Utkal University Journal of Computing & Communications,
Vol. 2, Issue. 1 (2024) *e-ISSN:2584-203X*

Convolutional Neural Network based Railway Track Hazard Detection System

Varad Pramod Nimbalkar[1], Vishu Kumar[1], Vedansh Gohil[1] , Siddharth Bharadwaj[1] , Akshita Chanchlani[1] , Kulamala Vinod Kumar[1]

[1]Department of Computer Engineering and Technology, Dr. Vishwanath Karad MIT World Peace University Pune, Maharashtra, India

akshita.s.chanchlani@gmail.com

Abstract: Ensuring railway safety is critical for preventing accidents and saving lives. This paper proposes an advanced deep learning-based system for real-time hazard detection on railway tracks. Using pre-trained Convolutional Neural Networks (CNN), including ResNet152V2, EfficientNetB7, and InceptionV3, the system identifies obstacles and recognizes railway signals. Extensive experiments on custom datasets demonstrate robust performance under various conditions, achieving a detection accuracy of over 90%. The proposed system is scalable for real-world applications with promising results in deployment scenarios.

Keywords: Railway safety, obstacle detection, Deep Learning, signal recognition, CNN, ResNet, EfficientNet, InceptionV3

1. Introduction

Railways are one of the most vital modes of transportation, connecting urban and rural areas while facilitating economic growth worldwide. However, the complexity of modern railway networks and increasing train frequencies present substantial challenges in ensuring safety. Accidents involving collisions, derailments, and track obstructions are often attributed to undetected hazards and missed signal interpretations. These incidents not only lead to significant financial losses but also endanger human lives, highlighting the critical need for robust safety mechanisms. Traditional methods of hazard detection and signal recognition primarily rely on human operators and mechanical systems. While these approaches have served well in the past, they are prone to human error and mechanical failures, particularly under adverse environmental conditions such as low light, fog, or heavy rain. Conventional systems often struggle with small or partially visible obstacles and fail to adapt to the dynamic nature of railway operations. CNN are explored in various image processing scenarios such as facial recording system for attendance assimilation [1]. Recent advancements in artificial intelligence (AI) and deep learning have revolutionized computer vision, enabling machines to detect and recognize objects with remarkable precision. The advent of Convolutional Neural Networks (CNNs), including models such as ResNet, EfficientNet, and Inception, has significantly improved the accuracy and efficiency of object detection systems. However, most existing AI solutions focus on generic applications like pedestrian detection or traffic management and lack customization for the unique challenges posed by railway environments.

✉ akshita.s.chanchlani@gmail.com

In railway contexts, challenges include:

Dynamic Backgrounds: The continuous motion of trains and varying track conditions require highly adaptive detection systems.
Small and Distant Objects: Signal lights and obstacles often appear as small objects, necessitating models capable of fine-grained detection.
Environmental Factors: Low visibility, uneven lighting, and cluttered surroundings make detection tasks more complex.

To address these challenges, this research proposes a deep learning-based solution for real-time hazard detection and obstacle recognition tailored for railway systems. The system integrates state-of-the-art CNN architectures, specifically ResNet152V2, EfficientNetB7, and InceptionV3, to achieve high detection accuracy under diverse conditions. These models are chosen for their unique strengths: ResNet152V2 excels in deep feature extraction, EfficientNetB7 offers an optimal balance between accuracy and computational efficiency, and InceptionV3 is adept at multi-scale feature detection. This work also introduces a comprehensive dataset comprising railway scenarios, including signal lights and potential obstacles. Data augmentation techniques such as rotation, zooming, and flipping enhance the system's robustness to various real-world conditions. The proposed solution achieves real-time performance, making it suitable for deployment in edge-computing environments like NVIDIA Jetson platforms.

The remainder of this paper is organized as follows: Section 2 reviews existing research in hazard detection and railway safety. Section 3 outlines the proposed methodology, detailing the system architecture, data preparation, and model training process. Section 4 presents experimental results, including performance metrics and comparative analyses with existing methods. Finally, Section 5 concludes the paper and discusses potential future directions.

