RV College of Engineering®
Mysore Road, RV Vidyaniketan Post,
Bengaluru - 560059, Karnataka, India

Go, change the world®

DEPARTMENT OF

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Project Report

On

Enhancing Railway Safety Through Human Activity Recognition

Submitted in partial fulfilment of the requirements for the V Semester

ARTIFICIAL NEURAL NETWORK AND DEEP LEARNING

AI253IA

By

1RV22AI047	**Kushaal S**
1RV22AI052	**Shiva Kumar**

Department of Artificial Intelligence and Machine Learning
RV College of Engineering®
Bengaluru – 560059

2024 – 25

RV COLLEGE OF ENGINEERING®

(Autonomous Institution Affiliated to Visvesvaraya Technological University, Belagavi)

DEPARTMENT OF

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Bengaluru– 560059

CERTIFICATE

This is to certify that the project entitled "Enhancing Railway Safety Through Human Activity Recognition" submitted in partial fulfillment of Artificial Neural Networks and Deep Learning (AI253IA) of 5^{th} Semester BE is a result of the bonafide work carried out by Kushaal S (1RV22AI047) and Shiva Kumar (1RV22AI052) during the Academic year 2024-25

Faculty In charge

Date :

Head of the Department

Date :

RV COLLEGE OF ENGINEERING®

(Autonomous Institution Affiliated to Visvesvaraya Technological University, Belagavi)

DEPARTMENT OF
ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Bengaluru– 560059

DECLARATION

We, Kushaal S (1RV22AI047) and Shiva Kumar (1RV22AI052), students of fifth Semester BE hereby declare that the Project titled **“Enhancing Railway Safety Through Human Activity Recognition**” has been carried out and completed successfully by us and is our original work.

Date of Submission: **Signature of the Student**

ACKNOWLEDGEMENT

We are profoundly grateful to our guide, *Dr. Somesh Nandi*, Assistant Professor, RV College of Engineering, for his wholehearted support, valuable suggestions, and invaluable advice throughout the duration of our project. His guidance and encouragement were instrumental not only in the successful completion of the project but also in the preparation of this report. We also extend our special thanks to *Dr. Anupama Kumar* for her invaluable insights, support, and constructive feedback, which significantly contributed to the improvement of our work.

We would like to express our sincere thanks to our Head of the Department, *Dr. Satish Babu*, Head of Department AIML, RV College of Engineering for his constant encouragement and for fostering an environment of innovation and learning that greatly aided our progress.

We extend our heartfelt gratitude to our beloved Principal, *Dr. K. N. Subramanya*, for his unwavering appreciation and support for this Experiential Learning Project, which motivated us to give our best.

Lastly, we take this opportunity to thank our family members and friends for their unconditional support and encouragement throughout the project. Their backup and motivation were crucial in helping us overcome challenges and successfully complete our work.

ABSTRACT

Railway safety remains a critical concern, particularly in densely populated regions where unauthorized human presence on tracks leads to frequent accidents and fatalities. India, home to one of the largest and busiest railway networks globally, faces thousands of track-related casualties annually due to accidental trespassing. Conventional safety mechanisms such as manual surveillance, physical barriers, and CCTV monitoring have proven insufficient due to their high operational costs, limited coverage, and lack of real-time response capabilities. The rapid advancements in Artificial Intelligence and computer vision present an opportunity to address these challenges through automated monitoring and detection systems. This project aims to develop an intelligent railway safety system leveraging state-of-the-art deep learning techniques to detect and recognize human presence on railway tracks in real time, enabling proactive accident prevention. At the core of this system is YOLO (You Only Look Once), a cutting-edge object detection model renowned for its exceptional speed and accuracy. Unlike traditional object detection architectures such as R-CNN and Fast R-CNN, which involve multiple computationally expensive steps, YOLO employs a single-pass detection structure, making it highly suitable for real-time applications in dynamic environments. Its advanced architecture ensures efficient detection across varying conditions, including changes in lighting, weather, and occlusions, making it an optimal choice for railway monitoring. By integrating AI-driven automation into railway safety, this project addresses the shortcomings of existing monitoring systems and provides a scalable, cost-effective solution for real-time threat detection and alert generation.

The project is structured into four interlinked stages, starting with data collection and preprocessing, where a high-quality dataset is prepared using Roboflow for annotation and Python libraries such as OpenCV, NumPy, and Pandas for preprocessing and augmentation. A diverse dataset is curated to enhance model generalization across different environmental conditions. The training process is optimized to maximize detection accuracy using key performance metrics such as precision, recall, and mean average precision (mAP). Performance metrics such as false positive rates, detection latency, and robustness under occlusions and varying lighting conditions are analyzed to ensure reliable operation. Finally, the deployment and real-time integration stage incorporates the trained model into a real-time inference pipeline using Python and OpenCV, enabling continuous monitoring of live video feeds for immediate detection of unauthorized human presence.

By leveraging AI-powered automation, this solution not only improves railway safety and operational efficiency but also demonstrates the broader applicability of deep learning models in Intelligent Transportation Systems. The results of this study will contribute valuable insights into the feasibility of computer vision-based railway monitoring, paving the way for future advancements in AI-driven safety solutions.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

This chapter provides a comprehensive overview of the project on Enhancing Railway Safety Through Human Activity Recognition, focusing on the motivation, objectives, and significance of the study. It discusses the fundamental theories and concepts that form the basis of the proposed system, including deep learning techniques and object detection methodologies. The chapter also presents the overall structure of the report, detailing the organization of subsequent chapters for a clear and systematic understanding of the work.

1.1 Project Description

The project, Enhancing Railway Safety Through Human Activity Recognition, focuses on addressing critical safety concerns in the railway sector by employing cutting-edge deep learning and computer vision techniques. Railway tracks are often prone to human intervention, which can lead to severe accidents and operational disruptions. This project harnesses the power of YOLO, You Only Look Once, an advanced object detection algorithm celebrated for its exceptional speed and accuracy, to analyze live video feeds from surveillance cameras installed along railway tracks. The system is designed to detect the presence of humans and recognize specific activities, such as walking, standing, sitting, or lying on the tracks, that could pose potential hazards. By identifying these activities in real time, the system generates instant alerts to railway operators, enabling swift action to prevent accidents. This innovative approach not only enhances the safety of railway operations but also underscores the importance of integrating artificial intelligence and machine learning technologies in critical infrastructure to reduce human errors and save lives.

The primary objective of this project is to enhance railway safety by developing a system capable of accurately detecting and classifying human activities on or near railway tracks, using the YOLO deep learning model. With real-time monitoring as a central focus, the system analyzes live video feeds to promptly identify potential hazards and ensure swift intervention. The project aims to minimize accidents caused by human intervention on tracks by implementing an automated alert mechanism that notifies railway operators immediately. By leveraging YOLO model's high accuracy and speed, the system is designed to perform effectively even in challenging conditions, such as low light or crowded environments. Scalability is a key consideration, allowing for seamless deployment across diverse railway sites and adaptation to varying geographical and environmental conditions. The solution also aims to reduce reliance on manual monitoring, thereby decreasing the risk of human error and improving operational efficiency. Moreover, the project intends to integrate the system with existing railway management frameworks to ensure comprehensive safety coverage while promoting the adoption of AI-driven solutions in public infrastructure. Ultimately, the project not only addresses current safety challenges but also supports preventive measures by generating insights to guide long-term policy and strategy development for railway operations.

Key Objectives are,

- Enhance railway safety through human activity detection.
- Enable real-time monitoring and automated alerts.
- Achieve high accuracy and speed with YOLO models.
- Reduce reliance on manual monitoring to minimize errors.
- Promote AI adoption and support preventive measures.

The project "Enhancing Railway Safety Through Human Activity Detection" leverages advanced theories and concepts from deep learning, computer vision, and safety engineering. At its core, it utilizes You Only Look Once (YOLO), a state-of-the-art object detection algorithm, which combines high accuracy with real-time performance. YOLO works by dividing an image into a grid and predicting bounding boxes and class probabilities simultaneously, making it ideal for applications requiring quick and reliable detection of objects or activities. This capability is crucial for real-time railway monitoring, where detecting human presence on or near tracks must happen with minimal latency.

The project also incorporates principles from human activity recognition (HAR), which is the task of identifying human actions or behaviors through data analysis. HAR is widely used in safety systems, healthcare, and surveillance. In this case, the focus is on recognizing actions like crossing tracks, loitering, or unauthorized access to restricted areas. Deep learning models for HAR often utilize convolutional neural networks (CNNs) for feature extraction and classification, a methodology that YOLO inherently employs.

From a safety engineering perspective, the project aligns with the Hazard Prevention and Control Framework, which involves identifying potential risks, analyzing their likelihood and impact, and implementing solutions to mitigate them. Real-time detection and alerts enhance the operational response to hazards, minimizing the chances of accidents and injuries.

Additional concepts include,

- Feature Extraction and Classification: YOLO extracts spatial and temporal features from video streams, identifying specific human activities.
- Real-Time Processing: Utilizing GPUs or specialized hardware to ensure efficient handling of high-resolution image data.
- Integration with Railway Infrastructure: Adapting the detection system to work seamlessly with existing control systems and surveillance networks.
- Scalability and Deployment: Addressing challenges related to implementing AI models in diverse environments, such as urban, rural, and extreme weather conditions.
- By uniting these theories and concepts, the project exemplifies how cutting-edge AI techniques can be harnessed to address real-world safety challenges effectively.

1.2 Report Organization

The report is structured to provide a comprehensive understanding of the project "Enhancing Railway Safety Through Human Activity Recognition." It begins with the Introduction, which presents an overview of the project's background, significance, and objectives, highlighting the necessity of AI-driven surveillance in railway safety.

The Project Description elaborates on the scope, methodologies, and anticipated outcomes, setting the foundation for the subsequent sections. The Report Organization section guides readers through the structured layout of the report, ensuring clarity and logical progression.

Following this, the Literature Review explores existing research, current railway safety measures, and advancements in human activity recognition, discussing the limitations of conventional systems and the innovations introduced in this project. It also details the tools, techniques, and technologies employed, along with the hardware and software requirements. The Software Requirement Specifications section defines the system’s functional and non-functional requirements, external interfaces, and design constraints, providing a clear understanding of the system's capabilities and limitations.

The System Design section presents the architectural framework, data flow diagrams, and a detailed explanation of the algorithms and deep learning models used, with a focus on the object detection framework. The Implementation section highlights the core functionalities, key code snippets, and system performance with supporting screenshots, demonstrating the effectiveness of the developed system.

The report concludes with the Conclusion, summarizing the project's key findings, impact on railway safety, and overall effectiveness. The Future Enhancements section discusses potential improvements and directions for further research, followed by the References, which provide citations for all referenced materials, ensuring the report's scholarly rigor.

CHAPTER 2: LITERATURE REVIEW

This chapter provides a literature survey on railway safety enhancement, summarizing various deep learning and computer vision techniques employed in different studies to improve human activity recognition, accident prevention, and real-time monitoring in railway environments.

1.3 Literature Survey

The authors Cao, Zhiwei in paper [1] present a comprehensive survey on railway intrusion detection systems, emphasizing the role of machine vision in enhancing railway safety and accident prevention. The study systematically reviews various intrusion detection approaches, including ground-based monitoring, onboard inspections, and UAV-based systems, outlining their strengths and limitations. Key challenges identified include sensitivity to environmental conditions, high implementation costs, and real-time processing constraints. The paper highlights the potential of deep learning, sensor fusion, and pattern recognition in overcoming these limitations. Furthermore, the authors propose innovative perspectives leveraging emerging technologies such as multi-sensor integration, 5G communication, and AI-powered drones to develop more efficient and scalable railway intrusion detection solutions.

The authors Joseph Redmon in paper [2] introduce YOLO (You Only Look Once), a revolutionary real-time object detection framework that formulates detection as a single regression problem, directly predicting bounding boxes and class probabilities from full images in one pass. By eliminating traditional multi-stage pipelines, YOLO achieves high-speed processing, with the base model running at 45 fps and a faster variant, Fast YOLO, reaching 155 fps. This approach enhances detection efficiency while reducing false positives on background regions compared to methods like R-CNN and DPM. Furthermore, YOLO demonstrates strong generalization capabilities, making it effective across diverse domains. Despite its advantages, YOLO faces challenges in precisely localizing small objects and sometimes misclassifies complex scenes due to limited contextual understanding. These limitations open avenues for research in improved feature extraction, attention mechanisms, and hybrid approaches that integrate spatial and contextual information. Additionally, optimizing YOLO for edge devices and resource-constrained environments remains an active area of exploration. The framework's potential applications extend to autonomous navigation, smart surveillance, and real-time anomaly detection, driving further innovation in real-time object detection.

The authors Alexey Bochkovskiy in the paper [3] introduce YOLOv4, a state-of-the-art real-time object detection framework optimized for single GPU usage, making high-speed and accurate object detection accessible on conventional hardware. YOLOv4 integrates advanced features such as Weighted Residual Connections (WRC), Cross-Stage Partial connections (CSP), Cross mini-Batch Normalization (CmBN), Self-Adversarial Training (SAT), Mish activation, and Mosaic data augmentation, achieving exceptional performance. The model delivers 43.5% AP (65.7% AP50) on the MS COCO dataset at a real-time speed of ~65 FPS on a Tesla V100 GPU, surpassing prior models like YOLOv3 and EfficientDet in both speed and accuracy. By addressing challenges in training CNN-based detectors, such as dependency on multiple GPUs and large mini-batches, YOLOv4 introduces optimizations for single GPU training and parallel computations. This innovation enables YOLOv4 to cater to diverse real-world applications, including autonomous process management and urban surveillance, moving beyond recommendation systems. With its open-source availability, YOLOv4 sets a benchmark for accessible, efficient, and high-performing object detection systems.

