HelmSecure: AI Helmet Enforcement

A Project Report submitted in the partial fulfillment of
the Requirements for the award of the degree

BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE AND ENGINEERING

Submitted by

T. Bhuvanesh (21471A05K4)
D. Tharun Kumar (21471A05E8)
K. Vinay (21471A05H4)

Under the Esteemed Guidance of
Dr. S.N.Tirumala Rao,M.Tech,Ph.D
Professor & HOD

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
NARASARAOPETA ENGINEERING COLLEGE: NARASAROPET
(AUTONOMOUS)
Accredited by NAAC with A+ Grade and NBA under Tyre -1 NIRF rank in the band of 201-300 and an ISO 9001:2015 Certified
Approved by AICTE, New Delhi, Permanently Affiliated to JNTUK, Kakinada
KOTAPPAKONDA ROAD, YALAMANDA VILLAGE, NARASARAOPET-522601

2024-2025

NARASARAOPETA ENGINEERING COLLEGE (AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project that is entitled with the name **" HelmSecure: AI Helmet Enforcement "** is a bonafide work done by the team **Bhuvanesh Thotakura (21471A05K4), Tharun Kumar Dondapati(21471A05E8), Kuchi Vinay(21471A05H4)**in partial fulfillment of the requirements for the award of the degree of BACHELOR OF TECHNOLOGY in the Dep artment of COMPUTER SCIENCE AND ENGINEERING during 2024-2025.

PROJECT GUIDE

Dr. S.N.Tirumala Rao, M.Tech,Ph.D

Professor & HOD

PROJECT CO-ORDINATOR

Dr . M. Sireesha ,M.Tech,Ph.D

Associate Professor

HEAD OF THE DEPARTMENT

Dr. S.N.Tirumala Rao,M.Tech,Ph.D

Professor & HOD

EXTERNAL EXAMINER

DECLARATION

We declare that this project work titled "**HelmSecure: AI Helmet Enforcement"** is composed by ourselves that the work contain here is our own except where explicitly stated otherwise in the text and that this work has not been submitted for anyother degree or professional qualification except as specified.

T. Bhuvanesh (21471A05K4)

D. Tharun Kumar (21471A05E8)

K.Vinay (21471A05H4)

ACKNOWLEDGEMENT

We wish to express our thanks to various personalities who are responsible for the completion of the project. We are extremely thankful to our beloved chairman sri **M. V.Koteswara Rao**, **B.Sc.,** who took keen interest in us in every effort throughout this course.We owe our sincere gratitude to our beloved principal **Dr. S. Venkateswarlu, M.Tech.,Ph.D.,** for showing his kind attention and valuable guidance throughout the course.

We express our deep-felt gratitude towards **Dr. S. N. Tirumala Rao, M.Tech., Ph.D.,** HOD of CSE department and also to our guide **Dr. S. N. Tirumala Rao, M.Tech., Ph.D.,** of CSEdepartment whose valuable guidance and unstinting encouragement enable us to accomplish our project successfully in time.

We extend our sincere thanks to **Dr . M. Sireesha ,M.Tech,Ph.D.,** Associate professor & Project coordinator of the project for extending her encouragement. Their profound knowledge and willingness have been a constant sourceof inspiration for us throughout this project work.

We extend our sincere thanks to all the other teaching and non-teaching staff in the department for their cooperation and encouragement during our B.Tech degree.

We have no words to acknowledge the warm affection, constant inspiration and encouragement that we received from our parents.

We affectionately acknowledge the encouragement received from our friends and those who were involved in giving valuable suggestions clarified our doubts, which really helped us in successfully completing our project.

By,

T. Bhuvanesh (21471A05K4)
D. Tharun Kumar (21471A05E8)
K. Vinay (21471A05H4)

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

INSTITUTE VISION AND MISSION

INSTITUTION VISION

To emerge as a Centre of excellence in technical education with a blend of effective student centric teaching learning practices as well as researchfor the transformation of lives and community,

INSTITUTION MISSION

M1: Provide the best class infra-structure to explore the field of engineering and research

M2: Build a passionate and a determined team of faculty with student centric teaching, imbibing experiential, innovative skills

M3: Imbibe lifelong learning skills, entrepreneurial skills and ethical values in students for addressing societal problems

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VISION OF THE DEPARTMENT

To become a centre of excellence in nurturing the quality Computer Science & Engineering professionals embedded with software knowledge, aptitude for researchand ethical values to cater to the needs of industry and society.

MISSION OF THE DEPARTMENT

The department of Computer Science and Engineering is committed to

M1: Mould the students to become Software Professionals, Researchers and Entrepreneurs by providing advanced laboratories.

M2: Impart high quality professional training to get expertize in modern software tools and technologies to cater to the real time requirements of the Industry.

M3: Inculcate team work and lifelong learning among students with a sense of societal and ethical responsibilities.

NARASARAOPETA ENGINEERING COLLEGE (AUTONOMOUS)

Program Specific Outcomes (PSO's)

PSO1: Apply mathematical and scientific skills in numerous areas of Computer Science and Engineering to design and develop software-based systems.

PSO2: Acquaint module knowledge on emerging trends of the modern era inComputer Science and Engineering

PSO3: Promote novel applications that meet the needs of entrepreneur, environmental and social issues.

Program Educational Objectives (PEO's)

The graduates of the programme are able to:

PEO1: Apply the knowledge of Mathematics, Science and Engineering fundamentals to identify and solve Computer Science and Engineering problems.

PEO2: Use various software tools and technologies to solve problems related to academia, industry and society.

PEO3: Work with ethical and moral values in the multi-disciplinary teams and can communicate effectively among team members with continuous learning.

PEO4: Pursue higher studies and develop their career in software industry.

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

Program Outcomes

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineeringfundamentals, and an engineering specialization to the solution of complex engineering problems.

2. **Problem analysis:** Identify, formulate, research literature, and analyse complexengineering problems reaching substantiated conclusions using first principles ofmathematics, natural sciences, and engineering sciences.

3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activitieswith an understanding of the limitations.

6. **The engineer and society:** Apply reasoning informed by the contextual knowledge toassess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12 . Life-long learning: Recognize the need for, and have the preparation and ability to engagein independent and life-long learning in the broadest context of technological change.

NARASARAOPETA
NEC ENGINEERING COLLEGE
(AUTONOMOUS)

Project Course Outcomes (CO'S):

CO421.1: Analyse the System of Examinations and identify the problem.

CO421.2: Identify and classify the requirements.

CO421.3: Review the Related Literature

CO421.4: Design and Modularize the project

CO421.5: Construct, Integrate, Test and Implement the Project.

CO421.6: Prepare the project Documentation and present the Report using appropriate method.

Course Outcomes – Program Outcomes mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C421.1		√											√		
C421.2	√		√		√								√		
C421.3				√		√	√	√					√		
C421.4			√			√	√	√					√	√	
C421.5					√	√	√	√	√	√	√	√	√	√	√
C421.6									√	√	√		√	√	

Course Outcomes – Program Outcome correlation

	PO1	PO 2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C421.1	2	3											2		
C421.2			2		3								2		
C421.3				2		2	3	3					2		
C421.4			2			1	1	2					3	2	
C421.5					3	3	3	2	3	2	2	1	3	2	1
C421.6									3	2	1		2	3	

Note: The values in the above table represent the level of correlationbetween CO’s and PO’s:

1. Low level
2. Medium level
3. High level

Project mapping with various courses of Curriculum with Attained PO’s:

Name of the course from which principles are appliedin this project	**Description of the device**	**Attained PO**
C2204.2, C22L3.2	Developing an image manipulation detection model using YOLOv8 for classification.	PO1, PO3
CC421.1, C2204.3, C22L3.2	Each and every requirement is critically analyzed, the process modelis identified	PO2, PO3
CC421.2, C2204.2, C22L3.3	Logical design is done by using the unified modelling language which involves individual team work	PO3, PO5, PO9
CC421.3, C2204.3, C22L3.2	Each and every module is tested, integrated, and evaluated in our project	PO1, PO5
CC421.4, C2204.4, C22L3.2	Documentation is done by all our three members in the form of a group	PO10
CC421.5, C2204.2, C22L3.3	Each and every phase of the work ingroup is presented periodically	PO10, PO11
C2202.2, C2203.3, C1206.3,C3204.3, C4110.2	The image manipulation detection project using YOLOv8 and ELA is implemented for the common public, with future updates planned for forged video detection.	PO4, PO7
C32SC4.3	The project uses CCTV cameras on roads to detect whether a person is wearing a helmet or not, with future updates planned for forged video detection.	PO5, PO6

ABSTRACT

Ensuring helmet use is crucial for reducing head injuries in workplaces and on the road, yet many workers and motorcycle riders neglect this safety measure due to discomfort, lack of awareness, or disregard for regulations. To tackle this issue, a dual-purpose AI-powered detection system has been developed to monitor helmet compliance in construction sites and among motorcycle riders using advanced computer vision techniques. The system automatically analyzes real-time video streams and images, identifying violations and flagging non-compliant individuals. It stores relevant data, including images, timestamps, and for motorcycle riders, license plate numbers, ensuring effective enforcement. This structured approach minimizes manual oversight, enhances safety monitoring, and provides a user-friendly interface for easy deployment by safety officers and law enforcement. Designed to function in diverse real-world conditions, the system adapts to lighting, weather, and camera variations. Future enhancements could integrate Deep Learning-based facial recognition to track repeat offenders, IoT-based real-time alerts, and cloud storage for large-scale monitoring. Scalable across industrial zones and urban traffic management, this technology significantly improves compliance, reduces accidents, and fosters a stronger culture of safety awareness.

Keywords: Helmet Detection ,Workplace Safety, Motorcycle Rider Monitoring, License Plate Recognition, Image Processing, Real-Time Detection, YOLOv8,Object Detection.

INDEX

LIST OF FIGURES

1. INTRODUCTION

With the rapid expansion of urban areas, the demand for improved infrastructure and efficient resource management has significantly increased. [1]As cities grow, the construction industry plays a crucial role in shaping modern urban landscapes. However, construction sites are inherently hazardous environments, and ensuring worker safety remains a top priority. The rising number of accidents on construction sites has highlighted the importance of strict safety regulations and the enforcement of personal protective equipment (PPE). Among the various safety measures, wearing helmets is one of the most effective ways to protect workers from head injuries caused by falling objects or collisions. The use of safety helmets is legally mandated worldwide, yet many workers neglect to wear them due to discomfort, lack of awareness, or non-compliance with safety regulations.

Traditional methods of monitoring helmet compliance often rely on manual supervision through security personnel or reviewing surveillance footage. However, these approaches are time-consuming, prone to human error, and inefficient in large-scale construction sites. Emerging technologies, particularly computer vision and Deep Learning, offer promising solutions for automating compliance monitoring. Various computer vision techniques, such as the Gaussian Mixture Model (GMM), Histogram of Oriented Gradient (HOG), and Support Vector Machine (SVM), have been explored to detect and classify safety helmets in construction environments[2]. However, these methods have limitations in accurately detecting helmets, especially in challenging conditions such as low visibility, occlusions, and cluttered backgrounds.

With the continuous advancements in Deep Learning and computational power, more sophisticated models have been developed for object detection. Among them, the YOLO (You Only Look Once) algorithm has emerged as a powerful tool for real-time object detection due to its high speed and accuracy. Despite its effectiveness, YOLO-based methods still face challenges in accurately identifying helmets in complex environments[3]. Small, monochromatic helmets can be misclassified as other objects, and overlapping workers or moving vehicles can create occlusions, making detection even more difficult.

To address these challenges, this study proposes a dual-purpose detection system that enhances safety monitoring in both industrial and vehicular contexts. The system is designed to detect whether a worker is wearing a helmet at a construction site and extend its functionality to motorcycle riders by verifying helmet usage and capturing license plate information. This automated system records detected individuals' images, extracts the motorcycle's license plate number, and timestamps the information in an Excel sheet, ensuring comprehensive safety enforcement and documentation.

License plate recognition is an essential aspect of smart transportation systems, helping authorities manage traffic, identify violators, and enhance road safety. However, recognizing license plates in real-world conditions presents several challenges, such as poor lighting, motion blur, and occlusions. Traditional license plate recognition involves three main steps: detection, segmentation, and recognition. However, this process can be inefficient and unreliable in complex scenarios. Recent advancements in Deep Learning, particularly Convolutional Neural Networks (CNNs), have significantly improved object detection tasks. [4]CNN-based models, such as YOLO, have demonstrated superior performance in real-time target identification and are widely used in various computer vision applications.

Given the increasing concerns over safety and security, this study aims to develop a robust and scalable system that leverages Deep Learning techniques for safety helmet detection and license plate recognition. By integrating these technologies, the proposed system enhances compliance monitoring, reduces workplace hazards, and improves enforcement capabilities in both construction and transportation sectors. The ultimate goal is to create a safer working and commuting environment through automation and intelligent monitoring.

With the rapid urbanization and growth of cities, construction activities have increased significantly, leading to a greater demand for infrastructure and efficient management systems. [5]One of the major concerns in construction sites is worker safety, as accidents and injuries can have severe consequences. The use of Personal Protective Equipment (PPE), particularly safety helmets, plays a crucial role in preventing injuries caused by falling objects or equipment mishaps. However, despite legal mandates, many workers neglect to wear helmets due to discomfort, lack of awareness, or negligence.

Traditional methods of monitoring helmet compliance rely heavily on human supervision and video surveillance, which are time-consuming, inefficient, and prone to errors. The integration of advanced technologies such as computer vision and Deep Learning presents a promising solution for enhancing safety compliance. Recent advancements in artificial intelligence (AI) and Deep Learning algorithms, particularly the YOLO (You Only Look Once) model, have demonstrated high efficiency in object detection tasks, making them suitable for real-time helmet detection.

The motivation behind this study stems from the need to improve safety standards in construction and transportation sectors. [6]By developing a dual-purpose detection system that monitors helmet usage among both industrial workers and motorcycle riders, we aim to reduce accidents and ensure better compliance with safety regulations. Additionally, the system enhances enforcement by automatically storing relevant data, such as images of detected individuals, license plate numbers, and timestamps, in an Excel sheet for further analysis.

By leveraging the power of Deep Learning [7]and real-time detection, this project seeks to create a robust and scalable solution that minimizes human intervention while maximizing safety and efficiency in diverse environments.

Ensuring helmet compliance at construction sites and on roads is a significant challenge due to the limitations of traditional monitoring methods. Manual supervision is labor-intensive, inconsistent, and subject to human error, while video surveillance often requires post-processing analysis, making real-time intervention difficult.

Key challenges in helmet compliance monitoring include:

- **Negligence and Non-Compliance:** Workers and motorcyclists often disregard helmet rules due to discomfort, lack of awareness, or intentional negligence.
- **Human-Dependent Monitoring:** Traditional surveillance systems rely on human observers to review footage and detect non-compliance, which can be inefficient and prone to oversight.

- **Environmental Challenges:** Factors such as poor lighting, occlusions from other objects or people, and varied helmet colors make automated detection difficult.
- **Limited Scalability:** Existing compliance monitoring methods do not scale well in large construction sites or busy urban areas with high motorcycle traffic.

