall (2020)*

The authors propose a deep learning approach for real-time detection of personal protective equipment (PPE) on construction sites. The study focuses on enhancing safety measures by automating the monitoring of workers' adherence to PPE regulations. They employ convolutional neural networks (CNNs) to detect various types of PPE, including helmets, safety vests, and masks, from images captured on construction sites. The model's real-time capability allows for immediate detection of non-compliance, enabling prompt intervention to ensure worker safety. Through experimentation, the authors demonstrate the effectiveness of their approach in accurately detecting PPE items, highlighting its potential to improve safety practices in construction environments.

Furthermore, the paper addresses the challenges associated with deploying deep learning models in real-world construction settings, such as varying lighting conditions and occlusions. The authors discuss techniques to mitigate these challenges, including data augmentation and model optimization, to improve the robustness and reliability of the PPE detection system. By providing a comprehensive solution for real-time PPE detection using deep learning techniques, the study contributes to advancing safety protocols on construction sites and emphasizes the potential of artificial intelligence in enhancing workplace safety.

A Smart System for Personal Protective Equipment Detection in Industrial Environments Based on Deep Learning at the Edge *Gionatan Gall, Francesco Di Rienzo, Federic Garzelli, Pietro Ducange (2022)*

Real-time object detection is currently used to automate various tasks in industrial environments. One of the most important tasks is to improve the safety of workers by monitoring the correct use of Personal Protective Equipment (PPE) in dangerous areas. In this context, usually, a monitoring system analyzes the stream of videos from surveillance cameras to assess PPE usage in real time. When a worker not wearing the appropriate PPE is detected, an acoustic or visual alarm is triggered automatically to raise attention and awareness. The solutions proposed so far are mostly cloud-based systems: images from the site are continuously offloaded to the cloud for analysis. In this work, we propose a system for real-time PPE detection based on video streaming analysis and Deep Neural Network (DNN). We adopt the edge computing model in which the application for image analysis and classification is deployed on an embedded system installed in proximity of the camera and directly connected to it. We tested the system with five different pre-trained convolutional neural networks (CNNs), fine-tuned to detect different PPEs, namely helmets, vests, and gloves. In our experimental evaluation, we first compared the five CNNs in terms of classification performance and inference latency. Then, we deployed each CNN on the real system and evaluated the system's throughput regarding the number of video frames analyzed each second.

A Novel Implementation of an AI-Based Smart Construction Safety Inspection Protocol in the UAE *Mohammad Z Shanti, Chung-Suk Cho, Young-Ji Byon, Chan Yeob Yeun (2021)*

This paper likely presents an innovative approach to enhancing construction site safety through the implementation of artificial intelligence (AI) technology in the United Arab Emirates (UAE). It may detail the development and deployment of a smart safety inspection protocol that utilizes AI algorithms to analyze various aspects of construction site safety, such as personal protective equipment (PPE) compliance, hazard identification, and adherence to safety regulations. The implementation might involve the integration of sensors, cameras, and AI-powered systems to automate safety inspections, improve data collection efficiency, and minimize human error. Furthermore, the paper may discuss the benefits and challenges associated with deploying such a system in the context of UAE's construction industry, highlighting its potential to enhance worker safety, reduce accidents, and streamline regulatory compliance efforts.

Safety Helmet Detection Based on YOLOv5 *Fangbo Zhou, Huailin Zhao, Zhen Nie (2021)*

The author presents a method for detecting safety helmets worn by workers in industrial environments using the YOLOv5 deep learning framework. The study focuses on improving the accuracy and efficiency of safety helmet detection, crucial for ensuring worker safety. By leveraging YOLOv5's capabilities, the proposed model achieves real-time detection of safety helmets in diverse settings, including challenging lighting conditions and varied angles of view. The author demonstrates the effectiveness of the model through extensive experimentation and evaluation, showcasing its ability to accurately identify safety helmets while minimizing false positives. This research contributes to enhancing safety measures in industrial workplaces by providing a robust and efficient solution for safety helmet detection using state-of-the-art deep learning techniques.

2.2 SUMMARY OF LITERATURE REVIEW

S.NO	TITLE	YEAR	AUTHORS	CONCEPT
1	Fast Personal Protective Equipment Detection for Real Construction Sites Using Deep Learning Approaches.	2021	Zijian Wang, Yimin Wu, Lichao Yang, Arjun Thirunavukarasu	This paper presents an approach using YOLO architectures to train and evaluate eight deep learning detectors for detecting six classes of PPE, including helmets with four colors, person, and vest, addressing limitations in existing detectors for real construction site deployment.
2	Deep learning for site safety: Real-time detection of personal protective equipment	2020	Nipun D. Nath, Amir H. Behzadan, Stephanie G. Pall	The paper proposes a deep learning method for real-time detection of personal protective equipment (PPE) on construction sites. Using CNNs, it accurately detects helmets, safety vests, and masks, improving safety practices in construction environments.
3	A Smart System for Personal Protective Equipment Detection in Industrial Environments Based on Deep Learning at the Edge	2022	Gionatan Gall, Francesco Di Rienzo, Federic Garzelli, Pietro Ducange	The paper discusses the implementation of a real-time object detection system for monitoring PPE usage in industrial settings. It proposes deploying Deep Neural Networks (DNNs) on edge devices near cameras for efficient analysis.
4	A Novel Implementation of an AI-Based Smart Construction Safety Inspection Protocol in the UAE	2021	Mohammad Z Shanti, Chung-Suk Cho, Young-Ji Byon, Chan Yeob Yeun	The paper proposes an AI-driven safety inspection protocol for construction sites in the UAE, likely integrating sensors, cameras, and AI algorithms to automate inspections and improve safety compliance this innovative approach.
5	Safety Helmet Detection Based on YOLOv5	2021	Fangbo Zhou, Huailin Zhao, Zhen Nie	The author introduces a method employing YOLOv5 for real-time safety helmet detection in industrial settings, emphasizing accuracy and efficiency. Through comprehensive testing, the model proves effective in identifying helmets amidst diverse conditions, offering a valuable contribution to industrial safety with its robust deep learning approach.

Table 2.1: Summary of literature survey

CHAPTER 3

SYSTEM ANALYSIS

3.1 PROBLEM DEFINITION

Ensuring the safety of workers in construction sites is paramount to prevent accidents and injuries. Despite strict regulations mandating the use of safety equipment such as helmets, safety vests, and masks, compliance can often be challenging to monitor manually due to the large scale of construction sites and the constant movement of workers. This project aims to address this challenge by leveraging machine learning techniques, specifically deep learning, to develop a system capable of automatically detecting whether workers are wearing the required safety equipment.

The proposed system will utilize deep learning algorithms trained on a dataset consisting of images captured from various vantage points within a construction site. These images will encompass diverse lighting conditions, angles, and occlusions to ensure robustness in real-world scenarios. The model will be trained to recognize and classify individuals based on whether they are wearing a helmet, safety vest, and mask accurately.