The authors of the paper [4] present the development of an Automatic Video-Surveillance (AVS) system aimed at improving safety at railway level crossings (LC) by utilizing a passive stereo vision principle. The system employs two color cameras to detect and 3D-localize objects, such as vehicles and pedestrians, within hazardous zones, providing timely alerts to train drivers via red lighting and real-time video feeds of potential risks. By incorporating advanced techniques like Spatio-temporal Independent Component Analysis (stICA) for motion and stationary object detection and a selective stereo matching algorithm for accurate 3D localization, the system effectively handles environmental noise, illumination changes, and poor image quality. It addresses the limitations of previous systems, such as false alarms and sensitivity to adverse weather conditions, by offering robust computational techniques. Designed to complement existing LC infrastructure, the AVS system automates obstacle detection, enhances situational awareness, and reduces accidents caused by human error, which accounts for 99% of LC incidents. This innovative solution significantly improves safety and provides a foundation for further advancements in intelligent surveillance and real-time video processing in dynamic environments. While the proposed Automatic Video-Surveillance (AVS) system demonstrates significant potential for improving safety at railway level crossings, several unresolved issues and emerging opportunities remain. One major challenge is the system's dependency on environmental conditions, such as adverse weather, which can impact camera visibility and object detection accuracy. Addressing these limitations requires further development in robust algorithms capable of handling extreme conditions like fog, rain, or snow. Additionally, while the system effectively integrates object detection and 3D localization, ensuring real-time performance for high-speed trains in dense traffic areas remains a challenge. Opportunities lie in leveraging advancements in artificial intelligence and machine learning to improve detection accuracy and adaptability to complex scenarios. Emerging technologies like LiDAR and thermal imaging could enhance object recognition and localization in low-visibility conditions.

The authors of the research paper [5] emphasize the significant role that organizational safety culture plays in influencing worker behavior and reducing unsafe practices in railway maintenance. Through qualitative interviews with staff, the study identifies a range of factors contributing to unsafe behavior, spanning from immediate, trackside factors (e.g., weather conditions) to more distant organizational influences (e.g., management style and contradictory rules). The findings highlight the complexity of improving safety, as individual behavior is deeply affected by the organizational environment. As a result of the research, the concerned maintenance company has developed a strategy to enhance its safety systems and overall safety culture. This strategy includes adopting a participatory approach to implement changes and incorporating Reason's five aspects of safety culture to guide the transformation. The study suggests that fostering a positive safety culture requires addressing organizational factors such as management style, safety communication, and consistent safety policies. It also provides practical insights for other organizations seeking to manage unsafe working behaviors and create a more proactive safety culture.

The authors of the research paper [6] emphasize the use of advanced video analytics and CCTV systems to enhance safety and security at railway stations. The paper presents a system designed to detect and prevent trespassing by utilizing intelligent video surveillance to monitor critical areas where unauthorized access could pose a risk. The system analyzes video feeds in real-time, using sophisticated algorithms to detect trespassers, track their movements, and alert security personnel immediately. By automating the detection process, the system not only reduces response time but also ensures a more efficient and reliable way of monitoring potential safety breaches. The paper highlights the effectiveness of integrating video analytics with existing CCTV infrastructure, providing a cost-effective solution for enhancing railway station security. The findings demonstrate that the combination of real-time detection, tracking, and alerting significantly improves the ability to prevent accidents and unauthorized access in high-risk zones. Overall, the research underscores the potential of CCTV and video analytics to improve railway station safety by providing accurate, timely alerts, reducing reliance on manual monitoring, and increasing the overall security of the station. Despite the promising potential of the CCTV and video analytics system for trespassing detection, several unresolved issues and emerging opportunities remain. One key challenge is the system's ability to accurately detect trespassers in various environmental conditions, such as low lighting, poor weather, or obstructed camera views.

This study [7] proposes a YOLO-UAT method for efficient foreign object detection in railways to improve safety and operational efficiency. The method replaces the backbone extraction network of YOLOv5s with EfficientNet for a lightweight model that enhances detection speed. A C3_CBAM module is introduced to improve feature extraction, especially for small-scale objects, and the K-means++ algorithm is used for better a priori frame clustering, improving accuracy and convergence speed. YOLO-UAT reduces parameters by 36% and increases mAP by 6.1% to 91.5%. The model, deployed on a Jetson Nano, achieves a detection rate of 26.4 FPS. Experimental results confirm improved accuracy, speed, and lightweight design suitable for railway deployment.

The study [8] proposes a track defect detection network (DSO-YOLO) based on an improved YOLOv5s model to address issues like missed detection, inaccurate positioning, and difficulty detecting small objects in traditional methods. The approach introduces a decoupling head to separate target position and classification data, improving generalizability. A new small-object detection layer enhances multiscale detection, and the ODConv module with a 4-D attention mechanism refines feature extraction, addressing issues like illumination and overlapping defects. The improved model achieves a mean average precision of 98.6%, surpassing YOLOv5s by 3.7%, demonstrating higher accuracy in complex environments for track defect detection.

This paper [9] proposes a concealed object detection method using an enhanced YOLO network to address challenges in millimeter-wave (MMW) imaging, such as low signal-to-noise ratio, poor resolution, and similarity between targets and backgrounds. The method incorporates an adaptive denoising technique to improve image clarity, and introduces two novel modules: the edge-spatial feature fusion (ESFF) module to enhance edge feature learning, and the hierarchical scale-aware feature fusion (HSAFF) module to reduce classification errors and false detections. These modules are integrated into the YOLOv8 framework. Experimental results on MMW images show that the model achieves mean average precision (mAP)@[0.5] of 98.3% and mAP@0.5:0.95 of 81.5%, outperforming the baseline and other detection models by 5.1% and 4%, respectively.

This study [10] proposes a UAV-based system for detecting and evaluating potential safety hazards (PSHs) along high-speed railroad tracks to enhance operational safety. The system introduces a novel hybrid learning architecture, UYOLO (U-shape YOLO), which integrates a CSP-based backbone and detection branch to generate high-level features at three scales. An innovative parsing branch helps transmit context information for pixel-level parsing tasks. A new loss function, minimum point distance IoU (MPD-IoU), optimizes the bounding box regression process. Additionally, an image-based hazard evaluation model is developed to assess the risk level of detected PSHs. Experimental results on a UAV imagery dataset show that the system achieves a high detection rate while remaining efficient and user-friendly.

This paper [11] proposes a novel method for generating railway intruding pedestrian image sequences using pose transfer, aimed at improving pedestrian intrusion detection for high-speed railway safety. The method addresses the issue of insufficient intruding images by creating synthetic samples with automatic annotations for deep learning training. A pose transfer module based on a generative adversarial network (GAN) is introduced to extract pose features, while a feature fusion module using spatial attention mechanisms helps preserve important pose details. Experimental results on the DeepFashion dataset show that the proposed method improves the quality of generated images, with a 5.6% increase in SSIM index and a 2.9% increase in pckh index compared to traditional pose transfer algorithms. Additionally, the generated images enhance the accuracy of pedestrian detection algorithms.

This paper [12] presents a novel unsupervised approach for detecting abnormal objects in railway track inspection images using a Random Network-Assisted Autoencoder (RNaAE). The method addresses the limitations of traditional supervised detection, which struggles with diverse abnormal object categories and insufficient abnormal samples. The RNaAE combines a learnable network that fits a randomly initialized stochastic network, using the difference between predictions to identify anomalies. The method integrates a traditional Autoencoder and applies a Gaussian mixture model to classify objects as normal or abnormal based on anomaly scores. Experimental results on a railway anomaly dataset show that RNaAE outperforms state-of-the-art methods, achieving 98.23% AUROC and 92.02% F1-score.

This study [13] proposes a novel deep learning-based method to improve YOLOv5 for real-time arc detection in pantograph-catenary systems, addressing challenges like low accuracy and high computational complexity. The method introduces two key innovations: (1) a Squeeze-and-Excitation-based C3 (SEC3) attention module to prioritize informative features, and (2) a Bi-directional Feature Pyramid Network (BiFPN) for enhanced multi-scale feature fusion. Comprehensive experiments on a diverse real-world dataset demonstrate that the improved YOLOv5 network outperforms state-of-the-art methods in both accuracy and efficiency. Ablation studies confirm that the SEC3 and BiFPN modules significantly enhance performance. The approach provides a promising solution for automatic arc detection, contributing to the safety and reliability of high-speed railways.

This paper [14] focuses on an automated track monitoring system intended to improve railroad safety is presented in this project. The device uses a GPS module and an ultrasonic sensor to track conditions and identify deviations in real-time. It detects deviations beyond predefined criteria, obtains GPS coordinates, and measures distances to surrounding objects. When deviations occur, the system notifies the operator and sends GPS locations for immediate assistance, aiming to reduce risks and enhance train safety. Additionally, the system includes an object detection and alert mechanism using computer vision. This component employs a pre-trained deep learning model via OpenCV to identify objects on or near the tracks in real-time. If an object remains in the detection zone for over a predefined duration , an email alert is triggered. This ensures prompt reporting of potential hazards, further enhancing railway safety.

This paper [15] presents a novel modular visual processing framework for continuous train positioning, addressing the limitations of traditional infrastructure-based methods. The system uses stereo vision sensors to capture environmental information, which is processed by a feature extraction algorithm to detect landmarks along the railway. These results are fed into a dual-stream parallel processing architecture that includes an anchor-based stereo matching module to calculate depth values and a dynamic region of interest-based multi-object tracking module to assign unique IDs to landmarks. This integration allows for the tracking of landmark movements and the calculation of displacement due to train motion, improving positioning accuracy. The system's performance is validated through field tests on Beijing Metro Line 9 and the Capital Airport Line, achieving an average mean relative error of 0.19% and positioning accuracy within 1 meter over a 100-meter range.

This article [16] proposes a method for detecting plastic waste intrusions along railway tracks using the YOLO-v5 algorithm and model ensemble with surveillance cameras. Experiments were conducted on various YOLO-v5 model sizes to identify the optimal one for detecting plastics. The largest model (YOLOv5l) achieved the highest performance with an overall accuracy (OA) of 82.6% and a mean Average Precision (mAP) of 0.822. Two ensemble strategies were explored: one combining nano, small, and medium-sized models, and another adding large-sized models to the mix. The second strategy yielded the best results, with an OA of 85.4% and mAP of 0.834. The study shows that YOLO-based ensemble models significantly improve plastic waste detection and can be applied to UAV and satellite-based high-resolution imagery for broader monitoring.

This article [17] proposes a method for identifying missing spare nuts on U-shaped hoops in high-speed railway contact network insulator areas using YOLO v3 and SENet. The process is divided into two stages: first, YOLO v3 detects and locates the U-shaped hoop in the high-speed railway image. The identified hoop is then extracted and resized to 224×224 pixels. In the second stage, SENet classifies the resized image to determine if a spare nut is missing. To improve image visualization, class activation maps are used to highlight the U-shaped hoop. The method addresses the challenge of high resource consumption when recognizing small objects in high-resolution images. The approach achieved an accuracy of 88.24% on a dataset from the Xinjiang high-speed railway contact network, demonstrating its potential for real-world application in detecting missing spare nuts.

This study [18] investigates a track foreign object intrusion detection algorithm to prevent safety accidents on railways caused by foreign objects. The paper proposes an enhanced YOLOv3 network specifically for high-speed railway foreign object detection, designed to improve the network's ability to use image features and enhance detection performance. The enhanced network achieves an average detection accuracy of 79.2%, which is 4.4% higher than the original YOLOv3 network, although with a slight reduction in detection speed. The enhanced YOLOv3 effectively improves detection accuracy across targets of different scales, making it suitable for railway safety applications.

This study [19] proposes a real-time automated railway freight vehicle inspection method based on channel pruning to address issues of labor intensity, low efficiency, and oversight in traditional manual inspection methods. The approach uses a YOLOv5s model, incorporating modules like Focus, BottleneckCSP, and SPP, and applies channel pruning to reduce model size and improve deployment on resource-constrained devices. Experimental results show that the pruned model reduces parameters by 91.9%, decreases size by 11.23 MB, and achieves a mean Average Precision (mAP) of 98.14%, only 0.183% lower than the unpruned model. The proposed method also demonstrates faster inference times and smaller model sizes compared to YOLOv5x, YOLOv5n, and YOLOv5m, with only minimal mAP losses. This demonstrates the effectiveness of the approach for automated inspection of railway freight cars, with a balance between detection speed, accuracy, and model size.

This paper [20] presents a system for identifying obstructions on railway tracks, particularly in forest areas where animals and environmental factors create significant safety risks. The system involves installing cameras along the railway track to capture real-time data. This data is processed and analyzed to detect any potential obstructions using Convolutional Neural Networks, specifically YOLOv8, a fast and accurate model. The model is trained on a custom dataset proposed in the paper. When an obstruction is detected, the system alerts the train pilot and the nearest stations via a mailing service, enabling timely actions to prevent accidents or delays. By providing real-time alerts about track obstructions, this system aims to enhance safety and efficiency in forested areas, ensuring a safer and more reliable railway network.

The summary of the literature survey

- Machine Vision & AI Integration: Many studies emphasize the integration of deep learning, sensor fusion, and machine vision to improve safety, highlighting the potential of emerging technologies like multi-sensor integration, AI-powered drones, and 5G communication.

- YOLO Framework: YOLO-based methods (such as YOLOv5, YOLOv4, and YOLO-UAT) dominate object detection tasks, providing real-time, high-speed solutions with varying performance levels for different applications like track defect detection, trespassing, and object intrusion.