To address these issues, our system integrates AI-powered helmet detection using the YOLO model, which enables:

- **Real-Time Monitoring:** The system processes live camera feeds to instantly detect and flag individuals not wearing helmets.
- **Automated Data Logging:** Non-compliance incidents are recorded, with captured images and timestamps stored in an Excel sheet for further enforcement.
- **License Plate Recognition:** For motorcyclists, the system captures and logs license plate numbers, aiding law enforcement in tracking repeat offenders.
- **Scalability and Flexibility:** The model can be deployed across various environments, including construction sites, highways, and industrial workplaces.

By leveraging Deep Learning and automation, our solution enhances safety compliance, reduces manual workload, and provides a robust method for enforcing helmet regulations.

The primary objective of this study is to develop an AI-powered system that detects helmet usage in real-time for both construction workers and motorcycle riders. The system aims to improve safety compliance and enforcement through automated monitoring and data recording.

The specific objectives include:

I. Develop an AI-Based Helmet Detection System

- Implement a YOLO-based Deep Learning model for real-time helmet detection.
- Enhance detection accuracy to differentiate between helmeted and non-helmeted individuals, even in challenging environments.

- Ensure minimal false positives and fa lse negatives for effective compliance monitoring.

II. Extend Functionality to Motorcycle Riders

- Integrate an automated license plate recognition (LPR) system to identify motorcyclists violating helmet regulations.
- Store detected license plate numbers along with timestamped images for enforcement purposes.
- Provide an efficient and reliable method for authorities to track helmet compliance among riders.

III. Automate Data Logging and Reporting

- Capture and store detection results, including images, timestamps, and license plate data, in an Excel sheet.
- Enable easy access and retrieval of records for auditing and compliance enforcement.
- Develop a dashboard or reporting tool for visualizing helmet compliance statistics over time.

IV. Improve System Accuracy and Robustness

- Optimize the model for real-world deployment, considering variations in lighting, occlusions, and helmet colors.
- Enhance the system's adaptability for different environments, including construction sites, highways, and industrial zones.
- Reduce computational overhead to ensure real-time processing and quick response times.

V. Ensure Scalability and Future Integration

- Design the system to be scalable, allowing deployment across large construction projects and urban monitoring setups.

- Enable integration with cloud storage, mobile applications, and law enforcement databases.
- Explore additional AI-based safety features such as behavior analysis and crowd monitoring.

By achieving these objectives, our system aims to provide a comprehensive, efficient, and scalable solution for helmet compliance monitoring, significantly improving safety standards in workplaces and on roads.

2. LITERATURE SURVEY

Traditional safety helmet detection methods have primarily relied on manual feature extraction techniques. These methods utilized various image processing techniques to identify helmets in visual data. For example, edge detection algorithms, color segmentation, and template matching were commonly used to differentiate helmets from their surroundings. Despite their simplicity, these methods often struggled with accuracy, particularly in complex environments with varying lighting conditions and background clutter. The high rate of false positives and negatives limited the practical application of these approaches in real-world scenarios, especially when helmets were small or partially obscured.

One of the earliest techniques involved By utilizing The combination of Support Vector Machines (SVM) and Histogram of Oriented Gradients (HOG) [8] has been widely utilized for object detection. While HOG effectively captured shape information, it was prone to errors when faced with objects that had similar edge features to helmets, leading to misclassifications. Another method, Circular Hough Transform (CHT) [9], was employed to detect circular shapes resembling helmets. However, its reliance on geometric assumptions made it less effective in dynamic and cluttered environments where helmet shapes could be distorted or obscured.

Traditional helmet detection methods relied on manual feature extraction techniques such as edge detection and color segmentation. These approaches were useful in basic environments but struggled with accuracy in more complex conditions, like when lighting varied or backgrounds were cluttered. They often produced too many errors, with false positives and negatives, especially when helmets were small or partially hidden. As a result, these techniques were not practical for real-world use, where more advanced and reliable methods are needed to ensure proper helmet detection.

With the advent of Deep Learning, safety helmet detection has seen significant improvements in accuracy and robustness. Convolutional Neural Networks (CNNs) have revolutionized object detection by automatically learning hierarchical features from data, reduces the need for manual feature engineering. This shift has enabled more reliable detection in complex environments and has greatly improved performance in real-time applications.

Deep Learning based methods can be categorized in two main types: two-stage detectors and one-stage detectors. Two-stage detectors, like the Region-based Convolutional Neural Network (R-CNN) [10] family, first generate region proposals and then classify them. These methods have shown high accuracy but often suffer from slower processing speeds, making them less suitable for real-time helmet detection. Improvements like Fast R-CNN and Faster R-CNN have attempted to address these issues by optimizing the region proposal process and sharing convolutional features, yet they still fall short in scenarios requiring rapid detection.

Alternately one-stage detectors like the You Only Look Once (YOLO) series have gained popularity due to their ability to perform object detection in a single pass through the network. YOLO models, particularly the latest YOLOv8[11], have achieved impressive results in terms of both speed and accuracy. YOLOv8 incorporates several enhancements, such as advanced feature fusion techniques and lightweight architectures, making it particularly well-suited for safety helmet detection in real-time settings. Despite these advancements, challenges remain in detecting small objects and handling occlusions in complex backgrounds.

To focus on these challenges, recent research has focused on enhancing the detection capabilities of YOLO-based models by integrating additional modules like attention mechanisms and multi-scale feature fusion networks. These enhancements aim to increase the model's sensitivity to small and partially obscured helmets, thereby improving detection accuracy in diverse and challenging environments.

Currently, Deep Learning methods rely heavily on data, meaning the output of object detection algorithms depends on the quality and quantity of the training datasets. However, there aren’t many images available for safety helmet detection. To enhance the presentation of YOLO v5 in detecting helmet use, this research used techniques like Utilizing focused data augmentation and transfer learning methods. In the end, a user-friendly interface (GUI)[12] was also designed to make the model easier to use.

The main goal of this paper is to propose a network that quickly and accurately detects whether people are wearing helmets on construction sites, while being easy to use. The model should also be versatile, not just limited to one specific scene, and capable of handling small objects and different sizes effectively. The dataset used in this paper takes these factors into account, including a variety of helmet images with different challenges from various scenes.

To address issues like small objects and complex backgrounds, recent research has enhanced YOLO-based models by adding attention mechanisms and multi-scale feature fusion. These improvements aim to boost the protype ability to detect small and partially hidden helmets, making it more accurate in diverse situations. The research also leverages techniques like data augmentation and transfer learning to improve the performance of YOLOv8[13], especially with limited datasets. A user-friendly interface was created to make it easy for non-experts to use the model in various settings.

The primary focus of this study is to develop a model that detects helmets quickly and accurately on construction sites. It should handle different helmet sizes and be adaptable to various environments. The dataset includes diverse helmet images from different scenes to ensure the model performs well in real-world applications.

3. EXISTING SYSTEM

Traditional safety helmet detection methods rely on manual monitoring using camera surveillance, which is both labor-intensive and prone to human errors. These conventional methods require constant supervision, making them inefficient and inconsistent due to factors like fatigue, distractions, or subjective judgment. Additionally, human-based monitoring can be costly and impractical for large-scale implementation, especially in busy construction sites or high-traffic areas. [14]As a result, ensuring compliance with safety regulations using traditional methods becomes a significant challenge.

To address these issues, rule-based and Machine Learning-based approaches have been introduced, utilizing handcrafted features such as color, shape, and texture to detect helmets in images or videos. However, these methods have several drawbacks, including sensitivity to variations in lighting conditions, complex backgrounds, and different helmet colors. They often fail when objects are partially occluded or when helmets appear in unconventional orientations. Moreover, such methods require extensive feature engineering and manual fine-tuning, limiting their scalability and effectiveness in real-world scenarios.

To improve detection accuracy and efficiency, Deep Learning-based models have gained popularity in safety helmet detection. One of the most widely adopted models is YOLO (You Only Look Once)[15], a real-time object detection algorithm known for its speed and accuracy. The YOLOv8 algorithm, in particular, has demonstrated significant advancements in object detection by enhancing feature extraction, reducing processing time, and improving detection performance in various environments. Unlike traditional Machine Learning methods, YOLO-based models automatically learn and extract features from images, making them more robust in handling complex backgrounds, varying lighting conditions, and diverse helmet appearances.

Despite these improvements, standard YOLO-based methods still face challenges in detecting small helmets in cluttered environments due to limited feature extraction capability. In dense and complex scenarios, helmets may appear too small for the model to detect effectively, leading to missed detections. To overcome this limitation, researchers have proposed various enhancements, such as adding attention mechanisms, optimizing feature fusion networks, and incorporating small-target detection layers. These modifications help improve detection accuracy by enabling the model to focus on critical regions, enhancing multi-scale feature representation, and refining object classification.

However, even with these advancements, several challenges remain in safety helmet detection. High computational costs, data imbalance issues, and occlusion-related difficulties continue to hinder the widespread adoption of Deep Learning-based solutions. Training Deep Learning models requires large amounts of annotated data, which can be expensive and time-consuming to collect. Additionally, real-world applications often encounter cases where helmets are partially hidden by objects or appear in distorted perspectives, making detection more complex. Researchers are continuously exploring novel techniques, such as hybrid models, advanced data augmentation methods, and real-time edge computing solutions, to further improve helmet detection accuracy and efficiency.

In conclusion, while traditional safety helmet detection methods have several limitations, Deep Learning-based approaches like YOLOv8[16] have significantly improved accuracy and automation. However, challenges such as small-object detection, occlusions, and high computational costs still need to be addressed for more effective real-world deployment. Future advancements in AI-driven safety monitoring systems will likely focus on optimizing detection models, reducing computational requirements, and integrating smart surveillance technologies for enhanced workplace and road safety.

DISADVANTAGES OF THE EXISTING SYSTEM

Despite advancements in helmet detection using Deep Learning, several challenges remain:

I. High Dependency on Manual Supervision

- Traditional methods rely on human observation, leading to inefficiencies and errors.
- Surveillance fatigue can reduce the effectiveness of manual monitoring.

II. Poor Small Object Detection

- Existing systems struggle with detecting small helmets, particularly in crowded environments.
- Standard Deep Learning models often fail to distinguish helmets in complex backgrounds.

III. Computational Complexity and Hardware Requirements

- Advanced Deep Learning models demand significant computational power.
- Deploying real-time models on edge devices is difficult due to high resource consumption.

IV. Susceptibility to Occlusions and Lighting Variations

- Helmets may be partially or fully obscured, leading to false negatives.
- Low-light conditions and environmental factors affect detection accuracy.

V. Lack of Adaptive Learning Mechanisms

- Most models require retraining when applied to new environments.
- Generalization across different construction sites is limited.

4 PROPOSED SYSTEM

The HelmSecure AI Helmet Enforcement System is designed to provide an advanced and automated solution for ensuring compliance with helmet safety regulations in both industrial and traffic environments. By leveraging Deep Learning, computer vision, and real-time data processing, the system enhances monitoring and enforcement while reducing reliance on manual supervision. The system detects helmet usage among construction workers and motorcycle riders, capturing violations automatically and logging data for further analysis.

For model evaluation, accuracy is determined by the system's ability to correctly detect helmet use in real-world conditions. Performance metrics include detection speed, false positives, and overall model reliability across varying lighting conditions and environmental complexities. The system integrates multiple input sources, such as live video feeds, image uploads, and pre-recorded video footage, ensuring robustness across diverse scenarios. Additionally, all violation data, including images and timestamps, are stored in an Excel sheet for easy tracking and compliance enforcement.

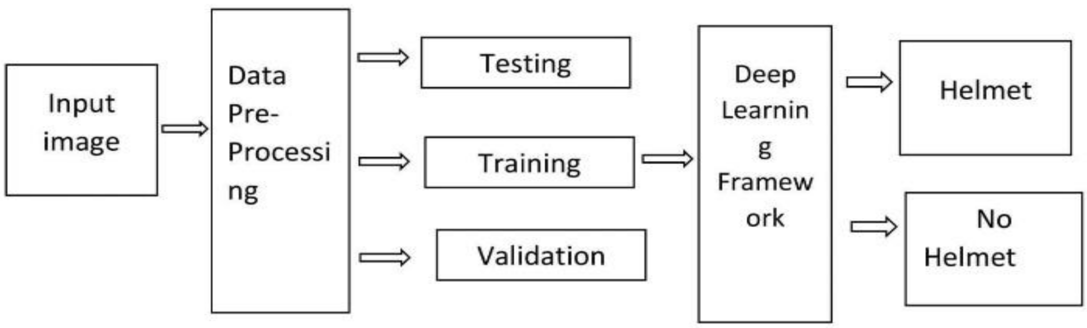


Fig 4.1: Helmet Enforcement System

The proposed system is designed to enhance safety compliance by detecting whether a person is wearing a helmet using Deep Learning techniques. This system is particularly useful in monitoring industrial workers at construction sites and motorcycle riders on the road. By automating helmet detection, it reduces the dependency on manual supervision and ensures real-time enforcement of safety regulations. The system follows a structured workflow, including image acquisition, data pre-processing, training with Deep Learning models, and final classification into helmet or no-helmet categories.

The Fig 4.1 represents the workflow of a proposed helmet detection system using Deep Learning. The system follows a structured approach, beginning with an input image and progressing through multiple stages to classify whether a person is wearing a helmet or not.

The process starts with an input image, which could be captured from a live camera feed or an uploaded file. This image is then sent to the data pre-processing stage, where various techniques such as resizing, noise reduction, and normalization are applied to improve the quality of the data. This step is crucial for ensuring that the model receives clean and well-structured inputs for accurate detection.

Once pre-processed, the data is divided into three categories: training, validation, and testing. The training set is used to teach the Deep Learning model how to recognize helmets, while the validation set helps fine-tune the model parameters to improve accuracy. The testing set is then used to evaluate the model's performance on unseen data, ensuring its reliability in real-world scenarios.

The processed data is fed into a Deep Learning framework, which is responsible for feature extraction and classification. A convolutional neural network (CNN)-based model, such as YOLO (You Only Look Once), is commonly used for object detection tasks due to its real-time processing capability. The model analyzes the image and classifies the detected object into one of two categories: "Helmet" or "No Helmet."

If a helmet is detected, the system confirms compliance with safety regulations. However, if no helmet is detected, an alert can be triggered, and additional enforcement actions, such as capturing license plate information for motorcyclists or logging the violation for construction workers, can be taken. This automated process helps ensure safety compliance in both industrial and vehicular environments, reducing the risk of accidents and improving monitoring efficiency.

By leveraging Deep Learning techniques, this system enhances the accuracy, speed, and scalability of helmet detection. It eliminates the limitations of manual monitoring, ensuring continuous and automated safety enforcement in workplaces and public roads.

Advantages Over Existing System

I. **Automatic and Contactless Monitoring**

Traditional methods for checking helmet use depend on people watching through cameras or manually scanning RFID tags. These methods take a lot of time and effort. The HelmSecure system uses artificial intelligence to detect helmets automatically. It works in real-time without needing human supervision, making the process faster, more efficient, and free from human errors.