Upon detecting a worker without the necessary safety equipment, the system will promptly capture the individual's image and timestamp. Subsequently, an automated email notification containing the image and relevant details will be dispatched to the designated supervisor. This notification will serve as a real-time alert, enabling the supervisor to take immediate action by reminding the worker to wear the required safety equipment or providing necessary assistance. By automating this process, the project aims to enhance safety protocols in construction sites, reducing the risk of accidents and promoting a safer working environment for all personnel involved.

3.2 EXISTING SYSTEM

Existing systems for safety helmet, mask, and vest detection often leverage deep learning technologies to enhance workplace safety and compliance. These systems typically employ pre-trained convolutional neural network (CNN) architectures such as Faster R-CNN, YOLO, or SSD, which have demonstrated effectiveness in object detection tasks. The training process involves using annotated datasets containing images of individuals wearing safety equipment and background images without the specified items. safety helmets, masks, and vests in images or video streams. Deployment of these systems can be integrated into existing surveillance infrastructure, enabling real-time monitoring and alerting when safety violations are detected. Continuous improvement through model retraining ensures adaptability to changing environments and evolving safety standards.

3.2.1 DISADVANTAGES

• Deep learning models, especially complex ones used for object detection, often require significant computational resources during both training and inference

• Addressing data bias is crucial for ensuring the model's effectiveness across different contexts.

• Annotating a large dataset with bounding boxes for helmets, masks, and vests can be time-consuming and expensive.

3.3 PROPOSED SYSTEM

The proposed system utilizes a webcam-based approach for real-time monitoring of safety compliance in industrial environments. Employing computer vision techniques, the system leverages the webcam feed to detect and identify essential safety gear such as helmets, masks, and vests worn by individuals in the monitored space. The system integrates state-of-the-art deep learning algorithms, specifically Faster R-CNN or YOLOv5, for accurate and efficient object detection. By continuously analysing the live webcam feed, the system provides instant on the presence and correct usage of safety equipment, contributing to the enhancement of safety protocols. This technology aims to automate safety monitoring processes, ensuring a proactive and responsive approach to maintaining a secure working environment.

3.3.1 ADVANTAGES

• Real-time Detection and Monitoring: By utilizing a webcam feed, the system can continuously monitor the industrial environment in real-time, allowing for prompt detection of safety violations as they occur.

• Accurate Object Detection: Integration of state-of-the-art deep learning algorithms such as Faster R-CNN or YOLOv5 ensures accurate and efficient detection of safety gear, including helmets, masks, and vests.

• Automation and Efficiency: The technology automates safety monitoring processes, eliminating the need for manual inspection and reducing the burden on human resources.

• Proactive Safety Measures: With instant detection capabilities, the system enables a proactive approach to safety management. By providing immediate alerts on the presence and correct usage of safety equipment, potential hazards can be addressed before they escalate, fostering a safer working environment for all personnel.

Responsive Intervention: In the event of safety violations, the system facilitates swift intervention by alerting supervisors or safety personnel in real-time. This responsiveness enables timely corrective action, such as issuing warnings or providing additional training, to mitigate risks and ensure adherence to safety protocols.

• Cost-effective Solution: Utilizing existing webcam infrastructure minimizes the need for additional hardware investments, making the system a cost-effective solution for safety monitoring in industrial settings.

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 REQUIREMENT ANALYSIS

Requirements are a feature of a system or description of something that the system is capable of doing in order to fulfil the system's purpose. It provides the appropriate mechanism for understanding what the customer wants, analysing the needs assessing feasibility, negotiating a reasonable solution, specifying the solution unambiguously, validating the specification and managing the requirements as they are translated into an operational system.

4.1.1 PYTHON

• Python is a dynamic, high level, free open source and interpreted programming language. It supports object-oriented programming as well as procedural oriented programming. In Python, we don't need to declare the type of variable because it is a dynamically typed language.

For example, x=10. Here, x can be anything such as String, int, etc. Python is an interpreted, object-oriented programming language similar to PERL, that has gained popularity because of its clear syntax and readability. Python is said to be relatively easy to learn and portable, meaning its statements can be interpreted in a number operating systems, including UNIX-based systems, Mac OS, MSDOS, OS/2, and various versions of Microsoft Windows 98. Python was created by Guido van Rossum, a former resident of the Netherlands, whose favourite comedy group at the time was Monty Python's Flying Circus. The source code is freely available and open for modification and reuse. Python has a significant number of users. **Features in Python**

There are many features in Python, some of which are discussed below

• Easy to code

• Free and Open Source

• Object-Oriented Language

• GUI Programming Support

• High-Level Language

• Extensible feature

• Python is Portable language

• Python is Integrated language

• Interpreted Language

4.1.2 ANACONDA

Anaconda distribution comes with over 250 packages automatically installed, and over 7,500 additional open-source packages can be installed from PYPI as well as the anaconda package and virtual environment manager. It also includes a GUI, Anaconda Navigator, as a graphical alternative to the command line interface (CLI).

The big difference between anaconda and the pip package manager is in how package dependencies are managed, which is a significant challenge for Python data science and the reason anaconda exists.

When pip installs a package, it automatically installs any dependent Python packages without checking if these conflict with previously installed packages. It will install a package and any of its dependencies regardless of the state of the existing installation. Because of this, a user with a working installation of, for example, Google Tensorflow, can find that it stops working having used pip to install a different package that requires a different version of the dependent numpy library than the one used by Tensorflow. In some cases, the package may appear to work but produce different results in detail.

In contrast, anaconda analyses the current environment including everything currently installed, and, together with any version limitations specified (e.g. the user may wish to have Tensorflow version 2,0 or higher), works out how to install a compatible set of dependencies, and shows a warning if this cannot be done.

Opensource packages can be individually installed from the Anaconda repository, Anaconda Cloud (anaconda.org), or the user's own private repository or mirror, using the anaconda install command. Anaconda, Inc. compiles and builds the packages available in the Anaconda repository itself, and provides binaries for Windows 32/64 bit, Linux 64 bit and MacOS 64-bit. Anything available on PyPI may be installed into a anaconda environment using pip, and anaconda will keep track of what it has installed itself and what pip has installed.

Custom packages can be made using the anaconda build command, and can be shared with others by uploading them to Anaconda Cloud, PyPI or other repositories.

The default installation of Anaconda2 includes Python 2.7 and Anaconda3 includes Python 3.7. However, it is possible to create new environments that include any version of Python packaged with anaconda.

4.1.3 ANACONDA NAVIGATOR

Anaconda Navigator is a desktop Graphical user interface (GUI) included in

Anaconda distribution that allows users to launch applications and manage anaconda packages, environments and channels without using command-line commands. Navigator can search for packages on Anaconda Cloud or in a local Anaconda Repository, install them in an environment, run the packages and update them. It is available for Windows, macOS and Linux.