- Challenges in Environmental Conditions: A recurring challenge is the sensitivity of detection systems to environmental factors (e.g., weather, lighting), which affects the accuracy of image-based monitoring systems in railway safety.

- Real-Time Processing & Efficiency: Real-time processing is critical for many applications, with YOLO and its variants offering high FPS and reduced false positives. However, issues like precise localization of small objects and adapting to edge devices remain.

- Track & Object Defect Detection: Advances in track defect detection networks (like DSO-YOLO) address problems like missed detection, small object localization, and poor environmental conditions, leading to higher accuracy and improved safety.

- Automated & Lightweight Solutions: Models like YOLOv5s and YOLO-UAT have been optimized for real-time deployment in resource-constrained environments (e.g., Jetson Nano), offering practical and lightweight solutions for railway operations.

- Organizational & Human Factors: Several papers stress the importance of organizational safety culture and its role in reducing unsafe behaviors, indicating a need for a holistic approach to safety in addition to technology.

- Practical Deployment & Performance: Most studies show that while the technologies proposed are promising, practical challenges such as high-speed train integration, real-time performance, and scalability still need further development.

The gaps identified after the literature survey

- Environmental Adaptability: Many detection systems are still not robust enough to handle diverse environmental conditions (e.g., fog, rain, and varying lighting). There is a need for improved models that can adapt to these changing conditions and still deliver high accuracy.
- Real-Time Processing vs. Accuracy: While real-time processing is emphasized, there is often a trade-off between speed and accuracy, particularly when detecting smaller objects or defects. The challenge is to optimize detection speed without compromising accuracy,

especially in high-speed rail environments.

- Small Object Detection: Despite advancements, detecting small objects (e.g., debris or minor track defects) remains a significant challenge. Current models may struggle with low-resolution images or small objects, leading to missed detections and safety risks.
- Edge Device Performance: While YOLO variants like YOLOv5s are optimized for edge devices, there is still limited research on optimizing detection models for other low-power or resource-constrained edge devices that may be deployed in railway systems.
- Model Generalization: The current models often perform well in controlled environments but struggle to generalize across different geographical regions, track types, and weather conditions. There is a need for more versatile models that can be applied universally across various railway systems.
- Integration with Existing Railway Systems: While many solutions focus on advanced detection techniques, fewer studies explore the seamless integration of these technologies with existing railway infrastructure, management systems, and safety protocols.
- Data Privacy and Security: As surveillance and real-time monitoring technologies advance, there are concerns about the privacy and security of the collected data, particularly with regard to passenger and operational information. This gap needs attention to ensure that AI-based systems do not compromise data security.
- Human Activity Recognition: Although several papers mention human activity recognition in the context of safety, there is a lack of focus on real-time human behavior analysis, which could help prevent accidents caused by track trespassing or improper actions around trains.
- Long-Term Reliability: Many of the proposed solutions are evaluated under limited test conditions, with little attention paid to the long-term reliability and maintenance of these systems. More research is needed on how these technologies can be maintained and scaled for long-term use across extensive railway networks.
- Comprehensive Safety Systems: While technologies like AI-based detection and UAVs offer advanced solutions, there is a gap in holistic safety frameworks that combine multiple technologies (e.g., machine vision, human activity recognition, and sensor fusion) into a unified safety system that continuously adapts and learns.

Objectives of the Project

- Develop a Real-Time Human Presence Detection System: Implement a YOLO based object detection and activity recognition system to identify unauthorized human presence on railway tracks in real-time, ensuring accurate and swift alerts.

- Optimize System Performance Across Diverse Conditions: Train and evaluate model using varied datasets that simulate different environmental conditions (e.g., varying weather, lighting, and occlusion) to ensure high detection accuracy and robustness.

- Integrate a Scalable Railway Safety Solution: Deploy the trained model within an operational system capable of processing real-time video feeds, issuing timely alerts, and demonstrating real-world efficiency in enhancing railway safety across India's extensive railway network.

1.4 Existing and Proposed System

The existing railway safety systems primarily rely on traditional methods such as manual surveillance, physical barriers, and CCTV cameras to monitor and protect railway tracks. While these mechanisms are essential for identifying potential hazards, they have significant limitations. Manual surveillance is labor-intensive and prone to human error, while physical barriers are often ineffective in preventing unauthorized access, especially in remote areas. CCTV cameras, though widespread, have limited coverage and cannot always provide real-time alerts or accurately detect threats in dynamic environments. Moreover, these systems often struggle with issues such as poor lighting, adverse weather conditions, and occlusions that affect the clarity of video feeds. Consequently, despite these existing efforts, railway tracks remain vulnerable to unauthorized human presence, leading to frequent accidents and fatalities. The need for a more reliable, real-time solution that leverages advanced technology like AI has become increasingly urgent.

The proposed system aims to overcome the limitations of traditional safety mechanisms by incorporating AI-driven object detection and activity recognition. This solution will use the YOLO model to monitor railway tracks in real-time, identifying unauthorized human presence swiftly and accurately. Unlike conventional CCTV systems, this AI-powered system will be capable of detecting human activity under diverse environmental conditions such as low lighting or adverse weather. It will also provide immediate alerts to safety personnel, enabling quicker responses and potentially preventing accidents. The system will be modular, incorporating data collection, preprocessing, model training, and real-time deployment, ensuring scalability and adaptability to different railway networks. By leveraging advanced AI techniques, the proposed system offers a more robust, cost-effective, and reliable solution to enhance railway safety and reduce fatalities.

1.4.1 Problem Statement and Scope of the Project

The problem this project seeks to address is the critical safety issue of unauthorized human presence on railway tracks, which remains a leading cause of accidents and fatalities in India's vast and busy railway network. Traditional safety measures such as manual surveillance, physical barriers, and CCTV cameras have proven inadequate in preventing these incidents. These methods fail to offer real-time alerts, have limited coverage, and often struggle with environmental challenges like poor lighting, varying weather conditions, and obstructions in the camera's line of sight. The lack of a fully automated, efficient, and scalable system for real-time monitoring of railway tracks has created a significant gap in safety measures. This project aims to develop an intelligent, AI-driven solution to enhance railway safety by employing advanced object detection and human activity recognition techniques.

By leveraging the YOLO-V8 deep learning algorithm, this system will enable real-time identification of unauthorized human presence on tracks, allowing for swift action and reducing response time to potential threats. The scope of the project includes the design and deployment of an object detection system capable of operating efficiently under diverse conditions such as varying lighting, weather, and camera angles. It will involve data collection and preprocessing to create a robust dataset, model training for high accuracy and performance, evaluation and testing to ensure the system's reliability, and finally, deployment for real-time monitoring. The system will be integrated with live video feeds and will trigger immediate alerts when unauthorized human presence is detected, offering a significant improvement over current safety practices and ensuring that railway stations and tracks are monitored continuously with minimal delay.

Ultimately, the project aims to create a scalable and adaptable safety solution that can be applied to other regions or transportation systems globally, demonstrating the transformative potential of AI in the field of Intelligent Transportation Systems (ITS).

1.4.2 Methodology of the Proposed System

The methodology for the proposed railway safety system incorporates a series of structured steps aimed at utilizing AI-powered object detection and activity recognition to monitor and secure railway tracks. The approach is designed to ensure scalability, accuracy, and real-time performance across diverse environments. The key phases of the methodology are as follows:

1 Data Collection and Preprocessing

- The first step involves gathering a diverse dataset of images and videos from railway tracks under different conditions (e.g., varying lighting, weather, and occlusions). These datasets will be annotated for human presence and activity, using tools like Roboflow for precise labeling.
- Data augmentation techniques (e.g., rotations, flipping, color adjustments) will be applied to expand the dataset, ensuring robustness in the model's performance under varied real-world conditions.
- Preprocessing will also involve resizing, normalization, and conversion of data into formats suitable for training the object detection model, using Python libraries such as OpenCV, Numpy, and Pandas.

2 Model Selection and Training

- The YOLO (You Only Look Once) model will be chosen for object detection due to its speed and accuracy, making it ideal for real-time deployment in dynamic environments like railway tracks.
- The training will take place on a cloud-based platform (e.g., Google Colab) to leverage powerful GPU resources for computational efficiency. The dataset will be split into training, validation, and testing subsets to ensure generalization and avoid overfitting.
- During training, the model will learn to identify human presence on railway tracks and classify different activities (e.g., standing, walking) to provide more granular alerts. Metrics like precision, recall, and mean average precision (mAP) will be used to evaluate the model's performance.

3 Evaluation and Testing

- The trained YOLO model will undergo rigorous evaluation using real-world test scenarios to assess its ability to detect unauthorized human presence on tracks in real-time. Performance metrics such as accuracy, precision, recall, and mAP will be calculated to evaluate its effectiveness across varying environmental conditions, including poor lighting and occlusions.
- The testing phase will also focus on the model's response time to ensure it meets the requirements for timely alerts in real-world scenarios.

4 Real-Time Deployment:

- The trained model will be deployed into a real-time inference pipeline that continuously monitors live video feeds from railway tracks. The system will process video frames and detect human presence instantaneously.
- Using Python and OpenCV, the model will be integrated into the deployment pipeline, capable of generating real-time alerts when unauthorized human presence is detected on the tracks. The system will notify relevant personnel or safety control centers immediately, providing information such as location and activity type for quick response.

This methodology ensures that the proposed system not only detects human presence effectively but also provides real-time, actionable insights that can significantly enhance railway safety.

1.4.3 Technical Features of the Proposed System

The proposed system combines state-of-the-art deep learning models for accurate human presence detection, utilizes data augmentation for model robustness, leverages Roboflow for efficient data annotation, and supports scalable deployment for real-time operation across diverse railway environments.

- Deep Learning Integration: Utilizes the YOLO model, a state-of-the-art deep learning algorithm, for real-time object detection and activity recognition on railway tracks.
- Data Augmentation: Enhances model robustness by introducing variations in the training data, including flipping, rotating, and shifting, to simulate diverse real-world conditions.
- Transfer Learning: Reduces computational overhead and improves accuracy by fine-tuning a pretrained YOLO model on the railway-specific dataset, ensuring faster training and higher precision.
- Roboflow Integration: Utilizes Roboflow for data annotation, augmentation, and dataset management, streamlining the preparation process and ensuring high-quality input data for model training.
- Scalable Deployment: Designed for adaptability, enabling deployment on edge devices, mobile platforms, or cloud services for seamless real-time monitoring and alerts across diverse environments.

1.5 Tools and Technology used

- Deep Learning Framework
  TensorFlow: For implementing and fine-tuning the YOLO model for object detection.
  Keras: To simplify model building and training through its high-level API.

- Data Processing and Augmentation
  OpenCV: For preprocessing tasks such as image resizing, normalization, and enhancement.
  Albumentations: To apply advanced data augmentation techniques like flipping, rotation, and brightness adjustments, improving model robustness.

- Data Annotation and Management
  Roboflow: For efficient image annotation, dataset management, and augmentation, ensuring high-quality training data.

- Development Environment:
  Google Colab: For GPU-accelerated training and model testing, enabling faster model iterations.
  VS Code: For code development, debugging, and version control management.

- Visualization and Evaluation Tools
  TensorBoard: For visualizing training metrics, including loss and accuracy curves during model evaluation.
  Matplotlib: For plotting performance graphs and visualizing the detection outputs.

1.6 Software and Hardware Requirements

Hardware Requirements refer to the physical components and specifications of the machine or equipment necessary to run and support the project. This includes things like the processor, memory (RAM), graphics card (GPU), and any other hardware tools required (e.g., camera, sensors).

- Processor
  Minimum: Intel i5 or equivalent processor.
  Recommended: Intel i7 or higher for faster and more efficient processing, especially when dealing with deep learning models.

- GPU
  Minimum: NVIDIA GTX 1050 Ti for effective model training and inference.
  Recommended: GPUs like NVIDIA RTX 2060 or higher for faster deep learning model training, especially when handling large datasets and complex models.

- RAM:
  Minimum: 8 GB of RAM.
  Recommended: 16 GB or more for better handling of large datasets and ensuring the smooth operation of machine learning tasks.

- Camera Specs

A camera with a resolution of at least 1080p, capable of capturing clear images and video feeds for detecting unauthorized human presence on railway tracks.

The Literature survey concludes with the Hardware and Software requirements for the proposed system, which involves real-time human presence detection using the YOLO model. This section outlines the potential for enhancing existing railway safety systems by achieving the objectives of accurate, real-time detection and alerting, thus addressing the limitations of traditional methods.

CHAPTER 3: SOFTWARE REQUIREMENT SPECIFICATION

This chapter introduces to definitions, acronyms and abbreviations used in the report, additionally it gives the general description of the product. It also describes the functional, non-functional requirements and external interface requirements.

3.1 Introduction

Software specifications are detailed description of the software system's functionalities, features, and interactions. They outline the software's intended behavior, system requirements, design constraints, and the environment in which it will operate. Software specifications serve as a blueprint for the development process, guiding developers in creating the system and ensuring that all stakeholders have a clear understanding of the software's capabilities and limitations. They typically include details on programming languages, frameworks, tools, libraries, and APIs used, as well as user interface requirements, performance standards, and security considerations. These specifications ensure that the software meets user expectations and operates efficiently within its intended environment.

3.1.1 Definitions

- YOLO (You Only Look Once): A state-of-the-art object detection algorithm designed for real-time applications. YOLO models detect and classify objects in a single pass, making it ideal for high-speed tasks like real-time monitoring and tracking in dynamic environments.

- AI (Artificial Intelligence): A branch of computer science focused on creating systems capable of performing tasks that would typically require human intelligence, such as image recognition, speech processing, decision-making, and more.