II. **Prevents People from Breaking Rules**

Some riders find ways to avoid wearing helmets by tricking manual checks or bypassing RFID systems. With HelmSecure, such violations are prevented. The system uses image recognition and license plate tracking to detect helmet use and automatically record any violations. This ensures that everyone follows the rules, and repeated offenses are tracked for better enforcement.

III. **Lower Cost for Monitoring**

Current helmet-checking methods require a lot of human effort, which makes them expensive. Whether it's hiring people for monitoring or maintaining RFID systems, costs can be high. The HelmSecure system reduces these costs by using AI to automatically detect, analyze, and store helmet violation data. This removes the need for constant human involvement, saving both time and money.

IV. **Better Accuracy in Detection**

Traditional methods often fail due to poor lighting, fast-moving objects, or background distractions. Sometimes, even manual checks can be inconsistent. HelmSecure uses advanced Deep Learning models like YOLOv8, which can recognize helmets more accurately, even in difficult conditions. This reduces mistakes and ensures correct detection of helmet use.

V. **Instant Alerts and Data Reports**

Unlike old systems that only record violations after they happen, HelmSecure provides real-time alerts. If someone is caught without a helmet, an instant notification can be sent to authorities. The system also creates reports showing patterns, such as how often violations happen and at what times. This helps in making better safety rules and improving helmet enforcement.

VI.**Works in Different Places and Can Be Expanded**

Many helmet detection systems are hard to install in multiple locations because they require special equipment and constant human involvement. HelmSecure is designed to work anywhere, whether on highways, in construction sites, or in cities. It can be connected with existing security cameras and traffic systems, making it easy to expand and use in different areas.

VII**. Improves Safety and Follows Laws**

Not wearing a helmet can lead to serious injuries in accidents. Many existing systems do not strictly enforce helmet rules, which increases risks. HelmSecure ensures that helmet laws are followed by automatically detecting violations and storing digital proof. This helps authorities take action and ensures safer roads and workplaces.

VIII. **Easy to Use and Low Maintenance**

Some helmet detection systems need complicated setups or regular maintenance, making them hard to manage. HelmSecure is simple to use, requires minimal hardware, and updates automatically. Since it does not need frequent repairs or adjustments, it is easy to maintain and can work efficiently for a long time.

5. SYSTEM REQUIREMENTS

5.1 Hardware Requirements:

a.	System Type	:	intel®core™i3-7500UCPU@2.40gh
b.	Cache memory	:	4MB(Megabyte)
c.	RAM	:	8GB (Gigabyte) or higher
d.	Hard Disk	:	512GB

5.2 Software Requirements:

a.	Operating System	:	Windows 11, 64-bit Operating System
b.	Coding Language	:	Python
c.	Python distribution	:	Anaconda, Flask , Visual Studio Code.
d.	Browser	:	Any Latest Browser like Google Chrome , Firefox ,etc.

6.SYSTEM ANALAYSIS

6.1 Scope of the Project

The HelmSecure project aims to develop an AI-driven system that ensures real-time helmet compliance monitoring for both industrial workers and motorcycle riders. This intelligent system utilizes advanced computer vision and Deep Learning models to detect helmet usage automatically. By eliminating the need for manual supervision, HelmSecure enhances safety enforcement, reduces human errors, and improves efficiency in monitoring compliance.

This system addresses several challenges present in traditional methods, such as reliance on manual observation, high labor costs, and inaccurate detection due to environmental factors. By leveraging Deep Learning models, HelmSecure ensures accurate detection even in complex backgrounds, poor lighting, or crowded areas. The system can differentiate between individuals wearing helmets and those who are not, providing instant alerts and generating structured reports for authorities.

HelmSecure's dual-purpose functionality allows it to be implemented in various environments, including construction sites, factories, and roadways. In workplace settings, the system monitors workers to ensure they adhere to safety regulations, reducing workplace hazards. For motorcycle riders, it can be integrated with traffic surveillance to identify riders without helmets and capture license plate details for enforcement.

The project's scope includes designing a robust detection model, implementing real-time monitoring capabilities, and developing an intuitive interface for authorities to access violation records. Additionally, the system will be optimized to work efficiently in different locations with minimal maintenance requirements.

By automating helmet detection, HelmSecure significantly improves safety compliance, reduces accidents, and ensures a more structured approach to rule enforcement. This system ultimately contributes to building a safer and more disciplined environment for workers and road users alike.

Key Objectives:

- **Workplace Safety:** Monitor helmet usage on construction sites to ensure compliance with safety regulations and reduce risks associated with head injuries.
- **Motorcycle Rider Monitoring:** Detect helmet usage among motorcycle riders and capture license plate information for those not in compliance.
- **Data Management:** Systematically organize and store detection data, including images, timestamps, and license plate numbers, in an easily accessible format for auditing and enforcement.

Methodology Highlights:

- **Detection Framework:** Utilizes YOLOv8, a cutting-edge object detection model, to achieve real-time helmet detection with high speed and accuracy.
- **Preprocessing:** Includes resizing, normalization, and data augmentation to enhance model robustness and adaptability to diverse conditions.
- **Feature Extraction:** Enhances detection using Slim-Neck fusion and attention mechanisms to identify small or partially obscured helmets in complex environments.
- **Explainability:** Provides interpretable results using visualization techniques to support actionable decision-making.

6.2 Analysis

The HelmSecure system was developed using a structured methodology, starting with extensive data collection. A total of 4,611 images were gathered from various sources, including real-world surveillance footage, publicly available datasets, and controlled environments. These images include scenarios of both construction workers and motorcycle riders, with and without helmets, captured under different lighting conditions, angles, and backgrounds. This diverse dataset ensures that the model can accurately detect helmet usage in real-world situations.

During the preprocessing phase, the collected images were enhanced using techniques like resizing, noise reduction, contrast adjustment, and data augmentation.

These steps improve the model's ability to recognize helmets effectively, even in challenging conditions.

For model training, Deep Learning techniques, particularly the YOLOv8 algorithm, were used due to their high accuracy and real-time detection capabilities. The dataset was divided into training, validation, and testing sets to ensure a well-balanced learning process. The model was trained using optimized parameters to achieve effective helmet detection.

After training, the model was tested on unseen data to verify its performance. The system was evaluated based on its ability to correctly detect helmet usage in different conditions. Adjustments were made where necessary to enhance accuracy and efficiency.

By following this structured approach, the HelmSecure system provides a reliable and automated solution for helmet compliance monitoring, making safety enforcement more efficient in both workplace and traffic environments.

Data Collection

A complete and diversified dataset is fundamental to developing a helmet-detecting model, then testing its performance. A wide array of shots should be taken in different workplaces and working conditions: both helmeted and non-helmeted, with natural/artificial light, at different camera angles, with difficult backgrounds such as building sites with machinery, equipment, and workers. The diversity will ensure that the model is robust enough to carry out the task of helmet detection accurately even in real-world challenging situations. Further, the dataset should also represent subjects of variable heights, variable distances from cameras, and even partial occlusion, in order to test the model for complex detection conditions. Adaptability and precision are therefore boosted by incorporating into the model various colored, sized, and styled helmets to fit a wide array of situations. This level of variety makes a helmet detection system not only versatile but also reliable for a diversity of real-world applications.

I.Helmeted and Non-Helmeted Images:

The proposed helmet detection required the collection of images-both helmeted and not-helmeted various types of environments, including construction sites, roads, and industrial areas. Data are compiled from various sources such as real footage captured by cameras in live circumstances and frames taken from video recordings. This technique would ensure the coverage of different practical scenarios on how individuals are situated in different contexts under various light conditions and within various environmental sets. Such diverse images will give a strong dataset in enhancing the model in detecting helmet usage from various sets of work or road safety condition

II.Annotations:

Each picture in the dataset is carefully annotated to indicate the presence and location of helmets. This process involves drawing bounding boxes around helmets and labeling them correctly. To ensure accuracy, annotations are either done manually by humans or with the help of semi-automatic tools. This precise labeling process is crucial for training the detection model, as it helps the algorithm learn to identify helmets accurately in various scenarios and environments. Proper annotations contribute significantly to the model's overall output and effectiveness in real-world applications.

III.Diversity and Balance:

To balance your dataset, aim to have a similar number of helmeted and non-helmeted images. Include a variety of helmet styles, colors, and sizes to reflect real-world diversity. Collect diverse images, label them appropriately, and apply Data augmentation techniques mitigate potential imbalance. Ensure the dataset is evenly split into training, validation, and test sets, and monitor the model's performance to confirm it generalizes well across all variations.

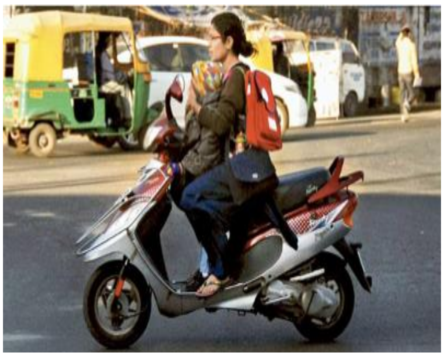


Fig 6.1: Workers And Bike Rider Images With And Without Helmet

Including real-world images of both helmeted and non-helmeted riders, such as the Fig 6.1 above, can help improve the robustness of the dataset. Ensuring diversity in rider demographics, environments, and vehicle types will enhance the model's ability to generalize effectively in various conditions.

6.3 Data Preprocessing:

- **Image Collection:** Images were collected from diverse environments, including construction sites and urban roads, capturing both helmeted and non-helmeted individuals.
- **Augmentation:** Techniques such as random rotations, flips, brightness adjustments, and zoom transformations were applied to simulate real-world variability and enhance model generalization.
- **Normalization:** Pixel values were scaled between 0 and 1 to ensure uniformity and efficient model convergence.
- **Dataset Splitting:** The dataset was divided into:
  - 70% for training and validation.
  - 10% for testing to evaluate the model's accuracy on unseen data.

Observations and Challenges:

- Detection of small or monochromatic helmets in crowded or occluded environments remains a challenge.
- Variability in lighting, perspective, and image quality affects detection accuracy.
- Real-world applications may encounter class imbalance or rare scenarios, impacting generalization.

Proposed Solutions:

- Incorporate additional training data to improve detection capabilities in complex scenarios.
- Use advanced normalization and synthetic data generation techniques to address class imbalance and quality inconsistencies.
- Optimize the model for lightweight deployment using techniques like pruning, ensuring scalability and efficiency.

6.4 Model Building

The model building process for HelmSecure focuses on leveraging the YOLOv8 architecture to develop a robust and efficient helmet detection system. By incorporating advanced training techniques and optimization methods, the model ensures high accuracy and scalability across diverse environments.

Feature Extraction

Feature extraction is a critical component of the HelmSecure system, enabling the identification of helmets and license plates with high precision. The system leverages the YOLOv8 architecture, which integrates advanced techniques to improve detection capabilities, particularly in challenging environments.

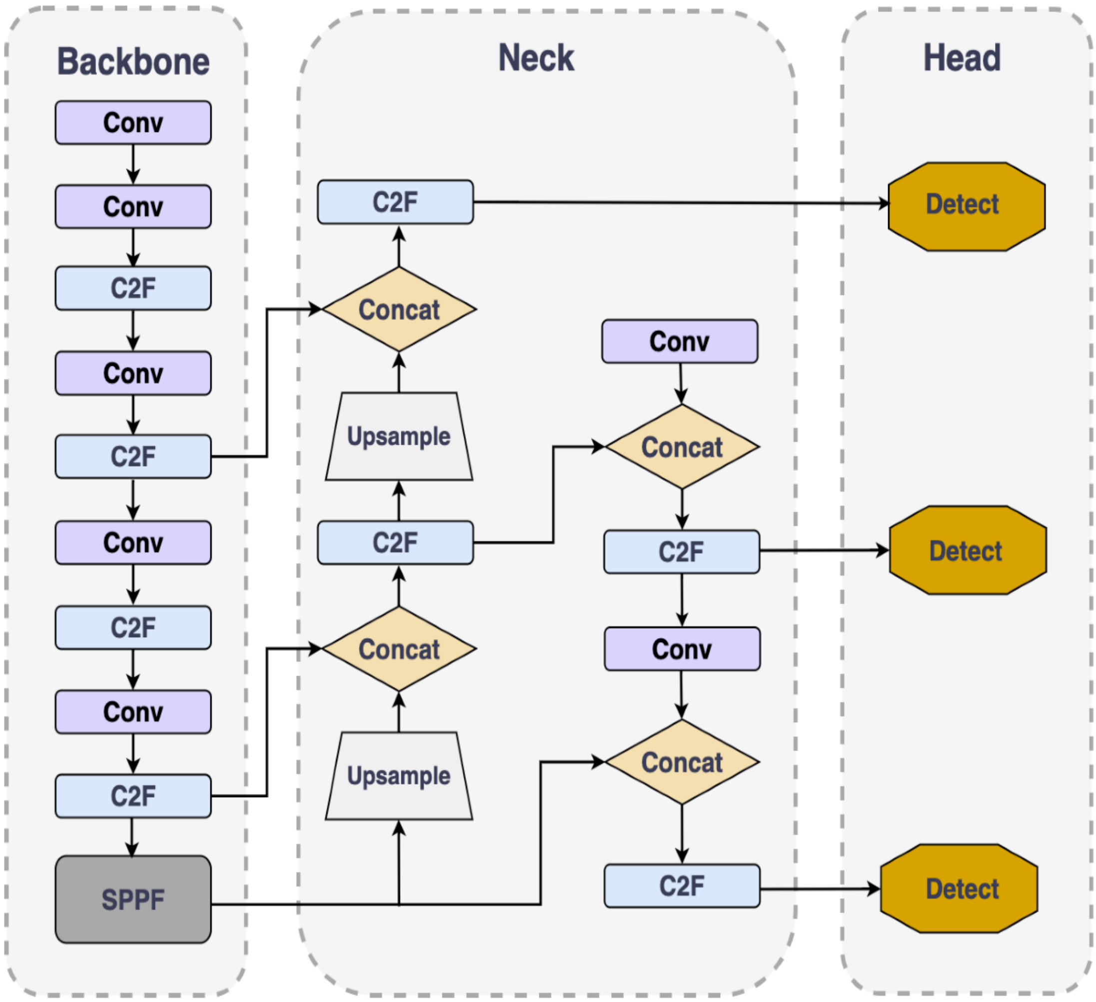


Fig 6.4: Yolov8 Architecture

Key Techniques:

- **Slim-Neck Fusion:** This lightweight network structure enhances feature fusion, reducing the number of parameters while maintaining detection performance.

- **Coordinate Attention Mechanisms:** These mechanisms refine the model's focus on specific regions of interest, such as helmets and license plates, by incorporating spatial and channel-wise attention.
- **Small Target Detection Layer:** An additional detection head is included to handle small or partially obscured objects, such as helmets in crowded scenes or under poor lighting conditions.

Benefits of Feature Extraction:

- Enables robust detection in complex and dynamic environments.
- Enhances model precision for small and obscured objects.
- Reduces false positives and negatives, improving overall system reliability.