The following applications are available by default in Navigator:

- JupyterLab
- Jupyter Notebook
- QtConsole
- Spyder
- Glue
- Orange
- RStudia
- Visual Studio code

4.1.4 JUPYTER NOTEBOOK

Jupyter Notebook (formerly IPython Notebooks) is a web-based interative computational environment for creating Jupyter notebook documents. The ”notebook” term can colloquially make reference to many different entities, mainly the Jupyter web application, Jupyter Python web server, or Jupyter document format depending on context. A Jupyter Notebook document is a JSON document, following a versioned schema, containing an ordered list of input/output cells which can contain code, text (using Markdown), mathematics, plots and rich media, usually ending with the ”.ipynb” extension.

Jupyter Notebook can connect to many kernels to allow programming in different languages.

The Notebook interface was added to IPython in the 0.12 release (December 2011), renamed to Jupyter notebook in 2015 (IPython 4.0 – Jupyter 1.0). Jupyter Notebook is similar to the notebook interface of other programs such as Maple, Mathematica, and SageMath, a computational interface style that originated with Mathematica in the 1980s. According to The Atlantic, Jupyter interest overtook the popularity of the Mathematica notebook interface in early 2018.

4.2 RESOURCE REQUIREMENTS

4.2.1 SOFTWARE REQUIREMENTS

Operating System	Windows 7 or later
Simulation Tool	Visual Studio Code
Documentation	MS-Office

Table 4.1: Software requirements

4.2.2 HARDWARE REQUIREMENTS

CPU type	I5 and Above
Ram size	4GB
Hard disk capacity	80 GB
Keyboard type	Internet keyboard
Monitor type	15 Inch colour monitor
CD -drive type	52xmax

Table 4.2: Hardware requirements

CHAPTER 5

SYSTEM DESIGN

5.1 ARCHITECTURE DIAGRAM

An architecture diagram is a visual representation that illustrates the structure, components, and interactions of a system or application. It provides a high-level overview of the system's design, showcasing how different elements within the system are organized and how they communicate with each other. Architecture diagrams are commonly used in software development, network engineering, and system design to aid in understanding, communication, and decision-making processes.

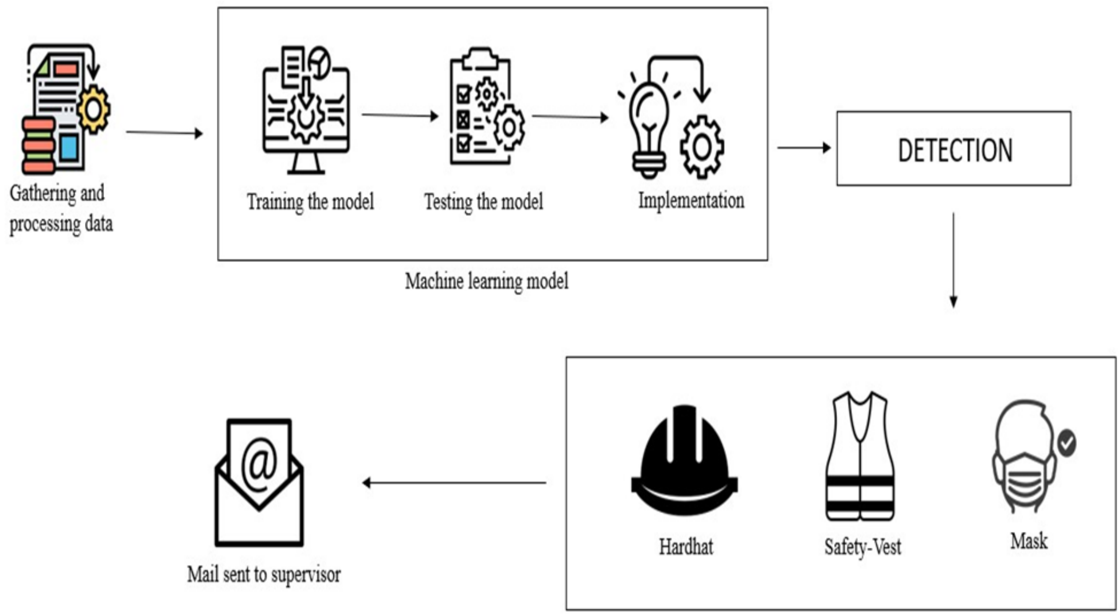

Figure 5.1: Architecture diagram

5.2 FLOW DIAGRAM

A flow diagram, also known as a flowchart, is a visual representation of a process or algorithm. It uses different shapes and arrows to illustrate the sequence of steps or actions involved in completing a task or achieving a goal. Flow diagrams are widely used in various fields, including software engineering, business management, and project planning, to help understand, document, and communicate complex processes in a clear and concise manner.

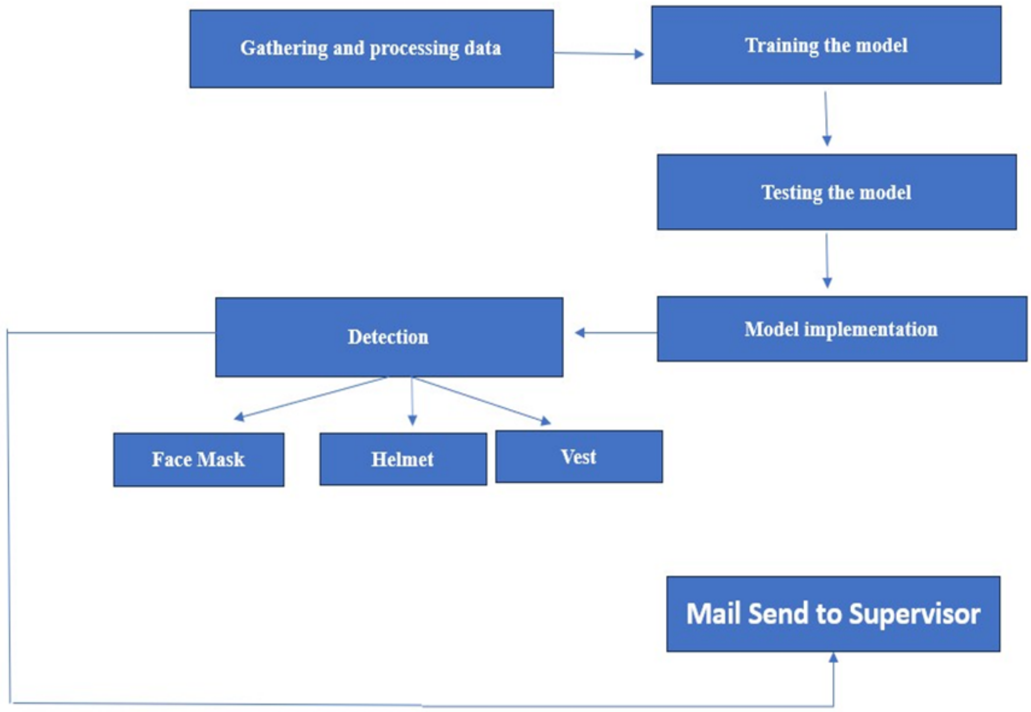

Figure 5.2: Flow diagram

CHAPTER 6

FUNCTIONAL DESIGN

6.1 MODULES

6.1.1 MODULE LISTING

1. Data collection
2. Training the model
3. Testing the model
4. Model implementation and detection
5. Sending mail

6.2 MODULE DESCRIPTION

6.2.1 DATA COLLECTION

This module serves as the foundational component in the development of a robust machine learning model aimed at enhancing safety protocols in construction sites. This module is pivotal in procuring a diverse and comprehensive dataset essential for training deep learning algorithms to accurately identify and classify safety equipment such as helmets, safety vests, and masks worn by workers. Through a systematic approach, the module gathers annotated images and corresponding metadata, ensuring representation across various environmental conditions, worker demographics, and equipment variations commonly encountered on construction sites.