- Deep Learning: A subset of machine learning that uses neural networks with many layers to analyze complex patterns in large datasets. It is particularly effective in tasks like image and speech recognition.

- TensorFlow: An open-source machine learning framework developed by Google for building and deploying deep learning models. It is widely used for creating neural networks and is optimized for both training and inference.

- OpenCV (Open Source Computer Vision Library): A comprehensive library for image and video analysis, widely used in computer vision tasks such as object detection, motion tracking, image processing, and facial recognition.

- Python: A high-level programming language known for its simplicity and extensive support for machine learning, data science, and artificial intelligence libraries, including TensorFlow, PyTorch, and OpenCV.

- Roboflow: A tool for automating the process of labeling, augmenting, and managing datasets for computer vision tasks. It integrates with various machine learning frameworks and helps in building custom models for tasks like object detection.

- Google Colab: A free, cloud-based platform for developing machine learning models, providing access to GPUs for fast model training and testing, particularly useful for deep learning tasks.

- Artificial Neural Networks (ANN): A computational model inspired by the way biological neural networks in the brain process information. ANNs are used to recognize patterns and make decisions, which is the backbone of most deep learning systems.

- CNN (Convolutional Neural Network): A type of deep learning algorithm specifically designed to process structured grid data, such as images. CNNs excel in visual recognition tasks, such as identifying objects in images and video feeds.
- Transfer Learning: A machine learning technique where a pre-trained model (usually on a large dataset) is adapted and fine-tuned for a different but related task, significantly reducing training time and improving performance.

- Mean Average Precision (mAP): A performance metric used to evaluate the accuracy of object detection models. It calculates the average precision across all classes and is widely used in computer vision tasks to measure overall model effectiveness.

- Computer Vision: A field of AI focused on enabling machines to interpret and make decisions based on visual inputs from the world. It includes tasks like image recognition, object detection, and motion tracking.

- Segmentation: A computer vision task that involves partitioning an image into multiple segments to simplify the analysis. It is often used to detect boundaries of objects or regions of interest in an image.

3.1.2 Acronyms

- YOLO: You Only Look Once
- AI: Artificial Intelligence
- ANN: Artificial Neural Network
- CNN: Convolutional Neural Network
- mAP: Mean Average Precision
- DL: Deep Learning
- RNN: Recurrent Neural Network
- GPU: Graphics Processing Unit
- IDE: Integrated Development Environment
- API: Application Programming Interface

3.1.3 Overview

This project aims to enhance railway safety by developing a computer vision-based system for detecting unauthorized human presence on railway tracks. By leveraging state-of-the-art object detection and deep learning techniques, particularly the YOLO model, the system will monitor tracks in real-time, identifying potential threats to prevent accidents. The system combines cutting-edge AI and image processing to continuously analyze video feeds, detect human activity, and provide immediate alerts. This approach offers a robust, scalable, and real-time solution to address the limitations of traditional railway safety systems, ultimately improving safety standards across India's vast railway network.

3.2 General Description

The general description section provides an overall summary of the project, highlighting its primary goals, objectives, and the technological approach used to achieve them. It outlines the key problem being addressed, in this case, enhancing railway safety by detecting unauthorized human presence on tracks, and introduces the proposed solution, such as leveraging AI, deep learning models like YOLO for real-time object detection, and activity recognition. This section provides context for the project, offering an overview of its scope, key features, and the value it brings to users, helping stakeholders understand the purpose and impact of the system without delving into technical specifics.

3.2.1 Product Perspective

The railway safety monitoring system is designed to address the critical issue of unauthorized human presence on railway tracks, which leads to frequent accidents and fatalities. This system leverages computer vision and deep learning techniques, specifically using the YOLO model, to detect human activity in real-time through video feeds from strategically placed cameras along railway tracks. The solution will be scalable, cost-effective, and suitable for large networks, ensuring effective monitoring across vast railway corridors. The product will be implemented as a real-time alerting system that can be integrated with existing railway surveillance infrastructure, providing immediate alerts to security personnel when unauthorized human presence is detected.

The primary stakeholders include railway operators, safety agencies, and government bodies. The system will be designed to operate efficiently in varied environmental conditions such as low lighting or adverse weather, improving the overall safety of railway operations.

3.2.2 Product Functions

- Real-Time Human Detection: Continuously monitors railway tracks for unauthorized human presence using YOLO-based object detection.
- Activity Recognition: Identifies and classifies human activities (e.g., walking, sitting) to assess the potential threat level.
- Environmental Adaptation: Detects human presence accurately in various conditions, such as poor lighting, weather changes, and occlusions.

3.2.3 User Characteristics

User characteristics refer to the defining traits, skill levels, and technical needs of different user groups interacting with the system, helping to tailor the system's design and functionality to meet their specific goals and abilities.

Primary Users (Railway Authorities, Station Operators)

- Skill Level: Non-technical users, with basic understanding of railway operations but limited technical expertise in AI and deep learning.
- Technical Needs: User-friendly interface for real-time monitoring, alerts, and easy access to detected threat information.
- Goal: To ensure the safety of the railway tracks by receiving accurate alerts and identifying unauthorized human presence in real-time.

Secondary Users (Safety Technicians, Engineers)

- Skill Level: Moderately technical, with a background in railway safety and operations.
- Technical Needs: Access to detailed reports, system performance metrics, and the ability to adjust or calibrate detection settings.
- Goal: To analyze the data generated by the system, perform maintenance tasks, and ensure system effectiveness over time.

End Users (Government Agencies, Railway Policy Makers)

- Skill Level: Highly technical, involved in large-scale policy making and railway safety infrastructure.
- Technical Needs: Ability to scale the system for national-level deployment, integration with existing safety systems, and detailed analytics for decision-making.
- Goal: To oversee the large-scale implementation of the system across various railway networks and improve national railway safety measures.

3.2.4 General Constraints

- Accuracy of Human Detection:
  The system's performance in detecting human presence on railway tracks may vary due to environmental factors such as lighting conditions, weather, and image quality. Accurate detection depends heavily on the quality and diversity of the dataset used for training.

- Hardware and Device Limitations:
  The devices used for real-time monitoring (such as cameras and edge devices) must meet certain specifications to ensure high-quality video feeds for effective human detection. Inadequate hardware may affect detection accuracy.

- Model Performance and Adaptability:
  The deep learning model may not perform equally well in all environmental conditions. Continuous retraining and fine-tuning of the model might be necessary to ensure high accuracy and effectiveness across different weather, lighting, and track conditions. There will be a need for active learning with random sampling to avoid model drifts.

3.2.5 Assumptions and Dependencies

- Dataset Availability and Quality
  The system assumes the availability of a comprehensive and diverse dataset for human activity detection on railway tracks, including various types of activities, environments, and weather conditions.

- Railway Infrastructure and Camera Placement
  It assumes that railway tracks are equipped with sufficient surveillance cameras or monitoring devices capable of capturing high-quality video footage under varying conditions.

- Model Update and Retraining
  The system will rely on periodic updates and retraining of the model with new data to improve the accuracy of human presence detection across different railway environments and weather conditions.

- User Awareness and Support
  The system assumes that railway staff or relevant authorities will receive adequate training and support to effectively use the alerts generated by the system and take appropriate action in the case of detected unauthorized human presence on the tracks.

3.3 Functional Requirements

Functional requirements for the Railway Safety Detection System refer to the specific tasks, actions, or services that the system must perform to achieve its primary objective of detecting unauthorized human presence on railway tracks. These requirements describe the expected behavior of the system, such as real-time detection of human activities, generating alerts, processing images using deep learning models like YOLO, and ensuring the system's usability for railway operators and stakeholders. The functional requirements define what the system must do to ensure safety, reliability, and effective monitoring of railway tracks to prevent accidents.

3.3.1 Introduction

The Railway Safety Detection System utilizes advanced deep learning models to identify unauthorized human presence on railway tracks in real-time. Using a YOLO deep learning model, the system is fine-tuned on a dataset of railway track images to accurately detect human activities. The system provides an intuitive interface for railway operators, alerting them to potential hazards as soon as detected. Through continuous monitoring and instant notifications, it helps in preventing accidents and ensuring the safety of the railway environment. The functional requirements focus on seamless real-time detection, notification generation, and efficient system performance under varying conditions.

3.3.2 Input

Track Images/Video Feeds: Real-time images or video frames captured by cameras installed along the railway tracks. These images should be clear, focused, and high resolution to ensure accurate detection of human presence.

File Format: The images or video feeds should be uploaded in standard formats like JPEG, PNG, or MP4 for video processing.

Camera Metadata (optional): Additional metadata such as the camera's location, angle, and environmental conditions (e.g., lighting, weather) to enhance model accuracy and performance in varying conditions.

3.3.3 Processing

- Image and Video Capture
  1. Video Stream Input: The system requires a continuous video stream (from cameras installed along the tracks) or real-time images, captured in high-definition to ensure clear and accurate detection.
  2. Preprocessing: Captured frames undergo preprocessing to resize and normalize the images for the YOLO-V8 model input. Frames may also be cropped or adjusted to enhance the region of interest (ROI) around the railway tracks.

- Human Detection Using YOLO-V8
  1. Real-Time Detection: The model processes the input frames in real-time to detect any humans within the frame. This involves using the model to identify bounding boxes that represent humans and other objects of interest in the environment.
  2. Object Localization: The system uses YOLO's capabilities to not only classify but also localize the detected humans accurately, identifying their exact position in the frame.
  3. Activity Recognition: For detected humans, activity recognition algorithms determine whether the person is engaging in a hazardous activity (e.g., walking, standing, or sitting on the tracks), helping to classify the type of risk.

- Post-Processing of Detection Data
  1. Event Categorization: Detected human presence and their activity (e.g., standing, sitting, walking) are categorized into predefined event classes. The system identifies whether the detected activity is a safety threat (e.g., standing near or on the tracks).
  2. Alert Generation: When a high-risk event is detected, the system triggers an alert to notify relevant stakeholders (e.g., railway personnel, control centers), providing details such as the location and nature of the detected activity.

3.3.4 Output

- Human Detection Results
  Bounding boxes with coordinates and confidence scores for detected humans.
  Bounding boxes with coordinates and confidence scores for Track detection with overlapping score for confidence.
- System Performance Metrics
  Detection accuracy metrics (precision, recall, F1 score).
  Feedback for model improvement.

3.4 Non-Functional Requirements

- Performance
  The system should process and detect human presence on railway tracks in real-time, with response times under 2 seconds to enable quick intervention.
- Reliability
  The system should ensure continuous monitoring, handle unexpected failures, and recover automatically to minimize downtime.
- Usability
  The interface should be intuitive, providing clear visual feedback and easy navigation for railway personnel with minimal technical expertise.
- Security
  Data transmission (video feeds, alerts) should be encrypted using secure protocols (e.g., HTTPS) to prevent unauthorized access and ensure privacy.
- Scalability
  The system should support large-scale deployment, handling multiple camera feeds simultaneously without performance degradation.
- Maintainability
  The software should be modular, well-documented, and support easy integration of updated detection models for long-term adaptability.

3.5 Design Constraints

- Standard Compliance
  The system should comply with AI safety and ethical guidelines, ensuring responsible deployment in real-world railway environments.
  It should follow industry standards for real-time object detection, adhering to frameworks like YOLO and OpenCV for image processing.
  The data collection and storage processes should align with privacy regulations, ensuring sensitive surveillance data is protected.

- Hardware Limitations
  The system should be optimized to run on edge devices (e.g., CCTV cameras, embedded systems) with limited computational power.
  The YOLO-based detection model should be lightweight, ensuring real-time inference on low-power GPUs or CPUs without performance bottlenecks.
  Offline functionality should be supported for areas with unstable internet connectivity, enabling local processing and delayed cloud updates.

This section concludes the software requirements for a railway safety monitoring system using deep learning-based human activity recognition. It covers the functional, non-functional, and interface requirements, ensuring real-time detection, scalability, and ease of deployment in railway environments.

CHAPTER 4: SYSTEM DESIGN

The System Design of our project provides an overview of the workflow architecture and data flow diagrams (DFD) of Level 0 and Level 1. Additionally, it describes the architecture of various YOLO model, which is used for real-time detection of unauthorized human presence on railway tracks. The design focuses on efficient processing, seamless integration with surveillance systems, and optimized deployment for real-world railway environments.

4.1 Architectural Design

The architectural design is divided into three primary modules, Data Collection and Preprocessing, Implementation of ANN/DL Algorithm, and Testing and Validation. Each module plays a crucial role in ensuring the overall effectiveness and efficiency of the system. Figure 4.1 is a detailed explanation of the architectural flow for each module.

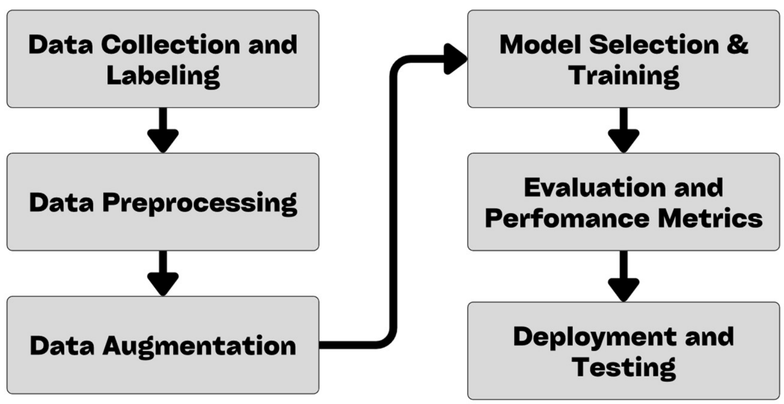

Figure 4.1 Architectural Block Diagram

The Figure 4.1 depicts a structured workflow for a deep learning project, highlighting essential stages such as data collection, preprocessing, model training, and evaluation. It emphasizes the importance of data augmentation, performance metrics, and deployment to ensure an effective and efficient deep learning system.