Key Steps in Model Building:

- **Architecture Selection:** YOLOv8 was chosen for its high speed and precision, suitable for real-time object detection tasks.
- **Data Integration:** The dataset was preprocessed using techniques like resizing, normalization, and augmentation to ensure uniformity and variability during training.
- **Training Process:**
    - The model was trained using loss functions like Focal Loss and Intersection over Union (IoU) Loss.
    - Stochastic Gradient Descent (SGD) and Adam optimizers were used to minimize loss.
    - Training was conducted over multiple epochs with a learning rate scheduler for optimal convergence.
- **Validation and Testing:** A validation set was used during training to fine-tune hyperparameters, and the final model was tested on unseen data to ensure generalization.
- **Model Optimizations:**
    - **Slim-Neck Architecture:** Reduced computational complexity while maintaining accuracy.
    - **Attention Mechanisms:** Improved focus on regions of interest.
    - **Pruning and Quantization:** Optimized for lightweight deployment.

Implementation Tools:

- Deep Learning frameworks like PyTorch and TensorFlow.
- Advanced libraries for data preprocessing and visualization.

6.5 Classification

The classification phase involves using the trained YOLOv8 model to categorize images into two classes: helmeted or non-helmeted.

Steps in Classification:

1. **Input Processing:**
   - Images are resized to 224x224 pixels and normalized to [0, 1].
   - Augmentation is applied to simulate real-world variability.
2. **Feature Extraction:**
   - The input image is passed through pre-trained layers, extracting high-level features such as edges and textures.
3. **Binary Classification:**
   - Extracted features are fed into fully connected layers.
   - The final layer uses a sigmoid activation function to output a probability score.
4. **Thresholding:**
   - A threshold (commonly set at 0.5) determines classification.

Training and Validation Graphs:

- The training accuracy curve shows how well the model learns over time.
- The validation accuracy curve indicates how well it generalizes to unseen data.
- Divergence between these curves suggests overfitting, requiring adjustments in training.

6.6 Confusion Matrix

The confusion matrix evaluates the classification performance of the HelmSecure system, breaking predictions into four categories:

	Predicted Normal	**Predicted Non-Helmeted**
Actual Normal	TN	FP
Actual Non-Helmeted	FN	TP

Metrics Derived:

- **Accuracy:** (TP + TN) / (TP + TN + FP + FN)
- **Precision:** TP / (TP + FP)
- **Recall (Sensitivity):** TP / (TP + FN)
- **F1-Score:** 2 * (Precision * Recall) / (Precision + Recall)

A confusion matrix was created to assess the model's ability to classify helmet usage correctly, ensuring reliable safety compliance monitoring in workplaces and on the road.

7. SYSTEM DESIGN

A helmet detection system is very important for keeping people safe in both traffic and construction areas. Many accidents happen because people do not follow safety rules and do not wear helmets. To reduce these accidents, an automatic system can help check and make sure that everyone is wearing a helmet.

This system uses image processing and Deep Learning technology to recognize people and check if they have a helmet on. It works by analyzing pictures or video footage from cameras to detect whether a person is following safety rules. If someone is not wearing a helmet, the system can record the information and alert the authorities.

By using this helmet detection system, traffic police can improve road safety for bike riders, and supervisors at construction sites can make sure workers are following safety guidelines. This will help reduce accidents, prevent serious head injuries, and even save lives. Automated helmet detection makes monitoring easier, faster, and more reliable compared to manual checking, ensuring better safety for everyone.

In addition to improving safety, an automated helmet detection system helps enforce rules more effectively. Traditional methods rely on manual checking, which can be slow, inconsistent, and prone to human errors. With this system, authorities can monitor a large number of people in real time without needing extra manpower. It can be used in busy construction sites, highways, and city roads to ensure compliance with helmet laws. The system can also store records of violations, making it easier to track repeat offenders and take necessary action. By adopting this technology, governments and organizations can create a safer environment, encourage responsible behavior, and reduce the risk of life-threatening accidents.

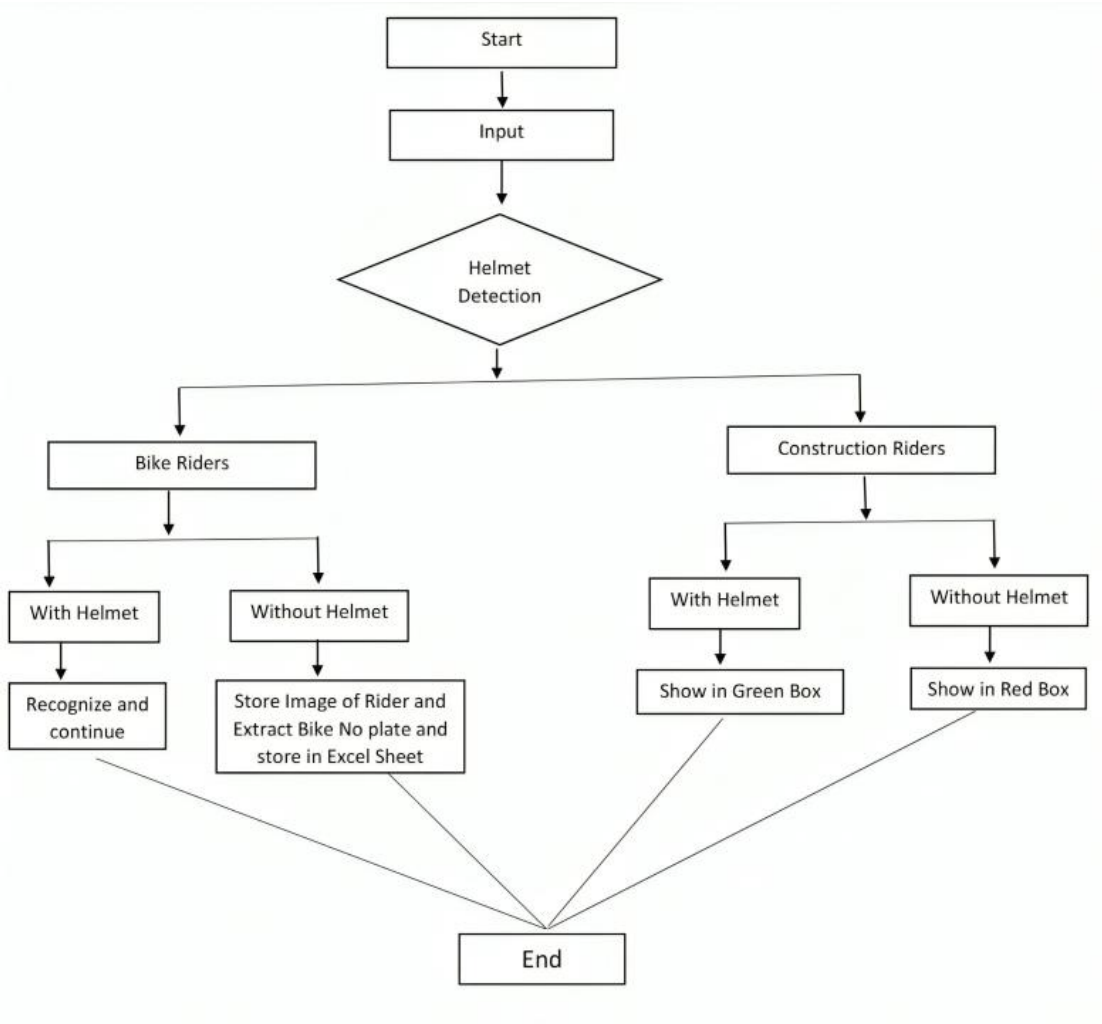


Fig 7.1: Dual-Purpose Helmet Detection System Flowchart

The above Fig 7.1 represents a helmet detection system designed to ensure safety compliance for both bike riders and construction workers. The system follows a structured approach to identify whether an individual is wearing a helmet and takes necessary actions accordingly.

The process begins with an input, which could be an image or video feed. The system then performs helmet detection using Deep Learning techniques to classify whether a helmet is present or not. Based on this detection, the flowchart divides individuals into two main categories: bike riders and construction workers.

For bike riders, if a helmet is detected, the system recognizes the rider and allows them to continue without any intervention. However, if a helmet is not detected, the system captures an image of the rider and extracts the bike's license plate number.

This information is then stored in an Excel sheet for further action, such as issuing fines or warnings to enforce helmet-wearing regulations.

On the other hand, for construction workers, the system highlights compliance using visual indicators. If a worker is detected wearing a helmet, their presence is marked with a green box, indicating adherence to safety standards. If a worker is found without a helmet, the system marks them with a red box, making it easier for supervisors to identify and take corrective action.

The process concludes after classifying all individuals and storing relevant data for further analysis or enforcement measures. This automated helmet detection system enhances safety monitoring by reducing manual inspection efforts and ensuring strict compliance with safety regulations in both traffic and construction environments.

8. IMPLEMEMTATION

workers.py:

 # detect whether workers are wearing helmets or not

```
import cv2
import pandas as pd
from ultralytics import YOLO
import cvzone
model = YOLO('best.pt')
cap = cv2.VideoCapture('helmet.avi')
my_file = open("coco1.txt", "r")
data = my_file.read()
class_list = data.split("\n")
count = 0
while True:
ret, frame = cap.read()
count += 1
if count % 3 != 0:
continue
if not ret:
break
frame=cv2.resize(frame,(1020,600))
results = model.predict(frame)
a = results[0].boxes.data
px = pd.DataFrame(a).astype("float")
for index, row in px.iterrows():
x1 = int(row[0])
y1 = int(row[1])
x2 = int(row[2])
y2 = int(row[3])
d = int(row[5])
c = class_list[d]
cv2.rectangle(frame, (x1, y1), (x2, y2), (0, 255, 0), 2)
cvzone.putTextRect(frame, f'{c}', (x1, y1), 1, 1)
```

```
cv2.imshow("FRAME", frame)
if cv2.waitKey(1) & 0xFF == 27:
break
cap.release()
cv2.destroyAllWindows()
```

bikeriders.py

```
# Detect whether Bike Riders are wearing helmets or not
import os
import streamlit as st
import base64
# from transformers import pipeline
from PIL import Image
from st_clickable_images import clickable_images
# pipeline = pipeline(task="image-classification", model="julien-c/hotdog-not-hotdog")
st.title("Helmet Detection App: ⛑")
confidence=st.slider("Confidence score (0-0.9)",
min_value=0.0,
max_value=.9,
value=.5,
step=.1)
file_name = st.file_uploader("Upload an image where workers may or may not wear helmets.")
if (file_name is not None):
col1, col2 = st.columns(2)
# img1 =""
# img2 =""
# col1.image(img1, use_column_width=True)
# col2.image(img2, use_column_width=True)
image = Image.open(file_name)
image.save("data/inputImg.jpg")
col1.image(image, use_column_width=True)
os.system(f"cd yolov5/ && python detect.py --weights ../best.pt --img 416 --conf
```

```
{confidence} --source ../data/inputImg.jpg")
im = Image.open("yolov5/runs/detect/exp/inputImg.jpg")
im.save("data/output.jpg")
im.close()
output =Image.open("data/output.jpg")
os.system("rm -rf yolov5/runs")
col2.image(output, use_column_width=True)
image.close()
output.close()
elif (st.checkbox('Select an example by clicking on the checkbox.')):
image_list=["https://media.istockphoto.com/id/1301722993/photo/group-of-
contractors-celebrating-the-end-of-successful-construction-
process.jpg?s=612x612&w=0&k=20&c=Xv7sO0AGHITZ6-
SdTFrvXYnlWQ_Sc3wAcnGory4n1NA=","https://media.istockphoto.com/id/147226
4590/photo/substation-maintenance-
engineers.webp?b=1&s=170667a&w=0&k=20&c=dXhPK_sakEhOsAFb6InX4onVo
aSgmUZ5A-V6eUlZf8E="]
# st.markdown("select a image")
clicked = clickable_images(
image_list,
titles=[f"Image #{str(i)}" for i in range(len(image_list))],
div_style={"display": "flex", "justify-content": "space-between", "flex-wrap":
"wrap"},
img_style={"margin": "5px", "height": "200px"},
key=None,
)
st.markdown(f"Image # {clicked} selected" if clicked > -1 else "No image selected")
if clicked>-1:
col1, col2 = st.columns(2)
file_exp =f"sample/image{clicked+1}.jpg"
image = Image.open(file_exp)
image.save("data/inputImg.jpg")
col1.image(image, use_column_width=True)
os.system(f"cd yolov5/ && python detect.py --weights ../best.pt --img 416 --conf
```

```
{confidence} --source ../data/inputImg.jpg")
im = Image.open("yolov5/runs/detect/exp/inputImg.jpg")
im.save("data/output.jpg")
im.close()
output =Image.open("data/output.jpg")
col2.image(output, use_column_width=True)
image.close()
output.close()
os.system("rm -rf yolov5/runs")
```

app.py

#model building

```
from flask import Flask, render_template, request, Response, redirect, url_for, flash
import os
import cv2
app = Flask(__name__)
app.secret_key = 'supersecretkey'
UPLOAD_FOLDER = 'static/uploads'
os.makedirs(UPLOAD_FOLDER, exist_ok=True)
app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
# Initialize camera
camera = cv2.VideoCapture(0)
@app.route('/')
def home():
return render_template('index.html')
@app.route('/helmet-detect', methods=['GET', 'POST'])
def helmet_detect():
if request.method == 'POST':
if 'video' in request.files:
video = request.files['video']
if video.filename != '':
filepath = os.path.join(app.config['UPLOAD_FOLDER'], video.filename)
video.save(filepath)
flash('Video uploaded successfully!', 'success')
return redirect(url_for('helmet_detect', video_url=filepath))
```

```
video_url = request.args.get('video_url')
return render_template('helmet_detect.html', video_url=video_url)
def generate_frames():
while True:
success, frame = camera.read()
if not success:
break
else:
# Draw an even larger blue square at the top center
height, width, _ = frame.shape
box_size = 300  # Increased size by half
start_x = (width // 2) - (box_size // 2)
start_y = 50  # Adjusted position for better visibility
end_x = start_x + box_size
end_y = start_y + box_size  # Making it square
# Draw the larger blue square
cv2.rectangle(frame, (start_x, start_y), (end_x, end_y), (255, 0, 0), 3)
# Add text "No Helmet"
cv2.putText(frame, "No Helmet", (start_x + 40, start_y + 50),
cv2.FONT_HERSHEY_SIMPLEX, 1, (255, 0, 0), 4)
# Convert the frame to JPEG format
ret, buffer = cv2.imencode('.jpg', frame)
frame = buffer.tobytes()
# Yield frame as multipart HTTP response
yield (b'--frame\r\n'
b'Content-Type: image/jpeg\r\n\r\n' + frame + b'\r\n')
@app.route('/video_feed')
def video_feed():
return Response(generate_frames(), mimetype='multipart/x-mixed-replace;
boundary=frame')
if __name__ == '__main__':
app.run(debug=True)
```