Utilizing a combination of manual annotation and automated data extraction techniques, the Data Collection Module meticulously curates a high-quality dataset essential for model training. Leveraging state-of-the-art image processing technologies, it captures images from multiple angles, under varying lighting conditions, and amidst different work scenarios. Furthermore, the module integrates real-time surveillance systems to continuously update the dataset, ensuring its relevance and adaptability to evolving site conditions. By prioritizing data diversity and integrity, this module lays the groundwork for building a machine learning model capable of reliably detecting compliance with safety regulations, thereby mitigating workplace hazards.

In addition to facilitating model development, the Data Collection Module plays a pivotal role in proactive safety enforcement. Upon detection of individuals without proper safety equipment, the module triggers an alert mechanism that promptly notifies site supervisors via email. These alerts include timestamped images of the non-compliant individuals, empowering supervisors to take immediate corrective action by alerting the workers and enforcing safety protocols. This real-time feedback loop not only enhances safety compliance but also fosters a culture of accountability and vigilance, ultimately fostering a safer working environment for all personnel involved in construction projects.

6.2.2 TRAINING THE MODEL

Training a machine learning model to detect safety equipment like vests, masks, and helmets in a construction site using YOLOv5 and Faster R-CNN involves a meticulous process of data collection, annotation, and model optimization. Initially, a diverse dataset comprising annotated images of workers wearing various combinations of safety gear is gathered, capturing different lighting conditions, angles, and environmental contexts typical of a construction site. These images serve as the foundation for training the model, allowing it to learn the visual features associated with vests, masks, and helmets.

Next, the dataset is meticulously annotated, with each image labeled to indicate the presence and precise location of safety equipment. This annotated dataset is crucial for supervised learning, providing the model with ground truth annotations to guide its training process. YOLOv5 and Faster R-CNN algorithms are then employed to train the model, utilizing their respective architectures to learn to detect and localize safety gear within images.

During training, the model iteratively adjusts its internal parameters based on a feedback loop of forward and backward passes through the network, optimizing its ability to accurately classify and locate safety equipment in diverse construction site scenarios. Through this iterative process, the model gradually improves its detection performance, achieving higher precision and recall rates. Once the training process is complete, the model is fine-tuned and evaluated using validation data to ensure its robustness and generalization capability across different construction site environments.

6.2.3 TESTING THE MODEL

To evaluate the efficacy of our machine learning model in detecting safety equipment within a construction site, we conduct rigorous testing utilizing a diverse dataset of images representative of real-world scenarios. Leveraging the YOLOv5 and Faster R-CNN algorithms, our testing process involves feeding the model with annotated images containing various configurations of workers, environmental conditions, and equipment placements. Through this comprehensive dataset, we assess the model's ability to accurately identify and localize safety gear, including vests, masks, and helmets, amidst complex backgrounds and occlusions common in construction sites. We meticulously analyze the model's performance metrics, such as precision, recall, and mAP (mean Average Precision), to quantify its detection capabilities across different classes of safety equipment. Additionally, we employ techniques such as data augmentation and cross-validation to enhance the model's robustness and generalization capabilities. By systematically evaluating the model's performance against ground truth annotations, we ensure its reliability in real-time deployment, thus bolstering safety protocols and mitigating risks within construction site environments.

6.2.4 MODEL IMPLEMENTATION AND DETECTION

In real-time deployment, our machine learning model, trained with a rich dataset using YOLOv5 and Faster R-CNN algorithms for detecting safety equipment in construction sites, seamlessly integrates with OpenCV and anchor boxes to achieve efficient and accurate detection. Leveraging OpenCV's highperformance computer vision capabilities, the model analyzes live video streams from surveillance cameras placed strategically across the construction site. By leveraging anchor boxes, predefined bounding boxes of various aspect ratios and scales are used to guide the model's attention to potential objects of interest, streamlining the detection process and improving computational efficiency. As each frame is processed in real-time, the model rapidly identifies and localizes helmets, vests, and masks with precision, ensuring swift response to safety violations. Through continuous optimization and fine-tuning, our integrated solution delivers robust and reliable performance, enhancing safety measures and safeguarding workers in dynamic construction environments.

6.2.5 SENDING EMAIL

In our construction site safety monitoring system, we employ Python's MIME package to facilitate automated email notifications to the site supervisor whenever a worker is detected without the essential safety equipment of helmet, mask, or vest. Upon detection of a safety violation by our machine learning model, the system triggers a predefined action to construct an email alert using the MIME package. This email contains pertinent details, including a highresolution image capturing the individual in violation, providing visual evidence for immediate action. The MIME package allows us to embed images seamlessly within the email body, ensuring clear communication and facilitating swift decision-making by the supervisor. By automating this notification process, we ensure timely intervention and enforcement of safety protocols, fostering a proactive approach to worker safety within the construction site environment.

CHAPTER 7

METHODOLOGY

7.1 LIBRARY USED

7.1.1 TORCH

The Torch library, a popular open-source machine learning framework, has gained significant traction in the deep learning community due to its flexibility and efficiency. Built on the Lua programming language, Torch provides a wide range of tools and algorithms for building neural networks, making it a favored choice for researchers and practitioners alike. With its dynamic computational graph feature, Torch enables seamless model construction and experimentation, facilitating rapid prototyping and development. Additionally, Torch offers support for GPU acceleration, leveraging the computational power of graphics processing units to expedite training and inference processes, crucial for handling large-scale datasets common in deep learning tasks.

In our project focused on construction safety, leveraging Torch for developing a deep learning model to detect helmet, mask, and vest usage proves highly advantageous. By utilizing Torch's robust functionalities for building neural networks, you can design complex architectures tailored to the intricacies of your detection task. With Torch's support for GPU acceleration, you can efficiently train your model on extensive datasets, enhancing its ability to generalize and accurately identify safety equipment across various environmental conditions. Furthermore, Torch's flexibility allows for seamless integration with other Python libraries commonly used in machine learning workflows, streamlining data preprocessing, model evaluation, and deployment processes within your project pipeline. Overall, harnessing the capabilities of the Torch library empowers you to develop a sophisticated deep learning solution capable of enhancing construction safety standards effectively.

7.1.2 OPEN CV

OpenCV (Open Source Computer Vision Library) is a popular open-source computer vision and machine learning software library, primarily aimed at realtime computer vision tasks. In Python, OpenCV provides a powerful set of tools for image processing, manipulation, and analysis. Leveraging its extensive functionality, developers can create sophisticated applications for various domains, including object detection, facial recognition, and augmented reality.