1. Data Collection, Labelling and Preprocessing

- To ensure the robustness and accuracy of the detection system, a comprehensive data collection strategy was adopted. The goal was to gather a diverse and representative dataset that accurately reflects real-world railway scenarios.
- High-resolution images were captured using a Samsung 50 MP GN5 sensor with an f/1.88 aperture to maintain clarity and detail.
- The median resolution of the images was set to 1080 × 1080 pixels, balancing image quality with computational efficiency.
- A total of 1,000 original images were collected, each featuring a person as the primary subject on railway tracks.

- To enhance model generalization, images were captured across different lighting conditions, including daytime, dawn, and dusk, ensuring the model could adapt to varying illumination levels. Additionally, diverse weather conditions such as clear sky, cloudy, and early morning mist were considered to simulate real-world scenarios. To further improve robustness, some images included partial occlusions, where the person was obstructed by objects like railway infrastructure (e.g., poles, fences) and vegetation (e.g., trees, bushes). This comprehensive data collection approach ensures that the model can effectively recognize human presence on railway tracks under a wide range of environmental factors.

Figure 4.2 Data Collection Strategy

This Figure 4.2 shows data collection strategy involved gathering information directly from railway tracks. Such an approach ensures the collection of accurate and contextually relevant data, essential for training robust machine learning models in railway-related applications.

2. Data Augmentation

- To enhance the dataset's diversity and improve model generalization, several data augmentation techniques were applied. These augmentations aimed to simulate real-world variations and ensure the model could handle different environmental conditions and perspectives. Horizontal flipping was used to create mirrored versions of the original images, helping the model recognize subjects from different orientations. 90-degree rotations were applied to account for different camera angles and accidental tilts during real-world deployments.
- To introduce color variability, 20% of the images were converted to grayscale, ensuring the model could detect humans even in low-color contrast scenarios. Further, brightness

adjustments of ±20% and exposure modifications of ±15% were introduced to simulate variations in lighting conditions, including shadows, bright sunlight, and artificial lighting.

- Additionally, rotating images by ±15 degrees helped account for slight camera tilts or dynamic scenarios where the subject might not always appear in a perfectly upright position. These augmentation techniques expanded the dataset to over 3700 images, significantly improving its robustness and adaptability. By exposing the model to a wide range of transformations, the augmentation process ensured that it could accurately detect humans on railway tracks under diverse lighting, weather, and occlusion conditions.
- The total images after data augmentation summed up to over 3700 images. Annotations by Class:
  Track: 1,819
  Standing: 1,782
  Sitting: 1,406
  Sleeping: 689

3. Model Selection and Training

- YOLO-V8, YOLO-V11, and YOLO-NAS are state-of-the-art object detection architectures that have been developed to enhance speed, accuracy, and efficiency in real-time applications. YOLO-V8 was chosen for its efficiency in processing high-resolution images quickly, making it suitable for the critical task of monitoring railway tracks for human presence. It introduces several advancements over previous versions, including an input layer designed for standardized image resizing (typically 640x640 pixels) to ensure consistency and optimal performance. The backbone of YOLO-V8, evolving from previous DarkNet architectures, integrates advanced convolutional layers, Cross-Stage Partial (CSP) connections, and a focus module, which together improve feature extraction and gradient flow, enhancing the model's ability to detect subtle details.
- Building on YOLO-V8, YOLO-V11 pushes the boundaries of object detection by incorporating transformer-based attention mechanisms and dynamic neural architecture search (NAS) for improved feature learning. YOLO-V11 enhances localization accuracy while maintaining high inference speeds, allowing for even more precise detection of humans on railway tracks under complex conditions such as occlusions, varying lighting, and environmental challenges.

- Meanwhile, YOLO-NAS represents a groundbreaking evolution in object detection, leveraging Neural Architecture Search (NAS) to automatically optimize the model architecture. This results in a highly efficient network that balances speed and accuracy, surpassing traditional manually designed architectures. YOLO-NAS significantly reduces computational overhead while maintaining high detection precision, making it ideal for deploying in edge computing environments where processing power is limited.

- Together, these architectures provide a comprehensive framework for real-time human detection on railway tracks. YOLO-V8 ensures fast and reliable detection, YOLO-V11 introduces advanced learning mechanisms for enhanced accuracy, and YOLO-NAS optimizes computational efficiency, making the system highly adaptable to real-world railway safety applications.

4. Evaluation and Performance Metrics

- Model evaluation is conducted using key performance metrics to assess the accuracy and reliability of the YOLO models in detecting human presence on railway tracks.
- Mean Average Precision (mAP) is used to measure the model's overall performance by averaging precision values at different recall levels, ensuring a balance between detecting objects correctly while minimizing false positives and false negatives. Precision quantifies the model's ability to correctly identify human presence without generating unnecessary false positives, which is crucial in railway monitoring to prevent unwarranted alerts.
- Recall measures the model's capability to detect all instances of human intervention on the tracks, directly impacting the system's safety and effectiveness in preventing accidents. To provide a balanced assessment, the F1 Score, which is the harmonic mean of precision and recall, is used to evaluate the model's overall reliability, particularly in scenarios where both false positives and false negatives have critical consequences.
- Together, these evaluation metrics provide a comprehensive analysis of the model's strengths and areas for improvement, ensuring its operational efficiency and robustness in real-time railway safety applications.

5. Deployment and Testing

- Deployment testing for the system involves validating the model's real-world performance in real-time environments, ensuring it operates efficiently and accurately on user devices. Roboflow's inference tool and Streamlit, a framework for building interactive web applications, are used for this process.

- Roboflow Inference: Roboflow provides a robust platform for deploying and testing computer vision models. After training the model, Roboflow allows for inference directly through their API or integrated into applications, streamlining the deployment process. The inference tool helps in validating the model by running predictions on real-time images uploaded by users, testing its ability to accurately detect and classify objects, such as human presence on the railway tracks. This step ensures that the model performs consistently and can handle the diversity of real-world scenarios, including different lighting and weather conditions.

- Streamlit for Deployment: Streamlit is used to create an interactive, user-friendly interface for deployment testing. Once the model is integrated with the application, Streamlit provides a simple way to visualize predictions, allowing testers and users to upload images, view the model's predictions, and assess its accuracy. Streamlit's real-time feedback allows for quick identification of issues or improvements needed in the deployment, such as ensuring seamless integration with Roboflow's inference API, managing server performance, and maintaining response times within acceptable limits.

- By combining Roboflow's powerful inference capabilities and Streamlit's intuitive interface, deployment testing becomes an efficient process to evaluate the model's functionality in dynamic environments. This helps ensure the system's readiness for real-world use, especially in applications like railway safety, where accurate, real-time predictions are critical.

4.2 Data Flow Diagrams

A Data Flow Diagram (DFD) is a visual representation that outlines how data flows through a system, showing the processes, data stores, external entities, and the movement of data between them. In the context of the Human Activity Recognition System for railway tracks, a DFD helps clarify the system's workflow, from image capture to activity detection and result presentation. It highlights the interaction between components like the detection model, image preprocessing, and user interface. This diagram aids in identifying system dependencies, optimizing performance, troubleshooting issues, and improving communication among stakeholders, making it an essential tool for ensuring the efficiency and accuracy of the real-time detection process.

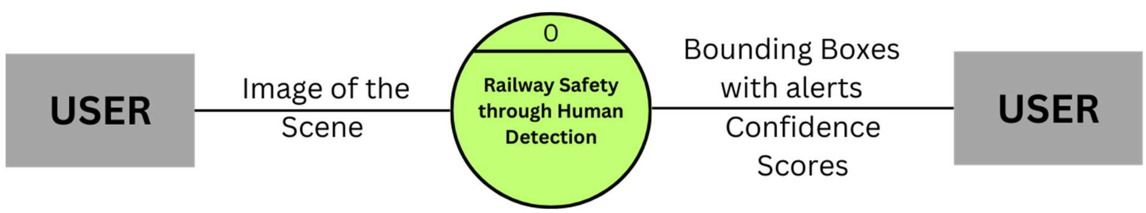

Figure 4.3 Data Flow Diagram Level 0

This Figure 4.3, a Level 0 Data Flow Diagram (DFD), also known as a context diagram. It provides a high-level overview of the system and its interactions with external entities.

- External Entities (Users)

The diagram shows two external entities labeled USER on both sides. These entities represent the users who will interact with the system. In the context of this project, the user could be a railway operator or a safety personnel, and they are the ones who provide inputs to and receive outputs from the system.

- System (Railway Safety through Human Detection):

The central process, shown as a circle, represents the system, which is titled Railway Safety through Human Detection. This indicates the core function of the system: to monitor and detect human presence on railway tracks for safety. The system processes the input data (images of the railway tracks) and generates output such as alerts or notifications based on human activity recognition.

- Data Flow

Input Data: The arrows pointing towards the system from the users represent input from the users. In this case, the system receives image data of the railway tracks, which may be uploaded by the users. The users provide images that are analyzed for detecting human activity on the tracks.
Output Data: The arrows pointing away from the system represent the output. The system processes the uploaded image and produces results, which could include alerts, notifications, or activity recognition information. These outputs are sent back to the users.
The diagram highlights the communication between the user and the core system. It shows that the system's purpose is to take inputs (images) from the users, process them for human detection, and then send outputs (results) to the users. The DFD helps stakeholders visualize the system at a high level and understand the basic interactions and data flow.

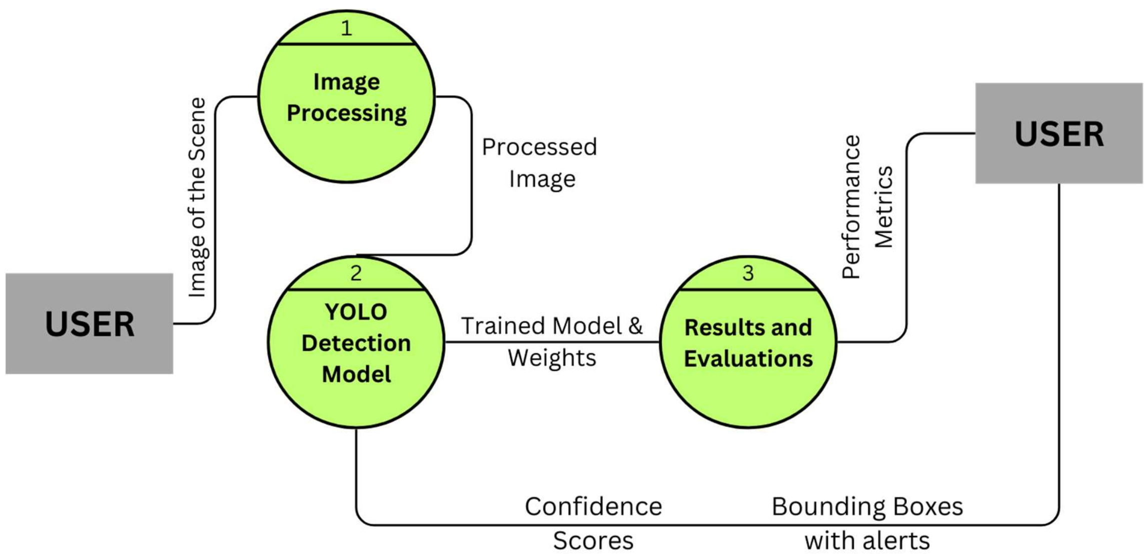

Figure 4.4 Data Flow Diagram Level 1

This Figure 4.4 represents a Level 1 Data Flow Diagram (DFD) for the Railway Safety through Human Detection system. It provides a more detailed view of the processes involved in the system's operation, focusing on the main components that handle the data flow.

- Image Processing (Process 1)

This is the first step in the system where the user uploads an image of the railway tracks. The image undergoes preprocessing, which can include resizing, noise reduction, normalization, etc., to prepare it for further analysis.

- YOLO Detection Model (Process 2)

After the image is processed, the YOLO model performs object detection to identify human presence or activity on the tracks. This process uses the trained model to detect and classify the human presence in the uploaded image. The model can detect different types of activities, such as sitting, standing, or lying on the tracks.

- Results and Evaluations (Process 3)

After the detection, the results are processed and evaluated. This step involves generating alerts or notifications based on the detection results. It may also involve a performance evaluation of the detection system using metrics such as precision, recall, and F1 score. The evaluation ensures that the system is working efficiently and accurately.

- Data Flow

The user uploads images to the system, and the images flow into the Image Processing process. The processed images are then passed to the YOLO Detection Model for human detection and activity classification.
The results from the detection model are sent to Results and Evaluations, where alerts or performance metrics are generated and shown back to the user.

The Level 1 DFD provides a detailed look at how the system processes the input data (images) step by step. It clearly defines the major components involved and how data flows between them, offering a better understanding of the system's operations.

4.3 Architecture of YOLO Models

YOLO (You Only Look Once) is a family of object detection models designed to detect and localize multiple objects in an image or video in a single pass, which makes them extremely efficient for real-time applications. Unlike traditional object detection models that perform classification and localization in separate steps, YOLO combines both tasks into a single network, making it faster and more efficient.

4.3.1 YOLO-v8

YOLO (You Only Look Once) is a state-of-the-art, real-time object detection system that has evolved through several versions, with YOLOv8 being one of the latest iterations. YOLOv8 builds upon the strengths of its predecessors, offering improvements in accuracy, speed, and flexibility.