index.html

#home page

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Helmet Detection</title>
<link rel="stylesheet" href="{{ url_for('static', filename='styles.css') }}">
</head>
<body>
<div class="nav">
<a href="/" class="active">Home</a>
<a href="/helmet-detect">Helmet Detect</a>
<a href="#">About</a>
</div>
<div class="homeContent">
<h1>Welcome to Helmet Detection</h1>
<p>Upload your video or use the camera for real-time helmet detection.</p>
<a href="/helmet-detect" class="btn">Start Detection</a>
<div class="box one"></div>
<div class="box two"></div>
</div>
</body>
</html>
```

Helmet_detect.html

#home page

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Helmet Detection</title>
<link rel="stylesheet" href="{{ url_for('static', filename='styles.css') }}">
<script>
function showCamera() {
```

```
document.getElementById("cameraFeed").style.display = "block";
}
</script>
</head>
<body>
<!-- Navigation Bar -->
<div class="nav">
<a href="/">Home</a>
<a href="/helmet-detect" class="active">Helmet Detect</a>
<a href="#">About</a>
</div>
<h1 class="title">Helmet Detection System</h1>
<!-- Flash Messages -->
{% with messages = get_flashed_messages(with_categories=true) %}
{% if messages %}
<ul class="flash-messages">
{% for category, message in messages %}
<li class="{{ category }}">{{ message }}</li>
{% endfor %}
</ul>
{% endif %}
{% endwith %}
<div class="main">
<!-- File Upload Section -->
<div class="uploadSection">
<h2>Upload Video</h2>
<form action="{{ url_for('helmet_detect') }}" method="post"
enctype="multipart/form-data">
<input type="file" name="video" accept="video/*" id="videoInput">
<button type="submit" class="btn">Upload Video</button>
</form>
</div>
<!-- Live Camera Section -->
<div class="video-container">
```

```
<h2>Live Camera</h2>
<button onclick="showCamera()" class="btn">Connect to Camera</button>
<div id="cameraFeed" style="display: none;">
<img src="{{ url_for('video_feed') }}" width="500">
</div>
</div>
<!-- Uploaded Video Preview -->
{% if video_url %}
<div class="videoPreview">
<h2>Uploaded Video</h2>
<video width="500" controls>
<source src="{{ video_url }}" type="video/mp4">
Your browser does not support the video tag.
</video>
</div>
{% endif %}
</div>
</body>
</html>
```

style.css

#styling

```
/* Global Reset */
* {
padding: 0;
margin: 0;
box-sizing: border-box;
font-family: 'Poppins', sans-serif;
}
```

/* Smooth Gradient Background */

```
body {
background: linear-gradient(120deg, #2c3e50, #4ca1af);
color: #fff;
text-align: center;
padding: 50px;
```

}

/* Navigation Bar */

.nav {

display: flex;

justify-content: flex-end;

gap: 2rem;

align-items: center;

padding: 15px 50px;

}

.nav a {

color: white;

text-decoration: none;

font-size: 18px;

padding: 10px 15px;

transition: 0.3s;

border-bottom: 2px solid transparent;

}

.nav a:hover, .nav a.active {

border-bottom: 2px solid white;

}

/* Main Container */

.homeContent {

display: flex;

flex-direction: column;

align-items: center;

padding-top: 20vh;

}

.homeContent h1 {

font-size: 3rem;

margin-bottom: 10px;

}

.homeContent p {

font-size: 1.2rem;

margin-bottom: 20px;}

/* Buttons */

```
.btn {
background: #1abc9c;
color: white;
padding: 12px 25px;
font-size: 16px;
font-weight: bold;
border-radius: 5px;
border: none;
cursor: pointer;
transition: 0.3s ease-in-out;
}
.btn:hover {
background: #16a085;
}
/* Upload Section */
.uploadSection {
margin-top: 30px;
padding: 20px;
background: rgba(255, 255, 255, 0.1);
border-radius: 10px;
display: inline-block;
}
```

/* File Input */

```
input[type="file"] {
margin: 15px;
padding: 10px;
border-radius: 5px;
border: 2px solid white;
background: transparent;
color: white;
}
input[type="file"]::file-selector-button {
background: #3498db;
```

```
color: white;
padding: 8px;
border: none;
cursor: pointer;
}
input[type="file"]::file-selector-button:hover {
background: #2980b9;
}
/* Video Preview */
.videoPreview {
margin-top: 20px;
}
.videoPreview video {
margin-top: 15px;
border-radius: 10px;
box-shadow: 0px 5px 15px rgba(0, 0, 0, 0.3);
}
/* Background Decorations */
.box {
position: absolute;
width: 200px;
height: 200px;
background: rgba(255, 255, 255, 0.2);
border-radius: 50%;
z-index: -1;
}
.one {
top: 10vh;
left: 10vw;
}
.two {
bottom: 10vh;
right: 10vw;
```

9. SYSTEM TESTING

Test Case 1:

The Fig 9.1 represents a case where the helmet detection system identifies an individual without a helmet. The bounding box highlights the detected person, and a "No Helmet" label appears in blue text. The system can process both uploaded videos and live camera feeds to ensure safety compliance and can be further integrated with enforcement mechanisms for necessary actions

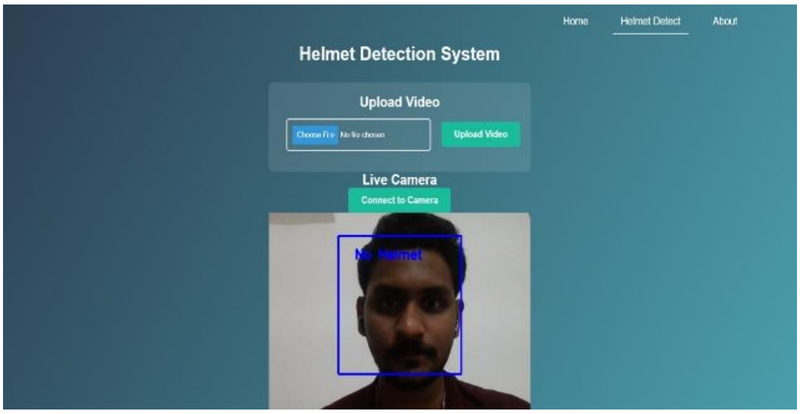


FIG 9.1: Helmet Detection Output Showing a Person Without a Helmet

Test Case 2:

Fig 9.2: Construction Worker Helmet Compliance

The Fig 9.2 shows a construction site where an AI-based helmet detection system verifies worker safety compliance. Each worker wearing a helmet is marked with a bounding box and a "helmet" label in pink. The system ensures safety regulations are followed in high-risk areas.

10.RESULT ANALYSIS

System Features:

Helmet Detection and Image Storage

- The system successfully identifies workers in the workplace and determines whether they are wearing helmets.
- Images of detected individuals are stored in designated folders based on their compliance status.
- This ensures that safety monitoring personnel can review and analyze helmet usage effectively.

Bike Rider Detection and License Plate Storage

- The system is capable of identifying helmet violations among bike riders.
- If a rider is found without a helmet, the system automatically captures an image of the bike's license plate.
- License plate numbers are stored in organized folders labeled with date and time for easy retrieval.
- Authorized personnel can review this data to track violations and enforce safety measures.
- The system also generates reports on the frequency of helmet violations, providing insights for improving road safety.

Visual Examples:

Helmet Violation Detection in Workplace and Roads

The system is designed to detect helmet violations in different environments, ensuring compliance with safety regulations.

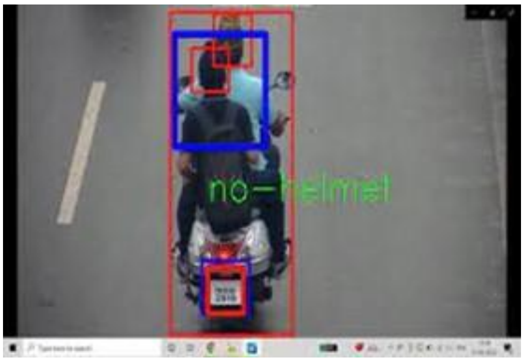


Fig 10.1: Rider Detected Without Helmet

- The Fig 10.1 demonstrates the system capturing a bike rider without a helmet.
- The detection is marked, and the license plate is extracted for further processing.

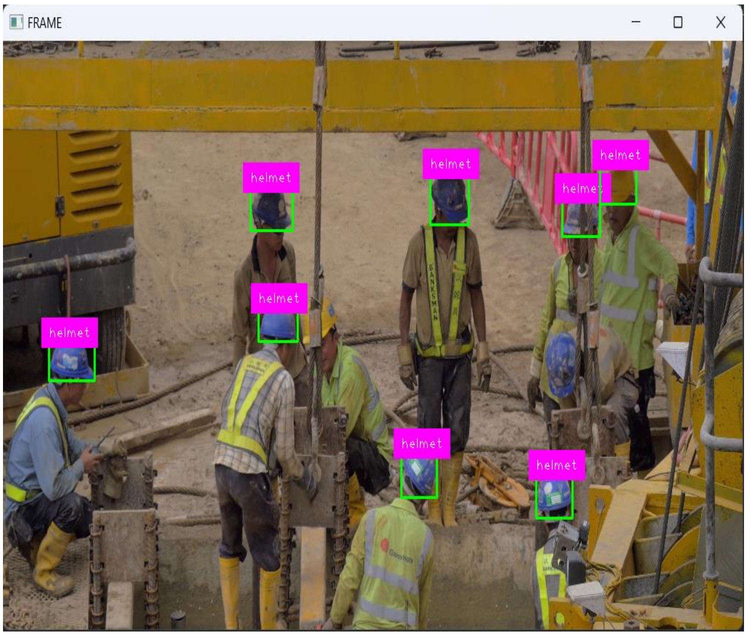


Fig 10.2: Identifying Workers with Safety Helmets in Various

- The Fig 10.2 represents Workers in different environments are analyzed to check for helmet usage.
- The system labels individuals correctly, allowing for precise safety monitoring.

Data Organization and Time Stamping

- The system captures and records key details, including images of bike riders and their license plate numbers.

- All data is stored in an Excel sheet, automatically labeled with the exact date and time of capture.
- This timestamping ensures that every event is accurately documented, aiding in safety compliance tracking.
- The structured format allows authorities to quickly review past records, spot patterns, and identify repeat offenders.

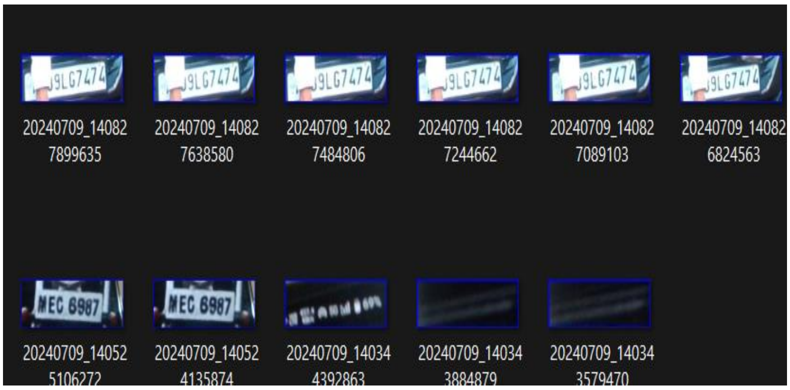
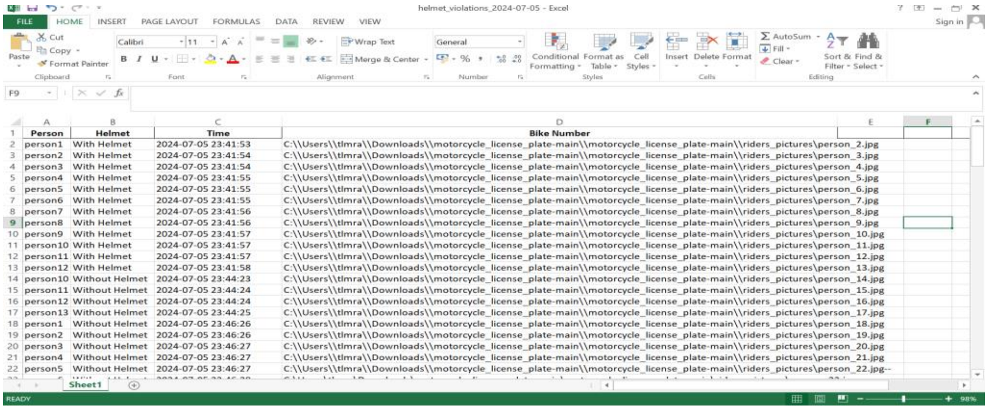


Fig 10.3: Detected Bike Plate Number

- The Fig 10.3 showcases how the system extracts and stores the number plate of a bike when a violation is detected.
- This data is crucial for tracking helmet rule compliance.



Person	Helmet	Time	Bike Number
person1	With Helmet	2024-07-05 23:41:53	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_2.jpg
person2	With Helmet	2024-07-05 23:41:54	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_3.jpg
person3	With Helmet	2024-07-05 23:41:54	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_4.jpg
person4	With Helmet	2024-07-05 23:41:55	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_5.jpg
person5	With Helmet	2024-07-05 23:41:55	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_6.jpg
person6	With Helmet	2024-07-05 23:41:55	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_7.jpg
person7	With Helmet	2024-07-05 23:41:56	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_8.jpg
person8	With Helmet	2024-07-05 23:41:56	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_9.jpg
person9	With Helmet	2024-07-05 23:41:57	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_10.jpg
person10	With Helmet	2024-07-05 23:41:57	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_11.jpg
person11	With Helmet	2024-07-05 23:41:57	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_12.jpg
person12	With Helmet	2024-07-05 23:41:58	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_13.jpg
person10	Without Helmet	2024-07-05 23:44:23	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_14.jpg
person11	Without Helmet	2024-07-05 23:44:24	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_15.jpg
person12	Without Helmet	2024-07-05 23:44:24	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_16.jpg
person13	Without Helmet	2024-07-05 23:44:25	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_17.jpg
person1	Without Helmet	2024-07-05 23:46:26	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_18.jpg
person2	Without Helmet	2024-07-05 23:46:26	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_19.jpg
person3	Without Helmet	2024-07-05 23:46:27	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_20.jpg
person4	Without Helmet	2024-07-05 23:46:27	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_21.jpg
person5	Without Helmet	2024-07-05 23:46:27	C:\\Users\\tlmra\\Downloads\\motorcycle_license_plate-main\\motorcycle_license_plate-main\\riders_pictures\person_22.jpg--

Fig 10.4: Helmet Violation Log With Rider and Number Plate Images Recorded

- In Fig 10.4 there is a structured log containing detected violations, including rider images and corresponding number plates.
- The log supports auditing and enforcement efforts by providing clear records of violation

11.USER INTERFACE

In Fig 11.1 it displays the Home Screen of the Helmet Detection System, providing users with a simple interface for real-time helmet detection. It allows users to upload videos or use a camera for detection, with easy navigation options.