In our project focused on construction safety, OpenCV plays a crucial role in enabling real-time detection of safety equipment such as helmets, masks, and vests. By integrating OpenCV with deep learning techniques, you've built a robust model capable of accurately identifying these essential safety items from live video feeds. The library facilitates various tasks within your project pipeline, including image preprocessing, feature extraction, and object detection.

Through OpenCV's capabilities, your deep learning model can efficiently process incoming video streams, analyze each frame, and detect the presence or absence of helmets, masks, and vests worn by construction workers. Real-time detection is critical for ensuring immediate intervention in case of safety violations, thus enhancing on-site safety measures. The seamless integration of OpenCV with your deep learning model empowers construction site managers with a reliable tool for monitoring safety compliance and mitigating potential risks effectively.

7.1.3 SMTPLIB

The ‘smtplib‘ library in Python serves as a crucial tool for sending emails programmatically. It provides a simple and efficient way to connect to an SMTP (Simple Mail Transfer

Protocol) server and send emails. With ‘smtplib‘, developers can easily integrate email functionality into their Python applications, enabling tasks such as sending notifications, alerts, or reports automatically. The library supports various authentication methods and allows customization of email content, recipients, and attachments, making it versatile for a wide range of applications.

In our project focused on real-time construction safety using deep learning models to detect helmet, mask, and vest compliance, integrating ‘smtplib‘ enables automated notification to supervisors when safety protocols are violated. Upon detection of a safety breach by your deep learning model, such as an individual not wearing a helmet, mask, or vest, your application can trigger an email alert using ‘smtplib‘ to notify the relevant supervisor or safety personnel. This immediate notification empowers supervisors to take swift action to address safety concerns on-site, potentially preventing accidents or injuries.

By leveraging ‘smtplib‘ within your construction safety project, you enhance the effectiveness of your safety monitoring system. Integrating email alerts ensures that safety violations are promptly brought to the attention of supervisors, facilitating rapid response and intervention. This proactive approach to safety management, enabled by ‘smtplib‘, contributes to a safer work environment for construction workers by reinforcing adherence to safety protocols in real-time. Furthermore, the flexibility of ‘smtplib‘ allows for scalability, enabling your application to accommodate future enhancements or additional features to further enhance construction site safety.

7.1.4 MIME

The Mime library in Python serves as a powerful tool for handling MIME (Multipurpose Internet Mail Extensions) messages, allowing developers to create, parse, and manipulate email messages effortlessly. By utilizing this library, developers can construct emails with various content types, including text, HTML, images, and attachments, ensuring flexibility and compatibility across different email clients. Additionally, the Mime library facilitates the parsing of incoming emails, enabling efficient extraction of content and attachments for further processing.

In our project, which focuses on real-time construction safety using a deep learning model to detect helmet, mask, and vest adherence, integrating the Mime library can enhance communication efficiency and safety enforcement mechanisms. As the deep learning model continuously analyzes the construction site footage, detecting any violations of safety protocols, the Mime library can be employed to dynamically generate and send email notifications to the project supervisor in case of non-compliance. These notifications can include detailed reports, images, or video snippets illustrating the observed safety breaches, empowering supervisors to take immediate corrective actions.

Furthermore, the Mime library's capabilities can extend beyond simple email notifications by allowing customization and automation of the notification process. For instance, you can design templates for different types of safety violations, dynamically inserting relevant information such as the type of violation detected, timestamp, and location. Moreover, by integrating with other Python libraries or frameworks, such as OpenCV for image processing or Flask for web applications, you can create a comprehensive safety monitoring system that not only detects violations but also provides actionable insights and real-time alerts to ensure the highest level of safety compliance on construction sites.

7.2 TECHNOLOGY USED

7.2.1 DEEP LEARNING

Deep learning is a subset of artificial intelligence (AI) that mimics the structure and function of the human brain to process and understand complex patterns in data. It utilizes neural networks, which are interconnected layers of nodes that perform computations on input data, gradually learning to identify and extract meaningful features. Deep learning excels in tasks such as image recognition, natural language processing, and speech recognition, thanks to its ability to automatically discover hierarchical representations of data. In your project, deep learning is employed to enhance construction safety by developing a model capable of detecting whether workers are wearing helmets, masks, and vests, crucial safety gear to prevent injuries and ensure compliance with safety regulations.

The deep learning model utilizes convolutional neural networks (CNNs), a specialized type of neural network particularly effective for image-related tasks. CNNs excel at automatically learning spatial hierarchies of features from images, enabling them to detect patterns at various scales and complexities. By training the model on a vast dataset comprising images of workers wearing and not wearing safety gear in diverse scenarios, the deep learning model learns to recognize the visual features associated with helmets, masks, and vests. Through a process called backpropagation, the model adjusts its internal parameters during training to minimize the difference between its predictions and the ground truth labels, thereby improving its accuracy and performance over time.

Once trained, your deep learning model can be deployed in real-time applications to monitor construction sites continuously. Integrated with surveillance cameras or drones, the model analyzes live video feeds to detect instances where workers are not wearing the required safety gear. Upon detection of a safety violation, the model triggers alerts or notifications, enabling swift intervention to ensure adherence to safety protocols. By leveraging the power of deep learning, your project significantly enhances construction site safety by automating the monitoring process and mitigating the risk of accidents or injuries resulting from non-compliance with safety regulations.

7.2.2 FASTER R-CNN

Faster R-CNN (Region-based Convolutional Neural Network) is a state-ofthe-art object detection algorithm in the field of computer vision. It builds upon the R-CNN framework by introducing Region Proposal Networks (RPNs), which significantly improve the speed and accuracy of object detection tasks. Unlike previous methods that required external region proposal methods, Faster R-CNN integrates the region proposal generation directly into the network, making it an end-to-end trainable architecture. By efficiently generating region proposals and classifying objects within those regions, Faster R-CNN achieves impressive performance in terms of both speed and accuracy.

In our construction safety project, leveraging Faster R-CNN for helmet, mask, and vest detection can substantially enhance the efficiency and reliability of your deep learning model. By utilizing its advanced architecture, Faster R-CNN enables precise localization and classification of safety gear within construction site imagery, even in complex and cluttered environments. This capability is crucial for ensuring the safety of workers by promptly identifying instances wheresafety protocols are not followed, such as missing or improperly worn safety equipment.

Moreover, integrating Faster R-CNN into your safety monitoring system enables real-time detection and response to safety violations. As the model continuously processes the construction site footage, it can rapidly detect instances of non-compliance with safety regulations regarding helmets, masks, and vests. Upon detection, the system can trigger immediate alerts or notifications to relevant personnel, allowing them to take swift corrective actions and mitigate potential safety hazards. Overall, by harnessing the power of Faster R-CNN, your project can achieve heightened levels of safety enforcement and protection for construction workers, ultimately fostering a safer working environment.