YOLOv8 is designed to detect objects in images and videos with high accuracy and speed. It is widely used in applications such as autonomous driving, surveillance, and robotics. YOLOv8 is part of the YOLO family, which is known for its single-stage detection approach, meaning it predicts bounding boxes and class probabilities directly from full images in one forward pass through the network.

The architecture is as follows:

- Backbone

The backbone is responsible for feature extraction from the input image. YOLOv8 uses a modified version of the CSPDarknet53 architecture, which is a variant of Darknet. CSPDarknet53 incorporates Cross Stage Partial (CSP) connections to enhance gradient flow and reduce computational cost.
The backbone consists of multiple convolutional layers, batch normalization, and activation functions (usually Leaky ReLU or Mish).

- Neck

The neck of the network is designed to aggregate features from different layers of the backbone. YOLOv8 uses a modified Path Aggregation Network (PANet) as its neck.
PANet enhances the feature pyramid by adding bottom-up paths, allowing for better fusion of features from different scales. This helps in detecting objects of various sizes more effectively.

- Head

The head is responsible for generating the final output, which includes bounding boxes, objectness scores, and class probabilities.
YOLOv8 uses a decoupled head, which separates the classification and regression tasks. This allows for more precise predictions and better performance.
The head consists of multiple convolutional layers that predict the bounding box coordinates (x, y, width, height), objectness score (probability that an object exists in the box), and class probabilities.

Functioning of the Architecture

- Input Processing

The input image is resized to a fixed dimension (e.g., 640x640 pixels) and normalized.
The image is then passed through the backbone for feature extraction.

- Feature Extraction

The backbone processes the image through multiple convolutional layers, extracting features at different scales.
These features are then passed to the neck for further processing.

- Feature Aggregation

The neck aggregates features from different layers of the backbone using PANet.
This results in a feature pyramid that captures information at multiple scales, enabling the detection of objects of various sizes.

- Prediction

The head takes the aggregated features and generates predictions for bounding boxes, objectness scores, and class probabilities.
YOLOv8 uses anchor boxes to predict bounding boxes. These anchor boxes are pre-defined shapes that help in predicting the size and aspect ratio of objects.

The model predicts offsets for the anchor boxes to refine the bounding box coordinates.

- Post-Processing

Non-Maximum Suppression (NMS) is applied to filter out overlapping bounding boxes and retain the most confident predictions.
The final output includes the bounding box coordinates, class labels, and confidence scores for each detected object.

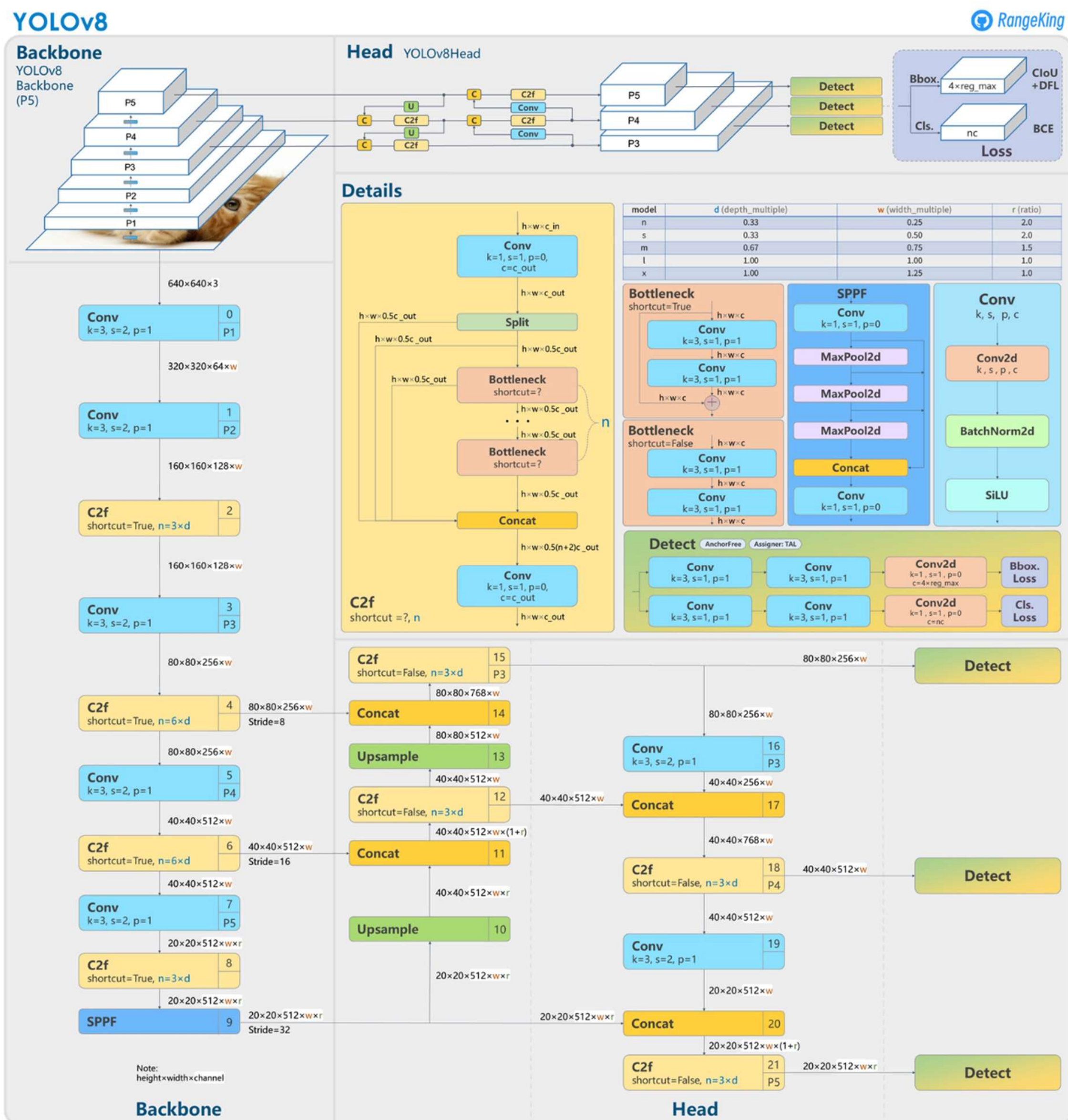

Figure 4.5 Architecture of YOLO-v8 [21]

Figure 4.5 represents the architecture of YOLO-v8 by ultralytics, the important parts such as the neck, backbone and the head can be clearly visualised from this figure. YOLOv8 is trained using a combination of classification and regression losses. The loss function typically includes: Localization Loss: Measures the accuracy of the predicted bounding box coordinates.

Confidence Loss: Measures the accuracy of the objectness score.

Classification Loss: Measures the accuracy of the predicted class probabilities. Data augmentation techniques such as random cropping, flipping, and color jittering are used to improve the model's robustness.

Advantages of YOLO-v8

- Speed: YOLOv8 is designed for real-time detection, making it suitable for applications requiring fast processing.
- Accuracy: The use of advanced techniques like CSPDarknet53 and PANet improves detection accuracy.
- Flexibility: YOLOv8 can be easily adapted to different tasks and datasets.

4.3.2 YOLO-NAS

The model uses techniques like attention mechanisms, quantization aware blocks, and reparametrization at inference time. These techniques help YOLO-NAS to identify objects of varying sizes and complexities better than other detection models.

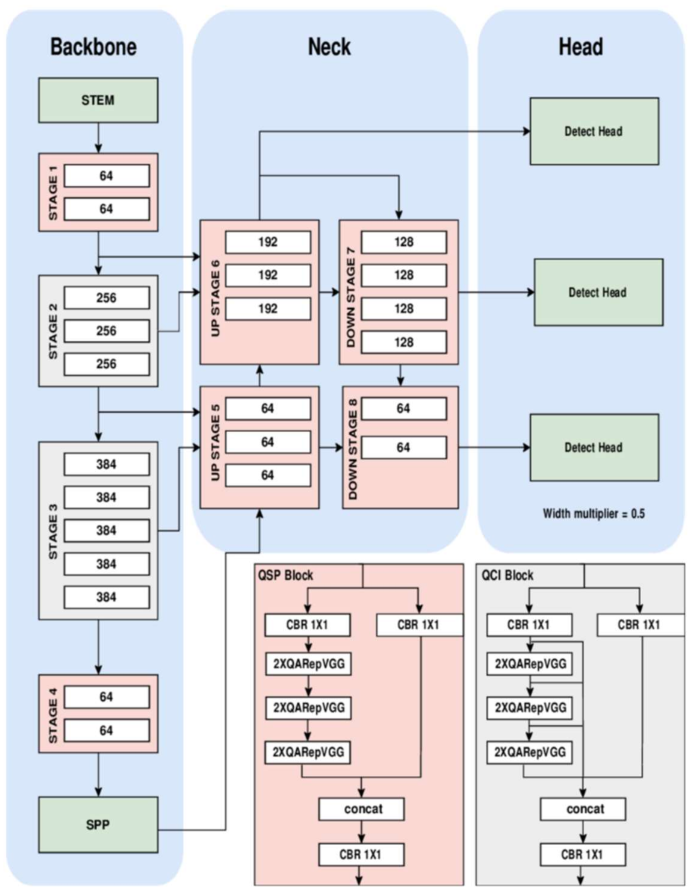

Figure 4.6 Architecture of YOLO-NAS [22]

The YOLO-NAS architecture consists of three main components, Backbone,Neck and Head as shown in Figure 4.6. The constituents of this system have been crafted and optimized using the NAS method. It results in a unified and robust object detection system.

YOLO-NAS (You Only Look Once - Neural Architecture Search) is an advanced version of the YOLO architecture that integrates Neural Architecture Search (NAS) techniques into the model. The key idea behind YOLO-NAS is to optimize the architecture of the YOLO model automatically using NAS methods, improving detection performance across various tasks and datasets.

- Neural Architecture Search (NAS) Integration

The major distinction between YOLO-NAS and earlier YOLO versions is that NAS is used to automatically search for the best possible neural architecture. This means that instead of manually designing the layers, YOLO-NAS searches for the most optimal network configuration.

NAS uses reinforcement learning or evolutionary algorithms to explore and optimize different architectures.

- Backbone Architecture

Backbone: In YOLO-NAS, the backbone typically uses a CSPNet (Cross-Stage Partial Network) or EfficientNet as the backbone to efficiently extract features from input images. These backbones are chosen for their ability to maintain high accuracy while reducing computation, which is critical for real-time detection tasks.

The backbone in YOLO-NAS is specifically tuned through NAS to improve feature extraction, ensuring it is highly effective for object detection in diverse environments.

- Neck and Head

Neck: YOLO-NAS uses FPN (Feature Pyramid Networks) or PANet (Path Aggregation Network) for the neck to ensure the model can generate multi-scale feature maps. This improves the detection of objects of varying sizes, from small to large.

Head: The head of the network is responsible for predicting the bounding boxes, class labels, and objectness score. The head architecture in YOLO-NAS is optimized for efficient prediction, and it might involve decoupled heads to predict object classification and localization separately, which improves the precision.

- NAS-based Optimization

The NAS process is used to automate the search for optimal layer types, kernel sizes, and connections, ensuring the model is both accurate and computationally efficient.

This search process leverages state-of-the-art NAS algorithms to discover architectures that balance accuracy, speed, and resource consumption, making YOLO-NAS particularly suitable for mobile and edge applications where computational resources are limited.

- Customizable Layers

YOLO-NAS allows flexibility in terms of layer types and configurations, including convolutional layers, depthwise separable convolutions, and attention mechanisms, which enhance the model's ability to focus on important features and discard irrelevant ones.

- Activation Functions

YOLO-NAS incorporates advanced activation functions like Mish or Swish to improve non-linear learning capability, helping the model perform better on complex tasks.

- Loss Function

YOLO-NAS uses a combined loss function, incorporating classification loss, bounding box regression loss, and objectness loss. This ensures that the model not only detects the presence of objects but also refines the bounding box predictions.

CHAPTER 5: IMPLEMENTATION

For the project on railway safety through human activity recognition, the design and implementation involve developing a real-time deep learning-based solution utilizing the YOLO-V8 architecture. The system is designed to detect human activity on railway tracks efficiently and accurately, enabling rapid response to potential safety hazards. The code is organized into various sections, each handling specific tasks such as image processing, model training, and real-time inference. The development environment is set up in a Python notebook format, using libraries like OpenCV for image processing, PyTorch or TensorFlow for training the YOLO-V8 model. The necessary libraries and dependencies were installed within a Python environment, ensuring smooth operation throughout the entire development and deployment cycle.

5.1 Code Snippets and Custom Training

5.1.1 Data Augmentation and Preprocessing

- Augmentation is the process of artificially expanding a dataset by applying various transformations to existing images. This improves model robustness by simulating real-world variations such as changes in lighting, perspective, noise, and object positioning. Roboflow provides built-in augmentation techniques that can be applied while generating datasets for training. It helps improve the generalization ability of machine learning models.

- Before training, the dataset undergoes preprocessing to enhance model performance and ensure data consistency. Preprocessing involves steps such as resizing, normalization, and format conversion, making the data suitable for YOLOv8 training.