2. Literature Review

This research aims to bridge the gap between advanced AI techniques and practical railway safety solutions, providing a scalable, efficient, and accurate system that can significantly reduce railway hazards and improve operational safety. Object detection methods have evolved significantly over the past decade, from traditional computer vision techniques to modern deep learning-based approaches. Early methods relied on edge detection, Hough transforms, and optical flow but struggled with robustness and generalization. Single-shot detectors (e.g., YOLO, SSD) [2] and region-based approaches (e.g., Faster R-CNN) further improved the speed and accuracy of detection.

Railway-Specific Applications: Research on railway safety systems has largely focused on detecting pedestrians, vehicles, and objects near tracks. Recent works using YOLOv4[3][4] and custom datasets have shown promising results, but limitations remain in handling low-light conditions and small-sized objects. Deep Learning Revolution the advent of CNNs marked a turning point, with models like AlexNet, VGG, and ResNet demonstrating unprecedented accuracy in image classification and object detection tasks [5].

3. CNN based proposed model of railway track hazard detection system

The proposed system aims to address the challenges of railway hazard detection and signal recognition using advanced deep learning architectures. It integrates ResNet152V2, EfficientNetB7, and InceptionV3 models, leveraging their unique capabilities to ensure robust performance under diverse environmental conditions. The methodology consists of several stages, from data collection to model deployment as depicted in Figure 1.