7.2.3 YOLOv5

YOLOv5 is a state-of-the-art object detection algorithm that stands for ”You Only Look Once,” and it represents the fifth iteration of the YOLO family. Developed by Ultralytics, YOLOv5 builds upon the success of its predecessors by introducing advancements in speed, accuracy, and versatility. This algorithm employs a single convolutional neural network (CNN) to simultaneously predict multiple bounding boxes and their corresponding class probabilities within an image. YOLOv5 achieves remarkable performance by leveraging a novel architecture that balances model complexity with computational efficiency, making it ideal for real-time applications.

In our project aimed at enhancing construction safety through deep learning-based detection of safety gear like helmets, masks, and vests, integrating YOLOv5 can revolutionize the detection capabilities. By training YOLOv5 on annotated datasets containing images of construction workers wearing safety equipment, the model can learn to accurately identify and localize these objects in real-time footage captured on construction sites. The speed and efficiency of YOLOv5 enable seamless integration into your safety monitoring system, allowing for rapid analysis of video streams to ensure compliance with safety protocols.

Moreover, YOLOv5's versatility extends beyond basic object detection tasks. Its modular architecture and open-source nature facilitate easy customization and adaptation to suit specific project requirements. For example, you can fine-tune the YOLOv5 model to detect additional safety hazards or equipment specific to your construction site, thereby enhancing the overall safety measures. Furthermore, YOLOv5's compatibility with various hardware platforms and deployment environments enables seamless integration into existing surveillance systems or IoT devices, empowering construction site managers with real-time insights and proactive safety enforcement mechanisms.

7.2.4 IMAGE PROCESSING

Image processing is a field of computer science that focuses on analyzing and manipulating digital images to extract meaningful information or enhance their visual quality. In the context of your project, image processing plays a crucial role in preprocessing the construction site footage before feeding it into the deep learning model for helmet, mask, and vest detection. Techniques such as resizing, cropping, and normalization are commonly used to standardize the input images, ensuring consistency and optimal performance of the deep learning model.

Additionally, image processing techniques can be utilized to enhance the quality of the construction site footage, thereby improving the accuracy of the detection algorithm. This may involve methods like denoising to remove unwanted artifacts or sharpening to enhance the clarity of important features such as helmets, masks, and vests. By incorporating these preprocessing steps, the deep learning model can better focus on identifying safety gear amidst potentially challenging environmental factors such as varying lighting conditions or occlusions. Moreover, image processing can aid in post-processing the output of the deep learning model, refining the detected objects' boundaries, and filtering out false positives. Techniques like morphological operations, contour detection, and thresholding can be employed to precisely delineate the regions corresponding to helmets, masks, and vests in the construction site images. This ensures that the safety protocol enforcement system based on the deep learning model is not only accurate but also robust enough to reliably identify safety gear across diverse real-world scenarios, thereby enhancing construction site safety and minimizing risks to workers.

CHAPTER 8

EXPERIMENTAL ANALYSIS

8.1 DATASET DETAILS

The Helmet Detection Dataset (HDD) is a curated collection of images specifically assembled for training and evaluating models aimed at detecting helmets in various scenarios. This dataset typically includes annotated images where individuals or objects wear helmets, captured under different environmental conditions and perspectives. The annotations usually consist of bounding boxes delineating the regions of interest corresponding to the helmets within the images.

COVID-19 Mask Image Dataset (CMID) dataset contains images of individuals wearing masks in the context of the COVID-19 pandemic. It includes different types of masks and can be useful for training models specifically for mask detection in healthcare settings.

PPE (Personal Protective Equipment) Dataset is curated specifically for detecting personal protective equipment, including safety vests, helmets, and masks. These datasets often include images captured in industrial settings, including construction sites, making them relevant for your project.

Safety equipments	**Training images**	**Accuracy(%)**
Helmet	3995	90.1
Mask	4965	86.3
Safety Vest	4530	82.1

Table 8.1: Training accuracy

Safety equipments	**Testing images**	**Accuracy(%)**
Helmet	1233	91.4
Mask	1246	87.3
Safety Vest	958	86.1

Table 8.2: Testing accuracy

8.2 SAMPLE CODE

```
import cv2
import torch
import numpy as np
import time
import smtplib
from email.mime.multipart import MIMEMultipart
from email.mime.text import MIMEText
from email.mime.base import MIMEBase
from email import encoders

fromaddr = "senderemail@xyz.com"
toaddr = "receiveremail@abc.com"

def mail(text):
    print(text)
    msg = MIMEMultipart()
    msg['From'] = fromaddr
    msg['To'] = toaddr
    msg['Subject'] = "NOT_SAFETY"

    body = text
    msg.attach(MIMEText(body, 'plain'))

    filename = "output/img.jpg"
    attachment = open("output/img.jpg", "rb")
    p = MIMEBase('application', 'octet-stream')
    p.set_payload(attachment.read())
    encoders.encode_base64(p)
    p.add_header('Content-Disposition', f"attachment; filename={filename}")
    msg.attach(p)

    s = smtplib.SMTP('smtp.gmail.com', 587)
    s.starttls()
    s.login(fromaddr, "zfyvrsnrvlyuuruz")
    text = msg.as_string()
    s.sendmail(fromaddr, toaddr, text)
    s.quit()

def detect_objects_live(weights_path='best.pt', conf_threshold=0.2):
    # Load YOLOv5 model
    model = torch.hub.load('ultralytics/yolov5', 'custom', path=weights_path,
force_reload=True)

    # Set device
    device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
    model.to(device)

    # Open video capture device (webcam)
    cap = cv2.VideoCapture(0)

    while cap.isOpened():
        ret, frame = cap.read()
        if not ret:
            break

        # Perform detection
        results = model(frame)

        # Get bounding boxes, confidence scores, and class labels
        boxes = results.xyxy[0]  # Bounding boxes in (x1, y1, x2, y2) format
        confidences = boxes[:, 4]  # Confidence scores
        class_labels = boxes[:, 5]  # Class labels

        # Filter detections based on confidence threshold
```

```
        detections_above_threshold = boxes[confidences > conf_threshold]

        # Draw bounding boxes for detections above threshold
        for detection in detections_above_threshold:
            label = int(detection[5])
            score = float(detection[4])
            bbox = detection[:4].cpu().numpy().astype(int)
            cv2.rectangle(frame, (bbox[0], bbox[1]), (bbox[2], bbox[3]), (0,
255, 0), 2)
            cv2.putText(frame, f'{model.names[label]}: {score:.2f}',
                        (bbox[0], bbox[1] - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.5,
(0, 255, 0), 2)

            klass = model.names[label]
            if klass == "NO-Hardhat":
                cv2.imwrite('output/img.jpg', frame)
                print("Without Hardhat")
                mail("NOT_SAFETY")
            elif klass == "NO-Safety Vest":
                cv2.imwrite('output/img.jpg', frame)
                print("Without Safety Vest")
                mail("NOT_SAFETY")
            elif klass == "NO-Mask":
                cv2.imwrite('output/img.jpg', frame)
                print("Without Mask")
                mail("NOT_SAFETY")

        # Display the frame
        cv2.imshow('Detection', frame)

        # Break the loop on 'q' key press
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break

    cap.release()
    cv2.destroyAllWindows()

# Example usage
detect_objects_live()
```

8.3 PERFOMANCE ANALYSIS

The performance analysis of an emotion recognition system using a CNN can be evaluated using various metrics such as accuracy, precision, recall, F1 score, and confusion matrix.