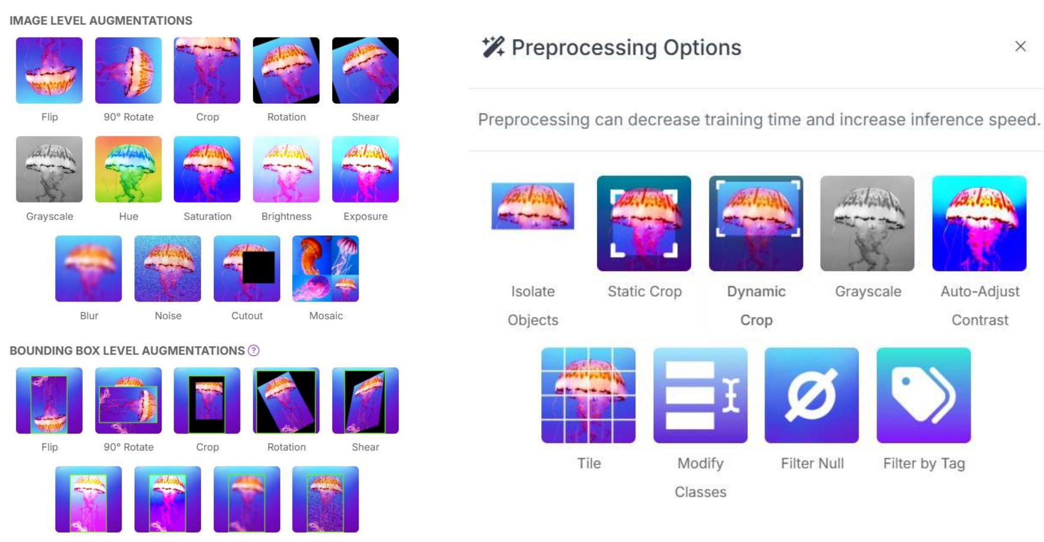

Figure 5.1 Data Augmentation and Preprocessing Techniques

Figure 5.1 outlines various image-level augmentation techniques such as flipping, rotation, and adjusting brightness, which enhance dataset diversity for training. It also lists preprocessing options like cropping and grayscale conversion to optimize training efficiency and model performance.

5.1.2 Dataset Import and Libraries

```
[ ] from ultralytics import YOLO
    from IPython.display import display, Image

    !mkdir {HOME}/datasets
    %cd {HOME}/datasets

    !pip install roboflow

    from roboflow import Roboflow
    rf = Roboflow(api_key="the_api_key")
    project = rf.workspace("train-glndj").project("track-xyz")
    version = project.version(2)
    dataset = version.download("yolov8")
```

Figure 5.2 Dataset Import and Libraries Installation

This code snippet in Figure 5.2 demonstrates setting up a dataset for a YOLOv8 model using the Roboflow API, including importing necessary libraries and downloading the dataset. It highlights the integration of Roboflow for dataset management and preparation, streamlining the workflow for training object detection models.
The training process begins with mounting Google Drive to access datasets. The Roboflow API is used to fetch a YOLOv8-compatible dataset, ensuring seamless integration.

5.1.3 Model Training

Google Colab, with its free access to powerful T4 GPUs, was used to train the YOLOv8 model for detecting human interference on railway tracks. The T4 GPU, featuring 2,560 CUDA cores and 16 GB of GDDR6 memory, significantly accelerated the training process, especially for large datasets. The training pipeline included data pre-processing such as normalizing pixel values, resizing images, and converting annotations to the YOLO-V8 format. Data augmentation techniques like random cropping, rotation, flipping, and color adjustments were employed to enhance the diversity of the training set and improve the model's robustness to varying conditions.

The training environment utilized Google Colab, leveraging its T4 GPU for training. In this work, training set accounted for 70% of the dataset, validation set for 20%, and testing set for 10% with 2,635, 754, and 377 images, respectively. YOLO-V8 models were used for training and produced the best accuracy. The parameters used in the training process for the model are shown in Table 5.1.

Batch Size	16
Epoch	50
Learning Rate	0.00125
Optimizer	AdamW

Table 5.1 The Model Parameters for Fine-tuning

```
%cd {HOME}

!yolo task=detect mode=train
model=yolov8s.pt data={dataset.location}/data.yaml
epochs=50
imgsz=1080
plots=True
```

```
Image sizes 1088 train, 1088 val
Using 2 dataloader workers
Logging results to runs/detect/train
Starting training for 50 epochs...

      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
       1/50      11.5G      1.571      2.682      1.848         28       1088: 100% 165/165 [02:44<00:00,  1.00it/s]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 24/24 [00:24<00:00,  1.01s/it]
                   all        754       1129      0.569      0.468      0.465      0.237

      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
       2/50      11.6G      1.552      1.859      1.817         22       1088: 100% 165/165 [02:41<00:00,  1.02it/s]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 24/24 [00:21<00:00,  1.12it/s]
                   all        754       1129      0.692      0.604      0.602      0.307

      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
       3/50      11.6G      1.564      1.864      1.841         36       1088: 100% 165/165 [02:41<00:00,  1.02it/s]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 24/24 [00:20<00:00,  1.19it/s]
                   all        754       1129      0.555      0.507      0.487      0.244
```

Table 5.3 Model Training with parameters in Table 5.1

This Figure 5.3 shows the training process of a YOLOv8 model for object detection, including metrics like box loss, class loss, and detection accuracy over multiple epochs. It provides detailed performance statistics, such as precision (P) and recall (R), along with mean average precision (mAP) values, indicating the model's learning progress and effectiveness as the epochs increase while training.

5.1.3 Model Validation

```
%cd {HOME}

!yolo task=detect mode=val
 model={HOME}/runs/detect/train/weights/best.pt
 data={dataset.location}/data.yaml
```

```
/content
Ultralytics YOLOv8.0.196 🚀 Python-3.10.12 torch-2.3.1+cu121 CUDA:0 (Tesla T4, 15102MiB)
Model summary (fused): 168 layers, 11127132 parameters, 0 gradients, 28.4 GFLOPs
val: Scanning /content/datasets/ravi-train-8/valid/labels.cache... 754 images, 0 backgrounds, 0 corrupt: 100% 754/754 [00:00<?, ?it/s]
WARNING ⚠ Box and segment counts should be equal, but got len(segments) = 795, len(boxes) = 1129. To resolve this only boxes will be used and all segments will
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 48/48 [00:27<00:00,  1.73it/s]
                   all        754       1129      0.945       0.81      0.923      0.694
               sitting        754        301      0.948      0.837      0.954      0.673
              sleeping        754        140      0.993      0.829       0.95      0.673
              standing        754        333      0.881       0.62        0.8      0.559
                 track        754        355      0.958      0.954      0.986       0.87
Speed: 2.5ms preprocess, 24.2ms inference, 0.0ms loss, 2.4ms postprocess per image
Results saved to runs/detect/val
💡 Learn more at https://docs.ultralytics.com/modes/val
```

Figure 5.4 Validation of the Model

The validation of this system is demonstrated through performance metrics inf Figure 5.4, such as mAP, precision, recall, and F1 score, which assess the model's accuracy in detecting human presence on railway tracks. The figure showcases the effectiveness of YOLO-V8 in minimizing false positives and false negatives, ensuring reliable real-time detection for railway safety.

In YOLOv8, model validation is a crucial step in assessing the performance and generalization ability of the trained model. The dataset is typically divided into training, validation, and test sets, with the validation set used exclusively for model evaluation during training. Validation involves computing key metrics such as Mean Average Precision (mAP), which measures the precision across different recall values and provides a comprehensive view of the model's performance. Precision and recall are also essential metrics, helping assess the accuracy of predictions and the model's ability to detect all relevant objects, respectively. The F1-score, which balances precision and recall, is another critical metric for overall evaluation. During training, the model is evaluated on the validation set after each epoch, and based on these metrics, adjustments such as hyperparameter tuning are made to improve performance. A confusion matrix may also be generated to further analyze misclassifications, while loss curves help detect overfitting or underfitting. After training, the model is evaluated on the test set to ensure it is ready for deployment. Through these validation techniques, YOLOv8 ensures that the model is robust and capable of accurately detecting human presence on railway tracks, even in real-world scenarios.

5.1.4 Roboflow Workflow

A Roboflow Workflow is a pipeline that chains multiple machine learning models together to process images in a structured sequence. It allows you to combine object detection, classification, and segmentation models into a single API call, making your computer vision tasks more efficient and automated.

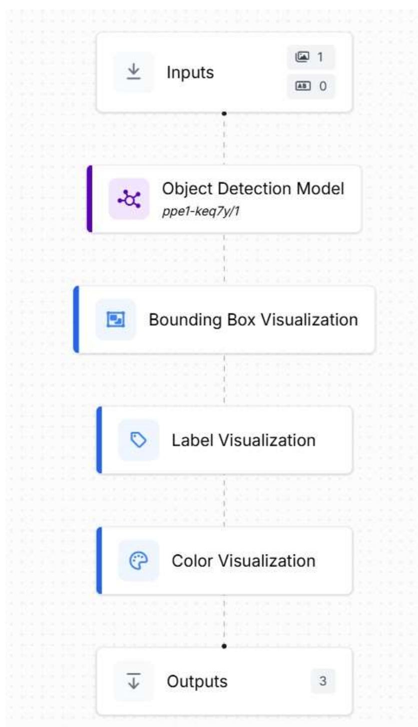

Figure 5.5 Workflow Pipeline

5.1.5 Local Inferencing

In Roboflow, local inferencing refers to running a trained machine learning model directly on a local machine or device (e.g., laptop, desktop, or edge device) rather than making API calls to the Roboflow cloud to perform inference. This is particularly useful when you want to process data without relying on an internet connection, maintain data privacy, or reduce latency by avoiding cloud communication.

```python
from inference_sdk import InferenceHTTPClient

client = InferenceHTTPClient(
    api_url="https://detect.roboflow.com",
    api_key="2cs75y0EMpV5mlce5Xa2"
)

result = client.run_workflow(
    workspace_name="train-glndj",
    workflow_id="detect-count-and-visualize",
    images={
        "image": "YOUR_IMAGE.jpg"
    },
    use_cache=True
)
```

Figure 5.6 Local Inferencing using Roboflow API Call

This Python script in Figure 5.6 uses Roboflow's local inference server with Docker to process a video stream using a predefined workflow. It initializes an InferencePipeline object that continuously processes frames and returns predictions.

5.2 Results and Discussion

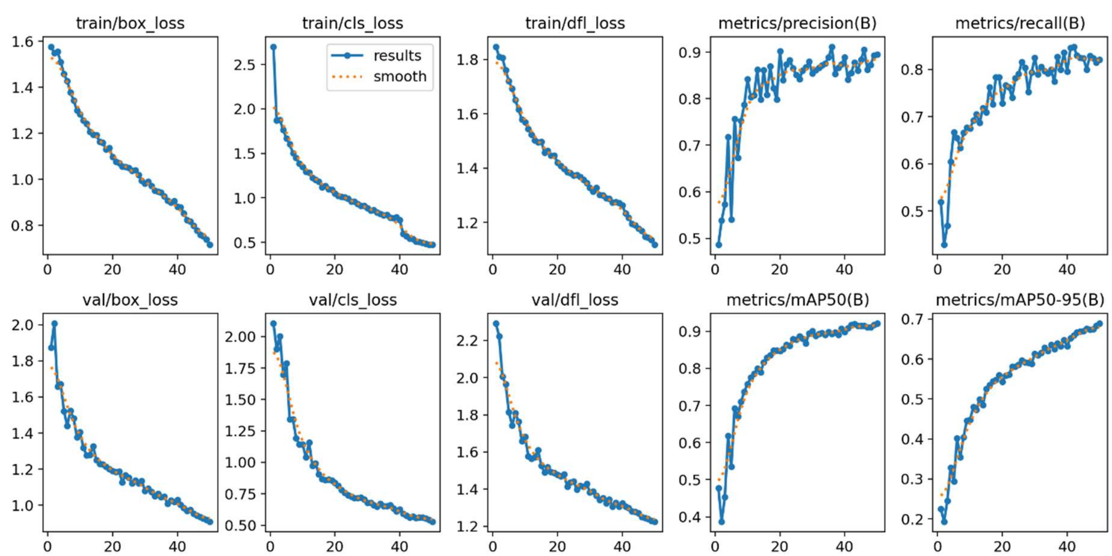

Figure 5.7 Results and Performance Metrics of YOLO v8

The figure 5.7 , training losses (box_loss, cls_loss, dfl_loss) show a decreasing trend over the epochs. This indicates that the model is learning and improving its ability to predict bounding boxes, classify objects, and manage distribution focal loss effectively. The model is converging well during training, which is a positive sign of effective learning.

Both precision and recall show an increasing trend over the epochs. Precision measures the accuracy of the positive predictions, while recall measures the fraction of true positives detected. The model is becoming more accurate in its predictions and better at detecting objects, which is crucial for object detection tasks.

Model	YOLO-V8
mAP50	92.3
mAP50-95	69.5
Recall	81.1
Precision	94.4

Table 5.2 Metrics of YOLO-v8

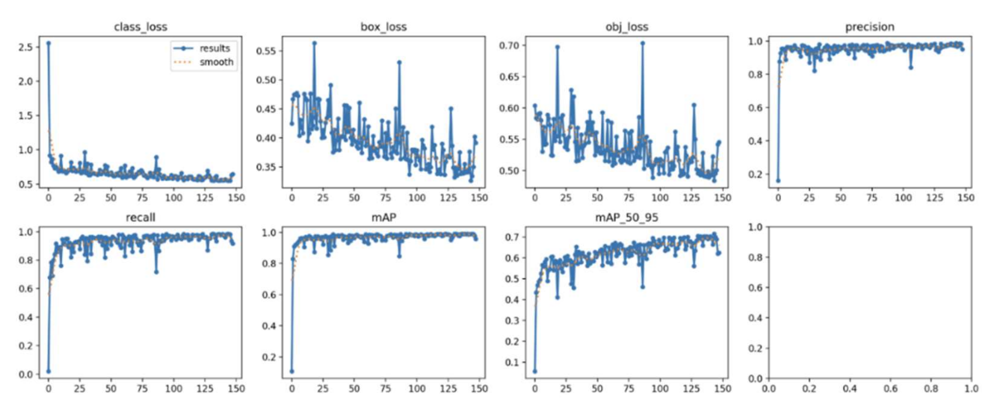

Figure 5.8 YOLO-NAS Metrics

The figure 5.8, training losses (box_loss, cls_loss, dfl_lossfor YOLO-NAS . This indicates that the model is learning and improving its ability to predict bounding boxes, classify objects, and manage distribution focal loss effectively. The model is converging well during training, but the high variations in losses indicate the model isn't able to generalise properly.