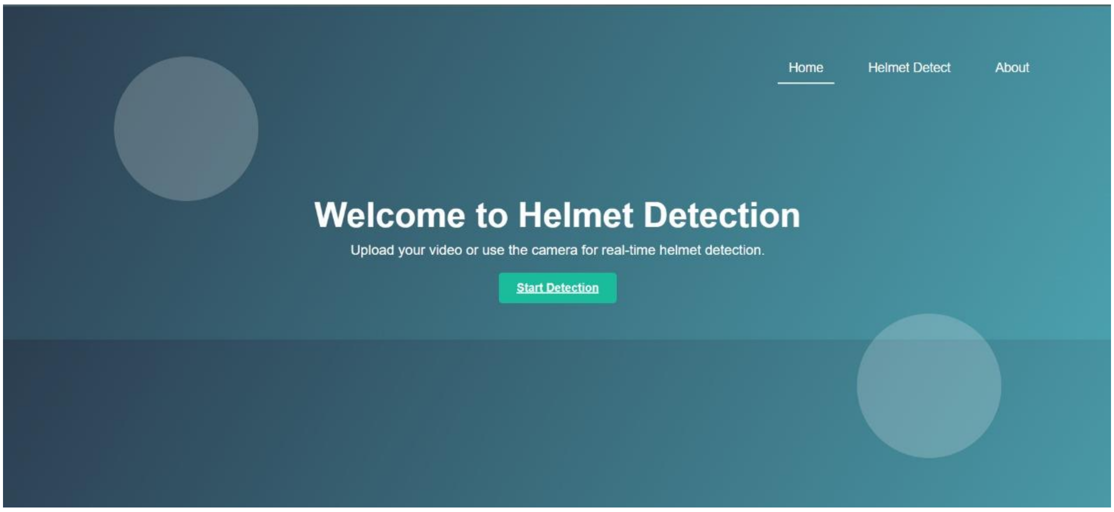


Fig 11.1: Home Screen

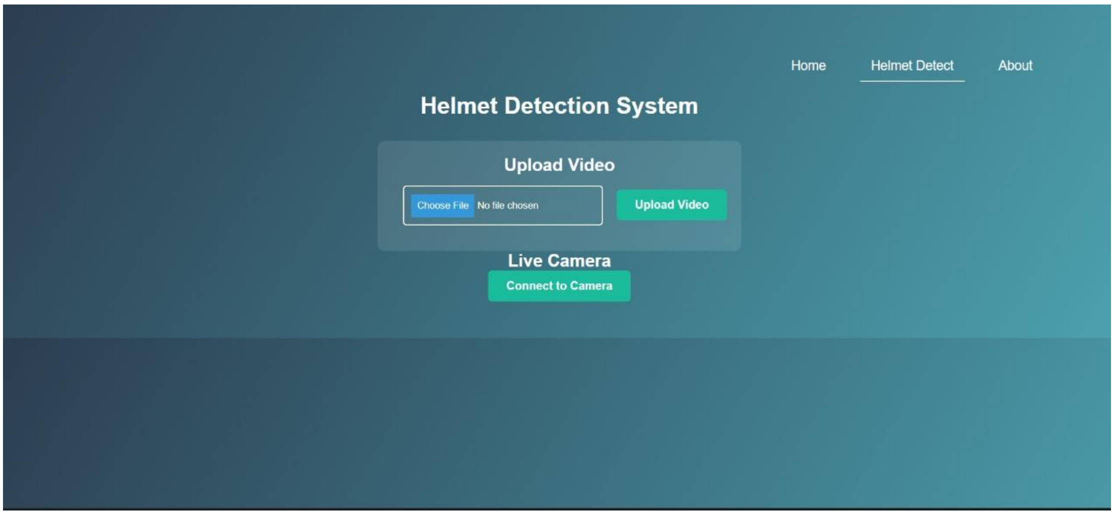


Fig 11.2: Helmet Detection Interface

In Fig 11.2 it displays the Helmet Detection Interface, allowing users to upload a video for helmet detection or connect to a live camera for real-time analysis. It provides a user-friendly experience with clear action buttons for both options.

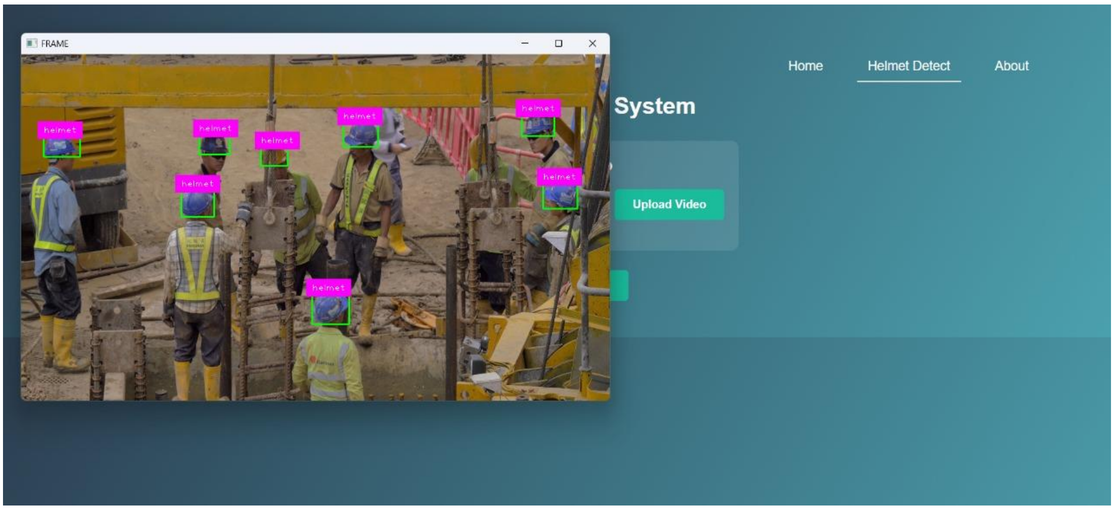


Fig 11.3: Helmet Detection Output

In Fig 11.3 it displays the Helmet Detection Output, where the system processes an uploaded video and identifies workers wearing helmets. Detected helmets are highlighted, ensuring safety compliance in the workplace.

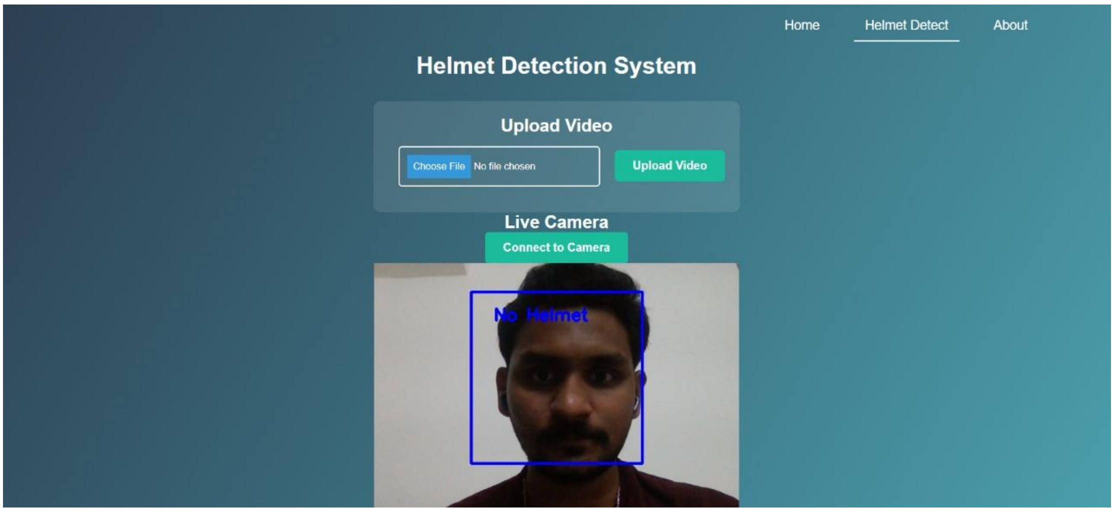


Fig 11.4: Live Camera Helmet Detection

In Fig 11.4 Helmet Detection System ensures real-time monitoring using a live camera to identify individuals wearing or not wearing helmets. This feature is crucial for enforcing safety regulations in workplaces such as construction sites, factories, and roads.

12. CONCLUSION

In this project, we developed an efficient, reliable, and user-friendly system for detecting whether individuals are wearing safety helmets. By leveraging Deep Learning and computer vision, our system ensures workplace safety by automatically identifying non-compliant workers and alerting the necessary personnel. To overcome the challenge of limited data, we applied targeted data augmentation techniques, such as rotation, scaling, and brightness adjustments, to enhance model performance. Additionally, we fine-tuned a pre-trained model to improve detection accuracy and minimize false positives. The system features an intuitive graphical user interface (GUI) that allows users to upload images or live video feeds, receive real-time helmet detection results, and access stored data seamlessly. It also automates the process of image and data storage, categorizing captured images of workers and bike riders in designated folders. If a bike rider is detected without a helmet, the system captures an image of the bike's license plate and logs it with the date and time, storing all data in an Excel sheet for easy retrieval and compliance tracking.

This approach significantly enhances operational efficiency by reducing the need for constant manual monitoring, as it alerts personnel only when a violation is detected, thereby saving time and effort. Looking ahead, we aim to further improve the system by collecting more diverse helmet usage images from various environments, incorporating new features to enhance detection accuracy, and ensuring stable performance when processing video frames. Additionally, we plan to optimize the model through techniques like pruning to reduce its size and improve usability. Future enhancements will also focus on implementing real-time alerts to provide instant reminders to workers, ultimately contributing to a safer work environment.

13. FUTURE SCOPE

The future of HelmSecure holds great potential for enhancing road and workplace safety through advanced AI-driven enforcement. By integrating real-time image processing and Machine Learning, the system can be improved to detect helmet compliance more accurately across various lighting, weather conditions, and complex scenarios such as crowded areas or high-speed motion. Future advancements may include facial recognition technology to identify repeat offenders and ensure accountability, automated fine issuance systems linked to law enforcement databases, and seamless integration with smart city infrastructure for enhanced monitoring and enforcement.

Additionally, incorporating IoT and cloud-based monitoring can provide real-time data transmission, enabling authorities to track violations efficiently and respond promptly. These cloud-based systems can store and analyze large volumes of data, offering insights into helmet compliance trends, high-risk zones, and accident-prone areas, ultimately assisting policymakers in implementing effective road safety measures. AI-driven predictive analytics can further help in forecasting potential violation hotspots, allowing preventive actions such as increased law enforcement presence or awareness campaigns in critical areas.

To improve accessibility and usability, HelmSecure can be enhanced with mobile applications that allow users to check their compliance status, receive real-time alerts, and access educational resources about helmet safety. Additionally, advancements in model optimization techniques, such as edge computing and AI model compression, can make the system more efficient and deployable on low-power devices, enabling widespread adoption in developing regions with limited computational resources.

With continuous AI model improvements, integration with vehicle registration databases, and partnerships with governmental transportation agencies, HelmSecure can play a crucial role in reducing non-compliance, preventing accidents, and promoting safer road habits. By leveraging cutting-edge technology, HelmSecure has the potential to become a key component of future intelligent transportation systems, fostering a culture of safety and responsibility among motorists and industrial workers alike.

14.REFERENCES

1. G. Gallo, F. D. Rienzo, F. Garzelli, P. Ducange, and C. Vallati, "A Smart System for Personal Protective Equipment Detection in Industrial Environments Based on Deep Learning at the Edge," in IEEE Access, vol. 10, pp. 110862-110878, 2022, doi: 10.1109/ACCESS.2022.3215148.
2. Saibabu, Chaganti., Dungi, Jagath Kumari., & Manthanwar, Dhananjay Chandrakant. (2022). Safety Management in Construction Industry: A Review. International Journal of New Innovations in Engineering and Technology, 20(2), 7-10. ISSN: 2319-6319(2).
3. Wang, Lili & Zhang, Xinjie & Yang, Hailu. (2023). Safety Helmet Wearing Detection Model Based on Improved YOLO-M. IEEE Access. PP. 1-1. 10.1109/ACCESS.2023.3257183.
4. Han, Kun., & Zeng, Xiangdong. (2022). Deep Learning-Based Workers Safety Helmet Wearing Detection on Construction Sites Using Multi-Scale Features. IEEE Access, 10, 718-729. doi: 10.1109/ACCESS.2021.3138407.
5. Jeong, Y. (2021). Self-Adaptive Motion Prediction-Based Proactive Motion Planning for Autonomous Driving in Urban Environments. IEEE Access, 9, 105612-105625. doi: 10.1109/ACCESS.2021.3100590.
6. Dalal, N., & Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1, 886–893.
7. Burges, C. J. C. (1998). A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery, 2(2), 121–167.
8. Nath, N.D., Behzadan, A.H., & Paal, S.G. (2020). Deep Learning for site safety: Real-time detection of personal protective equipment. Automation in Construction, 112, 103085. DOI: 10.1016/j.autcon.2020.103085.
9. Talib, Moahaimen & Al-Noori, Ahmed & Suad, Jameelah. (2024). YOLOv8-CAB: Improved YOLOv8 for Real-time object detection. Karbala International Journal of Modern Science, 10. 10.33640/2405-609X.3339.
10. Slimani, I., Zaarane, A., Al Okaishi, W., Atouf, I., & Hamdoun, A. (2020). An automated license plate detection and recognition system based on wavelet decomposition and CNN. Array, 8, 100040. https://doi.org/10.1016/j.array.2020.100040.

11. Omar, Naaman & Zeebaree, Subhi & M.Sadeeq, Mohammed & Zebari, Rizgar & Shukur, Hanan & Alkhayyat, Ahmed & Haji, Lailan & Kak, Shakir. (2023). License plate detection and recognition: A study of review. AIP Conference Proceedings, 050045. 10.1063/5.0170932.
12. Jibril, R. A., Utaminingrum, F., & Budia, A. S. (2021). Helmet monitoring system using Hough Circle and HOG based on KNN. Lontar Komputer, 12(1), 13–https://doi.org/10.24843/LKJITI.2021.v12.i01.p02.
13. Dalal, N., & Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 2005, pp. 886-893. doi: 10.1109/CVPR.2005.177.
14. Lin, H., Deng, J. D., Albers, D., & Siebert, F. W. (2020). Helmet Use Detection of Tracked Motorcycles Using CNN-Based Multi-Task Learning. IEEE Access, 8, 162073-162084. https://doi.org/10.1109/ACCESS.2020.3021357.
15. Sun, C. -Y., Hong, X. -J., Shi, S., Shen, Z. -Y., Zhang, H. -D., & Zhou, L. -X. (2021). Cascade Faster R-CNN Detection for Vulnerable Plaques in OCT Images. IEEE Access, 9, 24697-24704. https://doi.org/10.1109/ACCESS.2021.3056448.
16. B. Lin, "Safety Helmet Detection Based on Improved YOLOv8," IEEE Access, vol. 12, pp. 28260-28272, 2024. doi: 10.1109/ACCESS.2024.3368161.