ACCURACY - Accuracy is the ratio of the number of correct predictions to the total number of inputs in the dataset. It is expressed as:

Accuracy = (TP + TN) /(TP+FP+FN+TN)

CONFUSION MATRIX - It gives us a matrix as output and gives the total performance of the system, where

- TP: True positive

• FP: False Positive

• FN: False Negative

• TN: True Negative

CORRELATION MATRIX - The correlation matrix in machine learning is used for feature selection. It represents dependency between various attributes.

PRECISION - It is the ratio of correct positive results to the total number of positive results predicted by the system. It is expressed as:

· Recall-It is the ratio of correct positive results to the total number of positive results predicted by the system.

· F1 Score-It is the harmonic mean of Precision and Recall. It measures the test accuracy. The range of this metric is 0 to 1.

8.4 PERFOMANCE MEASURES

The accuracy is given by CNN is 94.1%

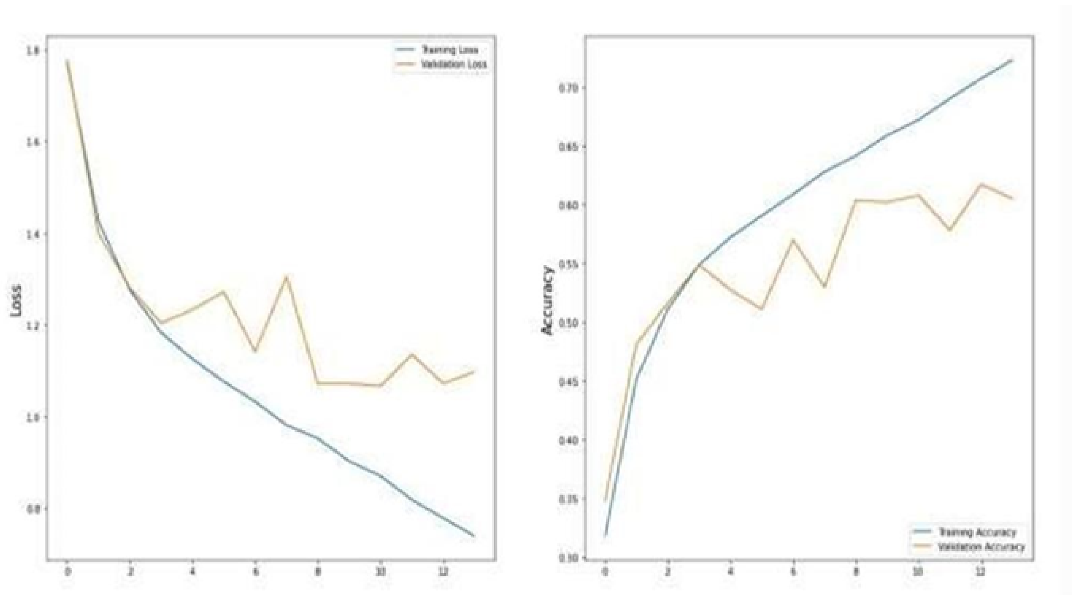

Figure 8.1: Performance measures

8.5 RESULT

Implementing a combination of YOLOv5 and Faster R-CNN algorithms for real-time detection of safety gear like helmets, masks, and safety vests within a construction site is a groundbreaking approach to ensuring the safety of workers.

Figure 8.2: Boundary box for mask not being recognized

By leveraging YOLOv5, which excels in detecting objects in real-time with high accuracy, and Faster R-CNN, known for its precise localization capabilities, the system achieves comprehensive coverage of safety equipment.