A few images were randomly selected for testing to assess the predictive capabilities of the trained models. Railway accidents are more common in recent days and the use of AI technologies helps in preventing it. In this project, we have created a unique dataset. The dataset is used to identify the different human activities through AI which enhances the railway safety and seamless commuting. The YOLO-V8 model was selected as the optimal choice for this work and shown in-depth analysis. The proposed method achieved 93.00% accuracy. The proposed system can be extended with the sharing of the autonomous notifications with the loco-pilots and nearest railway stations. Additionally, the dataset is now limited to a small number of activities, and it would be beneficial to include a greater variety of activities

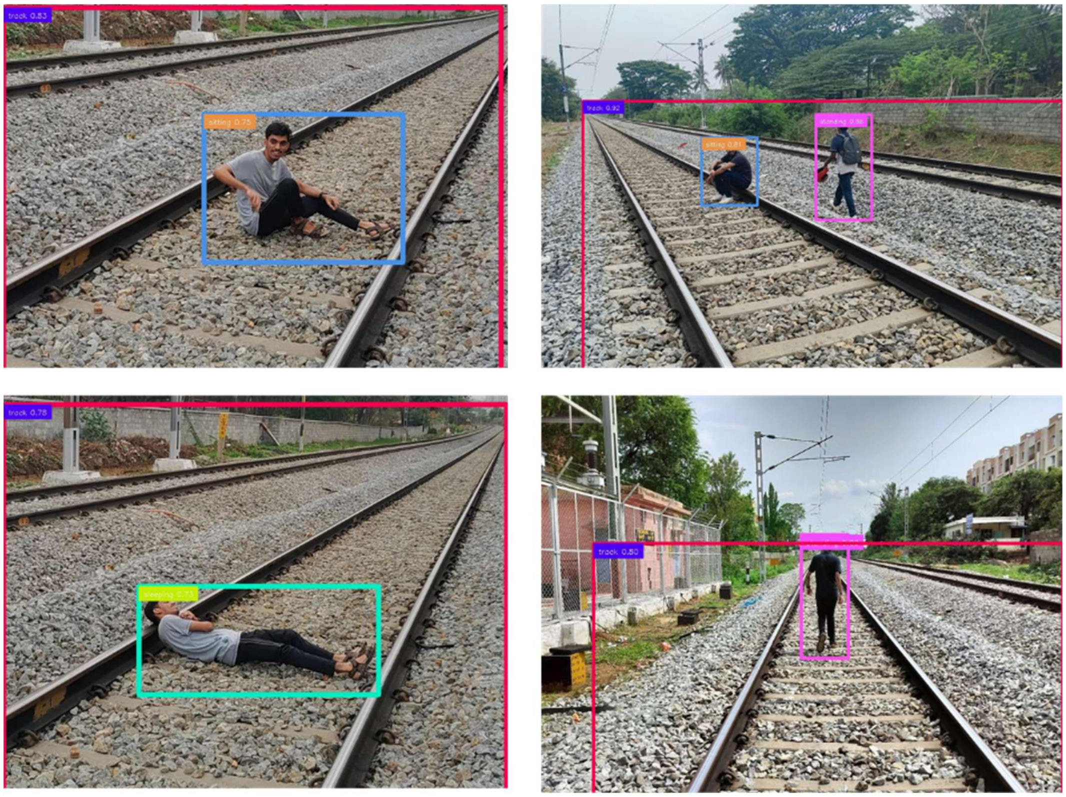

Figure 5.9 Final Results of the Model with Confidence Score

Figure 5.9 illustrates the final output of the system, showcasing the effectiveness of the YOLO-V8 model in detecting human presence on railway tracks. The system identifies and classifies objects by drawing precise bounding boxes around detected classes, accompanied by a confidence score that indicates the reliability of the detection. Each confidence score is associated with a specific class, ensuring accurate identification of human activities such as standing, sitting, or lying on the tracks. Furthermore, if multiple detections overlap or if the system recognizes a potential hazard, an alert mechanism may be triggered, providing real-time warnings to enhance railway safety. This visualization highlights the system's capability to function efficiently in real-world scenarios, reinforcing its role in preventing accidents and improving railway surveillance.

CONCLUSION

The project, focused on enhancing railway safety through human activity recognition, presents a comprehensive deep learning-based solution for real-time detection of unauthorized human presence on railway tracks. By leveraging the power of YOLOv8, an advanced object detection model, the project developed an efficient and accurate system capable of detecting various human activities such as sitting, lying, and standing on the tracks, in a wide array of environmental conditions. The extensive dataset used in this project, comprising high-resolution images taken under varying lighting, weather, and occlusion scenarios, provided the model with the necessary variability to handle real-world complexities.

Through the use of advanced techniques like data augmentation and precise annotation, the dataset was further enriched to ensure the model's robustness. By incorporating models such as YOLOv8 and validating them with metrics like precision, recall, F1-score, and mAP, the system's effectiveness was thoroughly evaluated. Google Colab, with its powerful T4 GPU, was instrumental in training the model efficiently, allowing for the fast processing of large datasets and the fine-tuning of model parameters. The deployment testing using tools like Roboflow and Streamlit further solidified the system's readiness for real-time use.

The results indicate that YOLOv8, with its remarkable detection capabilities, can significantly improve safety on railways by enabling rapid human activity detection, thereby reducing the risk of accidents and fatalities caused by unauthorized track intrusions. This project highlights the potential of AI and computer vision in transforming safety-critical applications, particularly in environments where timely intervention is crucial. The robust and scalable design ensures that the system can be deployed in various railway networks, offering a promising solution for safeguarding human life. Future improvements could involve the integration of more sophisticated models and additional features such as multi-camera tracking and predictive analytics to further enhance the system's performance and reliability.

FUTURE ENHANCEMENT

- Future enhancements for the railway safety system could focus on further improving the model's accuracy, robustness, and adaptability to a wider range of real-world conditions. One key area of improvement could be the integration of more advanced models, such as the next iterations of YOLO or hybrid architectures combining the strengths of object detection and activity recognition, enabling more precise detection and classification of human activities under challenging conditions.

- Additionally, expanding the dataset to include more diverse scenarios, such as various types of human attire, different train types, and more complex occlusions, would further strengthen the model's ability to handle a broader range of real-world situations.

- Another potential enhancement could involve multi-camera integration, where data from multiple cameras installed along railway tracks are combined to provide a more comprehensive view of the environment. This could help in detecting human presence across larger sections of track and overcoming the limitations of individual camera angles or partial occlusions.

- Incorporating real-time tracking and predictive analytics could also elevate the system's capabilities by forecasting potential risk situations, such as predicting whether a person is moving towards a dangerous zone on the track, and allowing for faster response times.

- Additionally, the system could be optimized for edge computing, enabling real-time processing directly on low-power devices installed along the tracks, without the need for constant cloud connectivity. This would enhance the system's reliability and reduce latency, especially in remote or rural areas with limited internet access. Integration of automated alert systems that can send instant notifications to railway authorities, train operators, or surveillance personnel upon detecting an intrusion would further enhance safety by facilitating prompt intervention. Lastly, extending the model's scope to include other potential hazards, such as animals or fallen objects on the tracks, would broaden the system's applicability and increase its value for comprehensive railway safety monitoring. By continuously refining the model and incorporating new features, the system can evolve to meet the growing demands of modern railway safety management, ensuring that it remains an effective tool in preventing accidents and saving lives.

BIBLIOGRAPHY

[1] Cao, Zhiwei & Qin, Yong & Jia, Limin & Xie, Zhengyu & Gao, Yang & Wang, Yaguan & Li, Ping & Yu, Zujun. (2024). Railway Intrusion Detection Based on Machine Vision: A Survey, Challenges, and Perspectives. IEEE Transactions on Intelligent Transportation Systems. PP. 1-22. 10.1109/TITS.2024.3412170.

[2] You Only Look Once: Unified, Real-Time Object Detection Joseph Redmon , Santosh Divvala, Ross Girshick , Ali Farhadi University of Washington , Allen Institute for AI , Facebook AI Research, 2016, arXiv:1506.02640

[3] YOLOv4: Optimal Speed and Accuracy of Object Detection Alexey Bochkovskiy, Chien-Yao Wang Institute of Information Science Academia Sinica, Hong-Yuan Mark Liao Institute of Information Science Academia Sinica, Taiwan, 2020, arXiv:2004.10934

[4] Khoudour, Louahdi & Bruyelle, Jean-Luc & El-Miloudi, El-Koursi. (2012). Intelligent Surveillance System Based on Stereo Vision for Level Crossings Safety Applications. 10.5772/30400.

[5] Farrington-Darby, Trudi & Pickup, Laura & Wilson, John. (2015). Safety culture in railway maintenance. Safety Science. 43. 39-60. 10.1016/j.ssci.2004.09.003.

[6] Haryono, Hollyana & Hidayat, Fadhil. (2022). Trespassing Detection using CCTV and Video Analytics for Safety and Security in Railway Stations. 01-04. 10.1109/ICISS55894.2022.9915245.

[7] Y. Yang, Z. Liu, J. Chen, H. Gao and T. Wang, "Railway foreign object intrusion detection Using UAV images and YOLO-UAT," in IEEE Access, 2025 doi: 10.1109/ACCESS.2025.3533304.

[8] Q. Zhao, S. Fang, Y. Li, H. Shang, H. Zhang and T. Shen, "Track Defect Detection Based on Improved YOLOv5s," in IEEE Transactions on Industrial Informatics, 2025 doi: 10.1109/TII.2024.3523593.

[9] S. Gui et al., "Enhanced Concealed Object Detection Method for MMW Security Images based on YOLOv8 framework with ESFF and HSAFF," in IEEE Sensors Journal, 2025 doi: 10.1109/JSEN.2024.3524441.

[10] Z. Zhao et al., "Automatic Potential Safety Hazard Evaluation System for Environment Around High-Speed Railroad Using Hybrid U-Shape Learning Architecture," in IEEE Transactions on Intelligent Transportation Systems, vol. 26, no. 1, pp. 1071-1087, Jan. 2025, doi: 10.1109/TITS.2024.3487592.

[11] S. Cao et al., "Railway intruding pedestrian image sequence generation method with pose transfer techniques," 10th International Symposium on Test Automation & Instrumentation (ISTAI 2024), Shenzhen, China, 2024

[12] J. Zhang, Q. Guan, J. Liu, Y. Huang and J. Guo, "RNaAE: A Novel Approach for Identifying Unseen Railway Anomalies," 2024 IEEE 17th International Conference on Signal Processing (ICSP), Suzhou, China, 2024, pp. 389-394, doi:

10.1109/ICSP62129.2024.10846075.

[13] Y. Ma, S. Xu and H. Jia, "Deep Arc Detection for High-Speed Railways: An Improved YOLOv5 Approach with SEC3 Attention and BiFPN Fusion," 2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Kuching, Malaysia, 2024, pp. 4243-4249, doi: 10.1109/SMC54092.2024.10830975.

[14] P. V. Mane, J. Poruthur, A. Pawar and M. Patil, "Automated Track Monitoring System using Object detection and Ultrasonic Sensor for Railway safety," 2024 5th International Conference on Data Intelligence and Cognitive Informatics (ICDICI), Tirunelveli, India, 2024, pp. 384-389, doi: 10.1109/ICDICI62993.2024.10810897.

[15] L. Lian et al., "A Continuous Autonomous Train Positioning Method Using Stereo Vision and Object Tracking," in IEEE Intelligent Transportation Systems Magazine, 2024 doi: 10.1109/MITS.2024.3509977.

[16] L. Liu, B. Zhou, G. Liu, D. Lian and R. Zhang, "Yolo-Based Multi-Model Ensemble for Plastic Waste Detection Along Railway Lines," IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium, Kuala Lumpur, Malaysia, 2022.

[17] Q. Chen, L. Liu, R. Han, J. Qian and D. Qi, "Image identification method on high speed railway contact network based on YOLO v3 and SENet," 2019 Chinese Control Conference (CCC), Guangzhou, China, 2019, pp. 8772-8777, doi: 10.23919/ChiCC.2019.8865153.

[18] Y. Ma, J. Fang, J. Zhao and Q. Zhang, "Research on the Application of YOLO v3 in Railway Intruding Objects Recognition," 2022 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), Dalian, China, 2022, pp. 583-586, doi: 10.1109/ICAICA54878.2022.9844532.

[19] S. Wang, C. Du, Y. Meng, F. Gao and Y. Cai, "Automatic Inspection Method for YOLOv5s Railway Freight Vehicles Based on Channel Pruning," 2023 International Conference on Intelligent Communication and Networking (ICN), Changzhou, China, 2023, pp. 307-313, doi: 10.1109/ICN60549.2023.10426037.

[20] P. B. Hemanthkumar and R. M. Benjamin, "System for Identifying and Avoiding Obstructions on Railway Tracks in Forest Areas," 2024 10th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2024, pp. 2409-2414, doi: 10.1109/ICACCS60874.2024.10716985.

[21] https://github.com/ultralytics/ultralytics/issues/189

[22] https://www.researchgate.net/figure/YOLO-NAS-l-Architecture-57_fig1_377694185