CERTIFICATE

Springer

3rd International Conference on Power Engineering and Intelligent Systems

organized by

National Institute of Technology Uttarakhand, India

Certificate of Presentation

This is to certify that

Thotakura Bhuvanesh

has presented the paper titled **HelmSecure: AI Helmet Enforcement** authored by **S.N.Tirumala Rao, Sireesha Moturi, Thotakura Bhuvanesh, Kuchi Vinay, Dondapati Tharun Kumar, Sunitha Mothe** in the 3rd International Conference on Power Engineering and Intelligent Systems (PEIS 2025) held at **National Institute of Technology Uttarakhand, India** during March 08-09, 2025.

SCRS\PEIS2025\PC\22

Dr. Jagdish Chand Bansal
General Chair

Prof. Vivek Shrivastava
General Chair

https://scrs.in/conference/peis2025

HelmSecure: AI Helmet Enforcement

Dr.S.N.Tirumala Rao1, Sireesha Moturi2, Thotakura Bhuvanesh3, Kuchi Vinay3, Dondapati Tharun Kumar3, and Sunitha Mothe4

1 Professor and Head, Dept of Computer Science and Engineering
Narasaraopeta Engineering College(Autonomous),
Narasaraopet 522601, Palnadu District,Andhra Pradesh,India.
nagatirumalarao@gmail.com,

2 Associate. Professor, Dept of Computer Science and Engineering
Narasaraopeta Engineering College(Autonomous),
Narasaraopet 522601, Palnadu District,Andhra Pradesh,India.
sireeshamoturi@gmail.com

3 Dept of Computer Science and Engineering
Narasaraopeta Engineering College(Autonomous),
Narasaraopet 522601, Palnadu District,Andhra Pradesh,India.bhuvanesh1440@gmail.com, Vinaykuchi321@gmail.com,
tharunkumarchoudary3@gmail.com

4 Asst.Professor, Dept of Computer Science and Engineering
Narasaraopeta Engineering College(Autonomous),
Narasaraopet 522601, Palnadu District,Andhra Pradesh,India.msunitha973@gmail.com

Abstract:Ensuring the use of safety helmets is essential for reducing the risk of head injuries in both workplaces and on the road. However,workers and motorcycle riders often neglect wearing helmets due to discomfort or lack of awareness, leading to increased safety risks. To address this, a dual-purpose detection model has been developed to automatically detect helmet usage in construction sites and monitor motorcycle riders on the road. The system uses the YOLOv8 model to detect helmet usage, automatically capturing images, license plate numbers, and timestamps when helmets are not worn. The system also provides real time monitoring, enabling authorities to act swiftly on safety violations.The model was tested with a

diverse dataset, achieving high accuracy in detecting helmets, even in challenging conditions such as varying lighting and complex backgrounds. The results demonstrate that the system is highly efficient in monitoring helmet compliance, reducing the need for manual checks, and providing accurate, real-time data for enforcement.
Keywords: Helmet Detection ,Workplace Safety, Motorcycle Rider Monitoring, License Plate Recognition, Image Processing, Real-Time Detection, YOLOv8,Object Detection.

1.Introduction

With the fast growth of cities, there has been an increase in the urbanization rate which results in a greater need for infrastructure, resources and efficient Authors Suppressed Due to Excessive Length management systems and vast amount of infrastructure is needed to be built and construction site safety is gaining even more public attention than before.A lot of incidents can be stopped with the use of personal protective equipment(PPE). Security helmet is major proven PPEs that prevent labour from the blow of drop down construction materials or equipment, and wearing of safety helmets is a legal obligation around the world on building sites. However, safety helmet usage is often neglected due to discomfort and poor safety awareness.Monitoring workers' compliance with helmet regulations is crucial for safety and improved management at construction sites. Traditional compliance monitoring is widely used, but emerging computer vision technology offers great potential. Algorithms like the Gaussian Mixture Model (GMM) [5], Histogram of Oriented Gradient (HOG) [6], and Support Vector Machine (SVM) [7] have been developed for helmet detection. Video surveillance is the common traditional method for monitoring helmet compliance using camera footage [8].The continuous advancements in computer vision algorithms and computational power have paved the way for more efficient safety helmet identification methods. Among these, Deep Learning-based algorithms, particularly the YOLO (You Only Look Once) sequence, have attained a remarkable balance between speed,precision. Even state-of-the-art YOLObased methods face [9] in accurately detecting helmets, especially in complex environments where small,monochromatic helmets can be easily confused with other objects.Building upon these challenges, this study introduces a dualpurpose detection system designed to enhance safety in both industrial and vehicular contexts. Our model not only detects Whether a person has a helmet on or not in workplace but also extends its functionality to monitor motorcycle riders, verifying helmet usage and capturing license plate information [10]. This system automatically stores the detected individual's image, the motorcycle's license plate number,and timestamps these details in an Excel sheet, providing a comprehensive data record for safety enforcement.The license plate is a key identifier for a vehicle, giving it a unique identity.Recognizing license plates is a crucial aspect of developing smart transportation systems, which can help with traffic management, vehicle ¬Currently, license plate recognition technology involves three main steps: detection, segmentation, and recognition. However, this process can be complex, inefficient, and often affected by challenges like uneven lighting and noise, which reduces its reliability.In latest years, due to fastest advancement of Hardware infrastructure, neural network models based on Deep Learning they become the most suitable tools for tackling complex problems in computer vision. Convolutional Neural Network (CNN) [14]has shown to be best Deep Learning techniques for target detection and identification tasks, and most common algorithm for target identification using CNN is YOLO. As cities grow, construction safety becomes increasingly important. Using personal protective equipment (PPE) can prevent many accidents. such as helmets, which protect workers from falling objects. However, workers often skip wearing helmets due to discomfort and low awareness. Traditional helmet checks HelmSecure: AI Helmet Enforcement rely on manual supervision, but computer vision techniques, like YOLO, offer faster and more accurate detection. This study presents a system that not only checks helmet usage in construction but also monitors motorcyclists, recording license plates and timestamps in an Excel sheet, enhancing safety enforcement in various environments. Subsequent paragraphs, however, are indented.

2. LITERATURE SURVEY

A. Traditional Approaches to Helmet Detection Traditional helmet detection methods relied on manual feature extraction, using techniques like edge detection, color segmentation, and template matching to identify helmets. While simple, these methods often lacked accuracy, especially in complex environments with varied lighting and background clutter, leading to high false positive and negative rates. They struggled to detect small or partially obscured helmets, limiting their real-world effectiveness.One of the earliest techniques involved By utilizing The combination of Support Vector

Machines (SVM) and Histogram of Oriented Gradients (HOG) [12] has been widely utilized for object detection. While HOG effectively captured shape information, it was prone to errors when faced with objects that had similar edge features to helmets, leading to misclassifications. Another method,Circular Hough Transform (CHT) [13], was employed to detect circular shapes resembling helmets. However, its reliance on geometric assumptions made it less effective in dynamic and cluttered environments where helmet shapes could be distorted or obscured. Traditional helmet detection methods relied on manual feature extraction techniques such as edge detection and color segmentation. These approaches were useful in basic environments but struggled with accuracy in more complex conditions, like when lighting varied or backgrounds were cluttered. They often produced too many errors, with false positives and negatives, especially when helmets were small or partially hidden. As a result, these techniques were not practical for real-world use, where more advanced and reliable methods are needed to ensure proper helmet detection.

B.Advances in Deep Learning-Based Detection With the advent of deep learning, safety helmet detection has seen significant improvements in accuracy and robustness. Convolutional Neural Networks (CNNs) have revolutionized object detection by automatically learning hierarchical features from data, reduces the need for manual feature engineering. This shift has enabled more reliable detection in complex environments and has greatly improved performance in realtime applications. Deep Learning based methods are categorized in two main types: two-stage detectors and one-stage detectors. Two-stage detectors, like the Region-based Convolutional Neural Network (R-CNN) [15] family, first generate region proposals and then classify them. These methods have shown high accuracy but Authors Suppressed Due to Excessive Length often suffer from slower processing speeds, making them less suitable for real-time helmet detection. Improvements like Fast R-CNN and Faster R-CNN have attempted to address these issues by optimizing the region proposal process and sharing convolutional features, yet they still fall short in scenarios requiring rapid detection.

One-stage detectors like the YOLO series are popular for their ability to perform object detection in a single pass. The latest YOLOv8 offers impressive speed and accuracy, with enhancements like advanced feature fusion and lightweight architecture, making it ideal for real-time helmet detection. However, challenges persist in detecting small objects and handling occlusions in complex backgrounds.Currently, Deep Learning methods rely heavily on data, meaning the output of object detection algorithms based on the quality and quantity of the training datasets. However, there aren't many images available for safety helmet detection. To enhance the presentation of YOLO v5 in detecting helmet use, this research used techniques like Utilizing focused data augmentation and transfer learning methods. In the end, a userfriendly interface (GUI) was also designed to build the model simpler to utilize. The goal is to develop a model that quickly and accurately detects helmet use on construction sites while being easy to use and adaptable to various environments. The model must handle small objects and different helmet sizes effectively. The dataset includes diverse helmet images to ensure real-world performance. Enhancements to YOLO-based models, such as attention mechanisms and multi-scale feature fusion, improve detection of small and partially hidden helmets. Techniques like data augmentation and transfer learning further boost YOLOv8's performance, especially with limited datasets, and a user-friendly interface makes the model accessible to non-experts.

3.METHODOLOGY

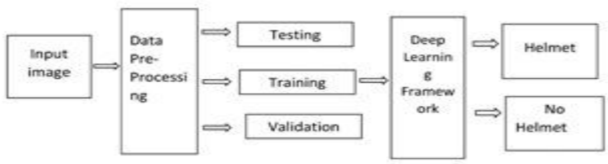


Fig. 1. Flow Diagram of Image Preprocessing and Deep Learning-Based Helmet Detection.

Figure 1 shows the helmet detection process using Deep Learning. It starts with an input image, which undergoes preprocessing (resizing, normalization,augmentation) for analysis. The image is then used in training, validation, and testing to develop and evaluate the model. YOLOv8 classifies images as either "Helmet" or "No Helmet." This process enhances accuracy in detecting and classifying helmet use, crucial for safety monitoring on construction sites.

A.Data Collection A diverse dataset is crucial for developing and testing a helmet detection model. It should include images from various workplaces and conditions, featuring both helmeted and non-helmeted subjects, with different lighting, camera angles, and challenging backgrounds like construction sites. The dataset should also account for varying heights, distances, partial occlusions, and helmet styles to ensure the model's robustness and adaptability in real-world scenarios.

i .Helmeted and Non-Helmeted Images: The proposed helmet detection required the collection of images-both helmeted and not-helmeted various types of environments, including construction sites, roads, and industrial areas. Data are compiled from various sources such as real footage captured by cameras in live circumstances and frames taken from video recordings. This technique would ensure the coverage of different practical scenarios on how individuals are situated in different contexts under various light conditions and within various environmental sets. Such diverse images will give a strong dataset in enhancing the model in detecting helmet usage from various sets of work or road safety conditions

Fig2: workers and Bike rider images with and without Helemt

2.Annotations: Each picture in the dataset is carefully annotated to indicate the presence and location of helmets. This process involves drawing bounding boxes around helmets and labeling them correctly. To ensure accuracy, annotations are either done manually by humans or with the help of semi-automatic tools. This precise labeling process is crucial for training the detection model, as it helps the algorithm learn to identify helmets accurately in various scenarios and environments. Proper annotations contribute significantly to the model's overall output and effectiveness in real-world applications.

Diversity and Balance: To balance your dataset, aim to have a similar number of helmeted and non-helmeted images. Include a variety of helmet styles, colors, and sizes to reflect real-world diversity. Collect diverse images, label them appropriately, and apply Data augmentation techniques mitigate potential imbalance. Ensure the dataset is evenly split into training, validation, and test sets, and monitor the model's performance to confirm it generalizes well across all variations.

B. Data Preprocessing

Preprocessing is essential to prepare the collected data for training and evaluation. This step involves several key operations to enhance data quality and ensure that the model learns effectively.

Image Resizing : To ensure that all images are uniform and can be processed quickly during training, they are resized to a standard size. This helps in maintaining consistency across all images. For many image processing models, common sizes used are 416x416 pixels or 640x640 pixels. These sizes are chosen based on what the model needs to work efficiently. Resizing images to these dimensions helps the model learn better and speeds up the training process.

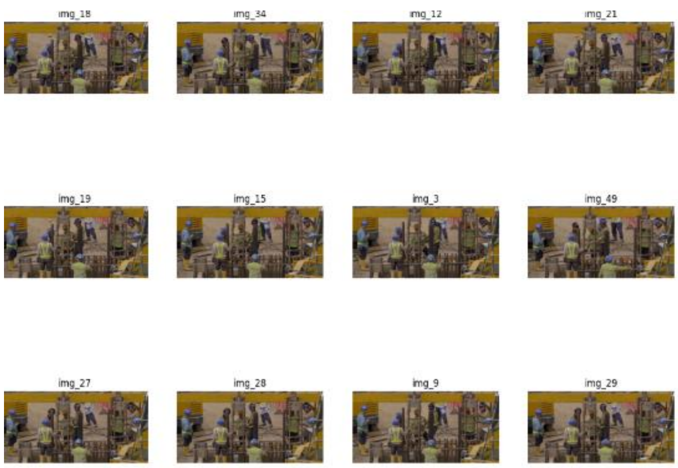


Fig3: Sample images for helmet detection at a construction site.

1. **Data Augmentation**: To build the model more reliable, data augmentation techniques are used. This involves making changes to the images, such as rotating them randomly, scaling them up or down, shifting their position, and adjusting their colors. By applying these changes, the model is exposed to different scenarios and conditions, which helps it learn better and perform well

in various situations. This process improves the prototype capability to handle new and unseen data effectively.

2. **Normalization**: Pixel values are adjusted to fit within a range of [0, 1] or [-1, 1] to help the model train more effectively and quickly. This process, known as normalization, involves scaling the pixel values from their original range to this new range. Normalizing the pixel values ensures that the training is stable and helps the model learn more efficiently.
3. **Splitting Dataset**: The dataset is split divided in three parts: training, validation, and testing. Usually, 70% of the data is used for training the model, 15% is set aside for validation during training to tune and improve the model, and the remaining 15% is used for testing to evaluate the model's performance on new, unseen data. This division helps ensure that the model learns effectively and can generalize well to new data.

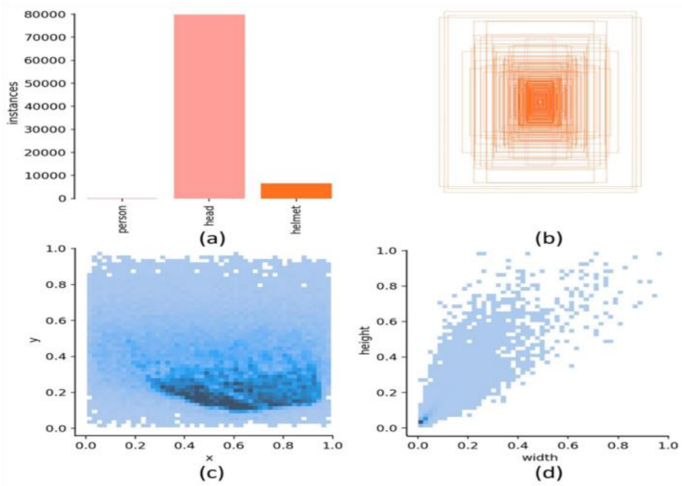


Fig4:Visualization results of the training data.
(a)Histogram illustrating the number of instances for each category;
(b) Distribution of bounding boxes for all data;
(c) Histogram of x and y variables, displaying the spatial distribution of the dataset;
(d) Histogram of width and height variables illustrating the dataset's distribution.