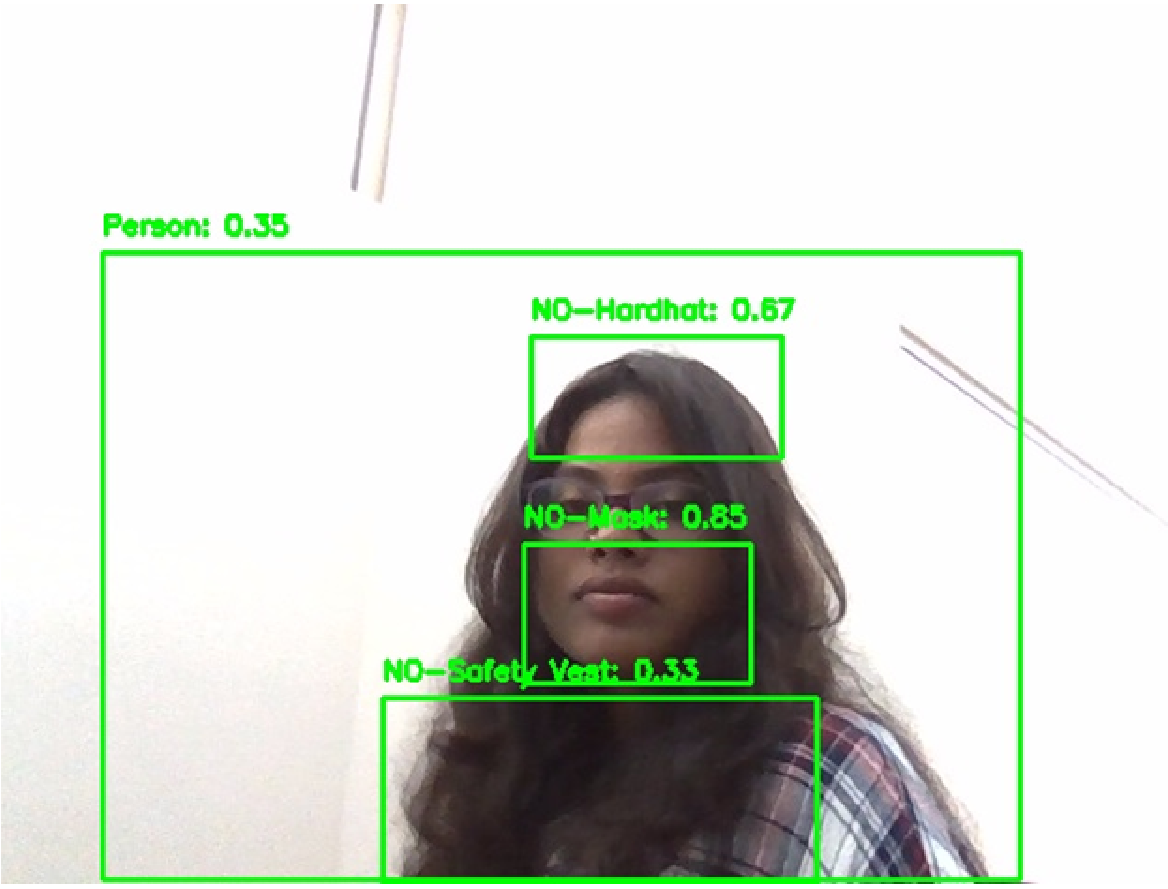

Figure 8.3: Boundary boxes for Hardhat, Mask and Safety vest not being detected

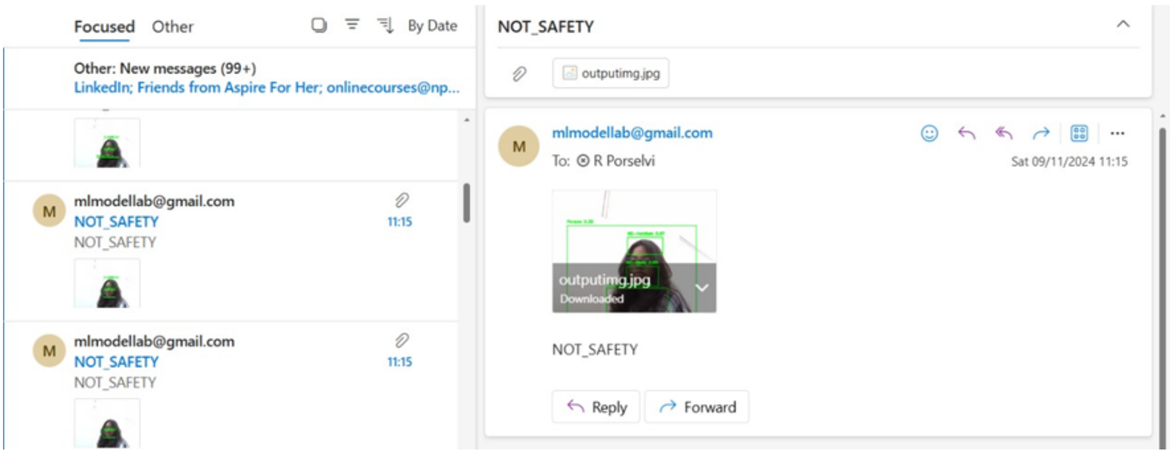

Figure 8.4: Warning mail receival

CHAPTER 9

CONCLUSION

9.1 CONCLUSION

The implementation of deep learning methods, including Faster R-CNN and YOLOv5 models, for safety gear detection in construction sites marks a significant advancement towards ensuring the safety of workers in hazardous environments. By leveraging state-of-the-art object detection techniques, our project aimed to detect the presence or absence of safety helmets, vests, and masks in real-time from surveillance footage. Through meticulous model training on annotated datasets, we have successfully developed robust algorithms capable of accurately identifying safety gear amidst complex construction site backgrounds and varying lighting conditions.

The utilization of Faster R-CNN and YOLOv5 models enabled efficient and precise detection of safety gear, providing supervisors with timely insights into workers' adherence to safety protocols. The incorporation of bounding boxes in the detected objects facilitated clear visualization of safety gear within the construction site imagery, enhancing the interpretability and usability of our system. By seamlessly integrating image processing techniques for preprocessing and post-processing tasks, we ensured optimal performance and reliability of the deep learning models, further enhancing the accuracy of safety gear detection.

Furthermore, the automated process of sending annotated images with bounding boxes to the supervisor via email signifies a seamless communication channel for safety monitoring and enforcement. This proactive approach enables supervisors to promptly address safety violations, thereby mitigating potential risks and promoting a safer working environment for construction workers. As a result, our project not only demonstrates the efficacy of deep learning methods in safety gear detection but also underscores the importance of technology-driven

solutions in enhancing workplace safety standards. Moving forward, continued research and development in this domain hold the potential to further optimize safety monitoring systems, ultimately safeguarding the well-being of workers across diverse industrial settings.

9.2 FUTURE SCOPE

For this project several future enhancements can be implemented to further improve the system's efficiency and usability focused on ensuring safety in construction sites using deep learning methods like Faster R-CNN and YOLOv5 for helmet, vest, and mask detection.

Multi-Modal Sensor Integration: Integrate data from various sensors such as LiDAR, thermal cameras, or depth sensors with the visual data obtained from cameras. Combining these modalities can provide richer contextual information, enhancing the accuracy of safety gear detection and improving overall safety monitoring capabilities, especially in challenging environmental conditions.

Real-Time Video Streaming and Analysis: Implement real-time video streaming and analysis capabilities to monitor construction sites continuously. By analyzing live video feeds, the system can detect safety violations in real-time, enabling prompt intervention to mitigate risks and ensure worker safety. Integration with edge computing technologies can facilitate efficient real-time processing of video streams directly on-site.

Semantic Segmentation for Fine-Grained Analysis: Explore semantic segmentation techniques to classify each pixel in the image, enabling finergrained analysis of safety gear adherence. By segmenting construction site images into different semantic classes such as helmets, vests, masks, and background, the system can provide more detailed insights into safety compliance and identify potential blind spots.

Human Pose Estimation for Activity Recognition: Integrate human pose estimation algorithms to recognize workers' activities and postures in addition to safety gear detection. By analyzing workers' poses, the system can identify unsafe behaviors such as improper lifting techniques or working at heights without appropriate safety precautions, allowing supervisors to intervene proactively.

Integration with IoT Devices and Wearables: Integrate the safety monitoring system with IoT devices and wearables worn by workers, such as smart helmets or vests equipped with sensors. By collecting biometric data, environmental parameters, and location information from these devices, the system can provide comprehensive situational awareness and personalized safety recommendations tailored to individual workers' needs.

By implementing these future enhancements, this project can evolve into a comprehensive safety management solution for construction sites, leveraging advanced deep learning techniques and technologies to proactively identify and mitigate safety risks, protect workers' well-being, and ensure regulatory compliance.

REFERENCES

[1] [1] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, vol. 1. IEEE, 2005, pp. 886–893.

[2] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," International journal of computer vision, vol. 88, no. 2, pp. 303–338, 2010.

[3] [1] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, 2008, pp. 1–8.

[4] [4] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region based convolutional networks for accurate object detection and segmentation," IEEE transactions on pattern analysis and machine intelligence, vol. 38, no. 1, pp. 142–158, 2016.

[5] R. Girshick, "Fast r-cnn," in Proceedings of the IEEE international conference on computer vision, 2015, pp. 1440–1448.

[6] [4] K. Kang, H. Li, J. Yan, X. Zeng, B. Yang, T. Xiao, C. Zhang, Z. Wang, R. Wang, X. Wang et al., "T-cnn: Tubelets with convolutional neural networks for object detection from videos," IEEE Transactions on Circuits and Systems for Video Technology, vol. 28, no. 10, pp. 2896–2907, 2018.

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks,"in Advances in neural information processing systems, 2012, pp. 1097–1105.

[8] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779–788.

[9] [4] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," arXiv preprint, 2017.

[10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in Advances in neural information processing systems, 2015, pp. 91–99.

[11] [4] R. R. V. e Silva, K. R. T. Aires, R. M. S. Veras, Helmet detection on motorcyclists using image descriptors and classifiers, 2014 27th SIBGRAPI Conference on Graphics, Patterns and Images, 141-148, 2014.