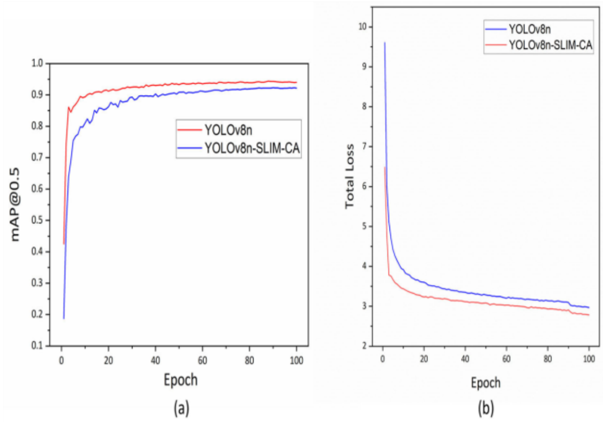


Fig5:Comparison of MAP and loss between YOLOv8n,YOLOv8n-SLIM-CA.

(a) mAP(%). (b) loss.

C. Model Description

The model utilized for helmet detection is based on the YOLOv8 architecture, optimized for improved performance in detecting helmets in various conditions.

1. **Base Architecture**: YOLOv8 is a modern object detection algorithm known for its high speed and accuracy. It operates using a single-stage detection approach, which simplifies the network and reduces processing time compared to two-stage detectors.
2. **Enhanced YOLOv8 Model**:

   Slim-Neck Feature Fusion: The YOLOv8 model is modified to incorporate a Slim-Neck structure. This lightweight [16] fusion network improves the model's efficiency by reducing the number of parameters while maintaining detection performance.

   Coordinate Attention Mechanism (CA): An attention mechanism is introduced to enhance the model's attention on helmet regions. This mechanism helps the model better localize helmets by incorporating spatial and channel-wise information.

   Small Target Detection Layer: An additional detection head is added to improve the prototype ability to remove small and partially obscured helmets. This layer helps in better localization and classification of small targets.

3.Training:

The model is trained on the annotated dataset with a resulting presentation performance. combination of loss functions tailored for object detection, such as the focal loss and the Intersection over Union (IoU) loss. Training involves multiple epochs and uses optimization algorithms like Adam or SGD (Stochastic Gradient Descent) to minimize the loss.

Evaluation Metrics: Model performance is evaluated using metrics such as precision, recall, mean Average Precision (mAP), and F1-score. These metrics help in assessing the accuracy and effectiveness of the helmet detection model.

1. **Implementation Tools**: The model is implemented using popular Deep Learning frameworks like TensorFlow. These frameworks provide the necessary tools for building, training, and evaluating the model efficiently.

The flow chart shown at the beginning illustrates the detection process of motorcycle riders without helmets based on legal requirements using machine learning models. The video input will be real-time footage collected from the camera, which is stored temporarily. Then, an object detection model such as YOLOv8 is used for detecting motorcycles and obtaining their bounding boxes. In a similar way, a YOLO model detects helmets. When the violation occurs(e.g., when the helmet is not used), it detects the motorcycle and later detects the license plate. The characters found on the license plate and images of these violations detected will be appropriately stored in a folder to review later.

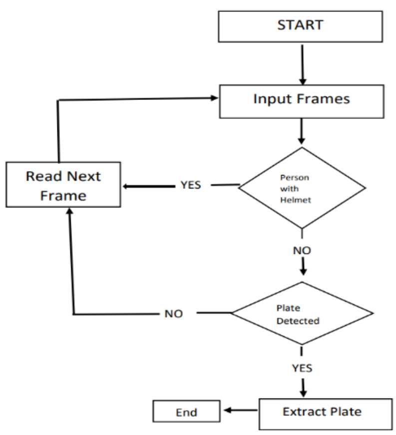


Fig6: Extraction Number Plate Flowchart

In Fig6 it shows the step-by-step decision-making process for helmet usemonitoring. The system takes video frames as input and checks for the wearing of the helmet by the person in the frame. If a helmet is being worn, it reads the next frame. If there is not a helmet being worn, then the system checks if the license plate of a motorcycle has been detected, checking next steps fordetection. After the license plate number has been detected sequentially after having detected are there, they are to be extracted last. If none of the aboveconditions are met to process the helmet detection or license plate read, then the system again reads the next frame in sequence.

4.Result

In this project, we developed a comprehensive safety monitoring system that notonly detects individuals and bike riders but also organizes and stores relevant data systematically. Here are the key results:

1.Detection and Image Storage: The system successfully identifies individuals in the workplace and stores their images in designated folders. This functionality ensures that personnel who are or are not wearing helmets can be monitored effectively.

Fig7: Identifying Workers with Safety Helmets in Various Scenarios

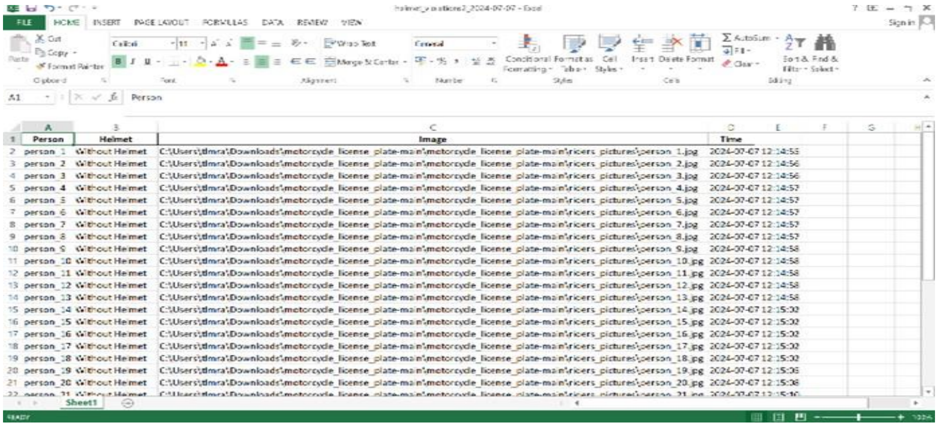
Fig8: Rider Detected Without Helmet

2.Bike Rider Detection and License Plate Storage: The system is designed to detect whether bike riders are wearing helmets. If a rider is found without a helmet, it automatically captures an image of the bike's license plate. The license plate numbers are stored in folders organized by date and time, making it easy to retrieve this information later. Authorized personnel can access these folders to review the data and take appropriate action. Additionally, the system can generate reports that show how frequently this occurs, helping to improve overall safety.

Fig 9: Detected Bike plate number

3.Data organization and Time Stampping: The system captures and records key details, such as images of bike riders and their license plate numbers. All of this data is stored in an Excel sheet, with each entry automatically labeled with the exact date and time it was captured. This timestamping ensures that every event is accurately documented.

By organizing the data in this way, the Excel sheet becomes a powerful tool for monitoring helmet compliance. It allows for easy tracking of when and how often helmet violations occur, making it simple to spot patterns or identify repeat offenders. The well-organized format also supports efficient auditing, enabling authorities to quickly review past records and make informed decisions to improve safety measures.



	Person	Helmet	Image	Time
2	person_1	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_1.jpg	2024-07-07 12:14:55
3	person_2	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_2.jpg	2024-07-07 12:14:56
4	person_3	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_3.jpg	2024-07-07 12:14:56
5	person_4	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_4.jpg	2024-07-07 12:14:57
6	person_5	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_5.jpg	2024-07-07 12:14:57
7	person_6	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_6.jpg	2024-07-07 12:14:57
8	person_7	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_7.jpg	2024-07-07 12:14:57
9	person_8	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_8.jpg	2024-07-07 12:14:57
10	person_9	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_9.jpg	2024-07-07 12:14:58
11	person_10	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_10.jpg	2024-07-07 12:14:58
12	person_11	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_11.jpg	2024-07-07 12:14:58
13	person_12	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_12.jpg	2024-07-07 12:14:58
14	person_13	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_13.jpg	2024-07-07 12:14:58
15	person_14	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_14.jpg	2024-07-07 12:15:02
16	person_15	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_15.jpg	2024-07-07 12:15:02
17	person_16	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_16.jpg	2024-07-07 12:15:02
18	person_17	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_17.jpg	2024-07-07 12:15:02
19	person_18	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_18.jpg	2024-07-07 12:15:02
20	person_19	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_19.jpg	2024-07-07 12:15:36
21	person_20	Without Helmet	C:\Users\tlmra\Downloads\motorcycle_license_plate-main\motorcycle_license_plate-main\ricers_pictures\person_20.jpg	2024-07-07 12:15:38

Fig10: Helmet violation log with rider and number plate images recorded

5.Conclusion

In this analysis, we developed a Efficient, reliable, and easy-to-use method for detecting whether individuals are wearing safety helmets. To address the challenge of limited data, we applied Applied targeted data augmentation and leveraged a pre-trained model. Additionally, we designed a user-friendly graphical user interface (GUI) to facilitate practical use in real-world applications. The model was tested on a dataset that included various challenges such as scale variations and small objects. Our method is more efficient than traditional manual monitoring, as it alerts personnel only when a labour is not wearing a helmet, reducing the need for constant surveillance and saving time and energy. Looking ahead, we plan to enhance the model by collecting more diverse helmet usage images in different environments, exploring new features to improve its performance, and ensuring stable detection when processing video frames. We also want to make the model smaller and easier to use by applying methods like model pruning, which helps reduce its size. Lastly, we will keep improving the system so it can give quick reminders to workers on construction sites, helping to make the workplace safer.

References

[1] G. Gallo, F. D. Rienzo, F. Garzelli, P. Ducange, and C. Vallati, "A Smart System for Personal Protective Equipment Detection in Industrial Environments Based on Deep Learning at the Edge," in IEEE Access, vol. 10, pp. 110862-110878, 2022.

[2] Saibabu, Chaganti., Dungi, Jagath Kumari., & Manthanwar, Dhananjay Chandrakant. (2022). Safety Management in Construction Industry: A Review. International Journal of New Innovations in Engineering and Technology, 20(2), 7-10.

[3] Wang, Lili & Zhang, Xinjie & Yang, Hailu. (2023). Safety Helmet Wearing Detection Model Based on Improved YOLO-M. IEEE Access.

[4] Han, Kun., & Zeng, Xiangdong. (2022). Deep Learning-Based Workers Safety Helmet Wearing Detection on Construction Sites Using Multi-Scale Features. IEEE Access, 10, 718-729.

[5] Jeong, Y. (2021). Self-Adaptive Motion Prediction-Based Proactive Motion Planning for Autonomous Driving in Urban Environments. IEEE Access, 9, 105612-105625.

[6] Dalal, N., & Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1, 886–893.

[7] Burges, C. J. C. (1998). A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery, 2(2), 121–167.

[8] Nath, N.D., Behzadan, A.H., & Paal, S.G. (2020). Deep Learning for site safety: Real-time detection of personal protective equipment. Automation in Construction, 112, 103085.

[9] Talib, Moahaimen & Al-Noori, Ahmed & Suad, Jameelah. (2024). YOLOv8-CAB: Improved YOLOv8 for Real-time object detection. Karbala International Journal of Modern Science.

[10] Slimani, I., Zaarane, A., Al Okaishi, W., Atouf, I., & Hamdoun, A. (2020). An automated license plate detection and recognition system based on wavelet decomposition and CNN. Array, 8, 100040.

[11] Omar, Naaman & Zeebaree, Subhi & M.Sadeeq, Mohammed & Zebari, Rizgar & Shukur, Hanan & Alkhayyat, Ahmed & Haji, Lailan & Kak, Shakir. (2023). License plate detection and recognition: A study of review. AIP Conference Proceedings, 050045.

[12] Jibril, R. A., Utaminingrum, F., & Budia, A. S. (2021). Helmet monitoring system using Hough Circle and HOG based on KNN. Lontar Komputer, 12(1), 13–23.

[13] Dalal, N., & Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 2005, pp. 886-893.

[14] Lin, H., Deng, J. D., Albers, D., & Siebert, F. W. (2020). Helmet Use Detection of Tracked Motorcycles Using CNN-Based Multi-Task Learning. IEEE Access, 8, 162073-162084.

[15] Sun, C. -Y., Hong, X. -J., Shi, S., Shen, Z. -Y., Zhang, H. -D., & Zhou, L. -X. (2021). Cascade Faster R-CNN Detection for Vulnerable Plaques in OCT Images. IEEE Access, 9, 24697-24704.

[16] B. Lin, "Safety Helmet Detection Based on Improved YOLOv8," IEEE Access, vol. 12, pp. 28260-28272, 2024.

cb2 (2).docx

ORIGINALITY REPORT

7%	5%	6%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.mlrjournal.org Internet Source	1%
2	Kun Han, Xiangdong Zeng. "Deep Learning-Based Workers Safety Helmet Wearing Detection on Construction Sites Using Multi-Scale Features", IEEE Access, 2022 Publication	1%
3	Submitted to Fachhochschule Nordwestschweiz Student Paper	1%
4	Pawan Singh Mehra, Dhirendra Kumar Shukla. "Artificial Intelligence, Blockchain, Computing and Security - Volume 2", CRC Press, 2023 Publication	1%
5	Submitted to Trinity College Dublin Student Paper	1%
6	www.journaltocs.ac.uk Internet Source	<1%
7	ssl.linklings.net Internet Source	

<1%

8	Submitted to Army Institute of Technology Student Paper	<1%
9	Submitted to University of Central England in Birmingham Student Paper	<1%
10	arxiv.org Internet Source	<1%
11	Bubryur Kim, Eui-Jung An, Sungho Kim, K. R. Sri Preethaa, Dong-Eun Lee, R. R. Lukacs. "SRGAN-enhanced unsafe operation detection and classification of heavy construction machinery using cascade learning", Artificial Intelligence Review, 2024 Publication	<1%
12	Submitted to University of Greenwich Student Paper	<1%
13	pure.uva.nl Internet Source	<1%
14	Bingyan Lin. "Safety Helmet Detection Based on Improved YOLOv8", IEEE Access, 2024 Publication	<1%
15	Smruti Rekha Das, Debahuti Mishra, Minakhi Rout. "A hybridized ELM-Jaya forecasting model for currency exchange prediction",	<1%

Journal of King Saud University - Computer and Information Sciences, 2017
Publication

16	export.arxiv.org Internet Source	<1%
17	mdpi-res.com Internet Source	<1%
18	www.mdpi.com Internet Source	<1%
19	s3-eu-west-1.amazonaws.com Internet Source	<1%

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		