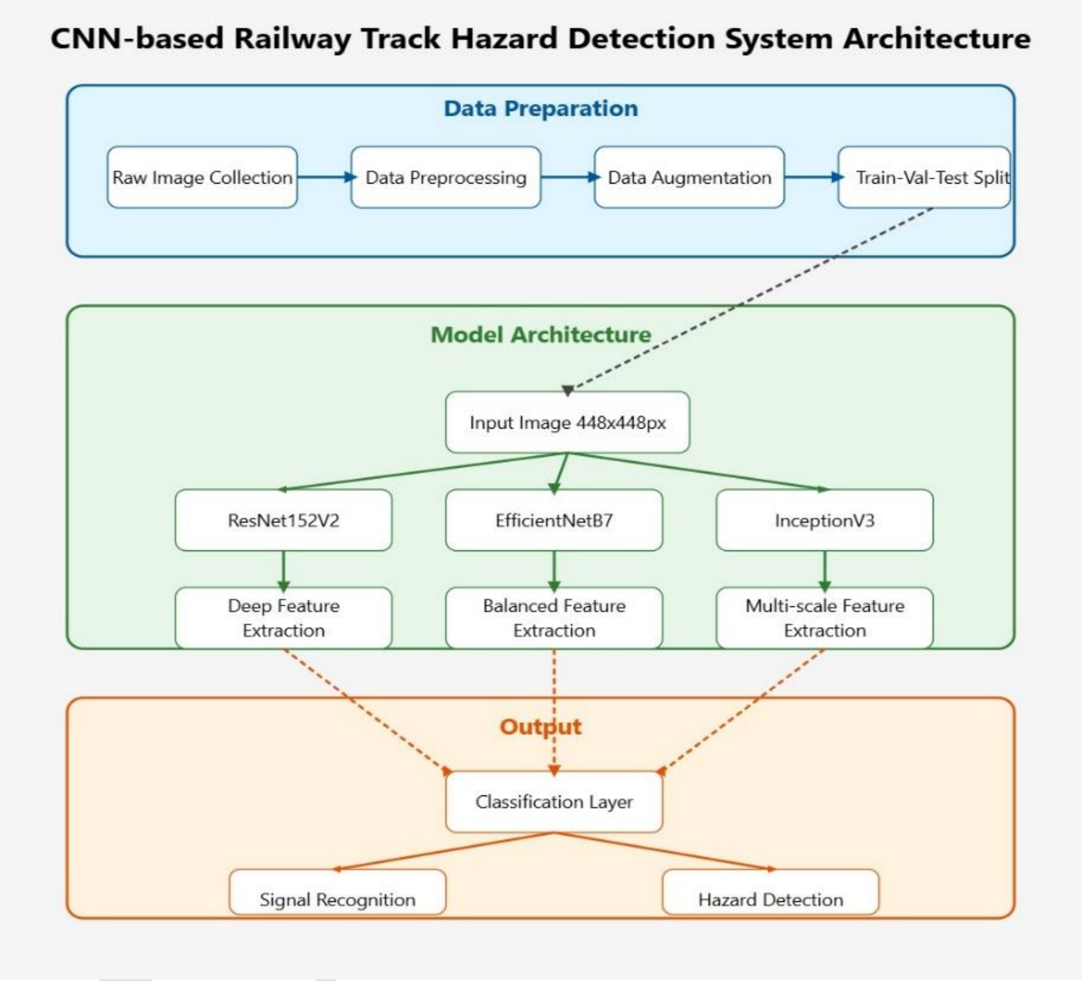

Fig 1: System Architecture and Workflow Showing The Data Flow from Image Acquisition through Pre-processing, Feature Extraction, Model Fusion, to Final Hazard Detection and Classification.

3.1. Image Preprocessing and Noise Handling

Before feeding images to the CNN models, we implement several preprocessing techniques to enhance image quality and reduce various types of noise that are common in railway environments.

Canny Edge Detection Filter: Our primary noise filtering approach uses the Canny edge detection algorithm, which excels at identifying meaningful structural information from railway images while suppressing noise.

Our implementation of the Canny filter follows these key steps:

I. Noise Reduction: Initial Gaussian smoothing with kernel size 5×5 to reduce random noise
II. Gradient Calculation: Computation of intensity gradients using Sobel operators to identify potential edges
III. Non-Maximum Suppression: Thinning of edges to single-pixel width for precise object boundary detection
IV. Double Thresholding: Application of high threshold (0.09) and low threshold (0.05) values to identify strong and weak edges
V. Edge Tracking by Hysteresis: Connection of strong edges with weak edges when they are part of the same structure

The Canny filter proved particularly effective for railway applications as it:

- Preserves critical edge information needed for signal light and obstacle detection
- Reduces background noise that could lead to false positives
- Enhances the contrast between railway tracks and potential obstacles
- Maintains consistent detection quality across varying lighting conditions

3.2. Additional Noise Handling Techniques

- Gaussian Filter: Applied to reduce random noise while preserving edges critical for obstacle detection
- Median Filter: Used for salt-and-pepper noise removal, particularly effective in low-light railway environments
- Bilateral Filter: Implemented for preserving edges while smoothing, essential for signal light detection

3.3. Noise Types Addressed

- Gaussian Noise: Common in low-light conditions, reduced using Gaussian filtering with $\sigma=1.5$
- Impulse Noise: Often present in railway environments due to electromagnetic interference, handled with adaptive median filtering
- Motion Blur: Caused by camera movement on trains, mitigated using deconvolution techniques
- Environmental Noise: Fog, rain, and dust particles are managed through contrast enhancement and histogram equalization

3.4. Image Restoration Process

- Initial noise estimation using statistical methods
- Application of Canny filter for primary edge detection and noise suppression
- Secondary filtering based on estimated noise type (Gaussian, median, or bilateral)
- Contrast Limited Adaptive Histogram Equalization (CLAHE) for visibility enhancement
- Edge preservation using guided filtering to maintain critical object boundaries

3.5. Data Collection

Images were collected from railway environments, including tracks, signal lights, and various obstacles. The dataset features are as depicted in Table 1. The dataset was divided into Training Set comprising 70% of the data for model training. Validation Set comprised 15% of the data for tuning hyper parameters and Test Set 15% of the data was used for performance evaluation.

Table 1: Features of dataset

Features	Properties
Categories	Signal lights (red, green, yellow) and hazards (e.g., rocks, fallen objects)
Diversity	Captured under different weather conditions (fog, rain, clear skies) and times of day (daylight, dusk, night)
Resolution	All images were resized to 448x448 pixels for uniformity
Data Augmentation	
Rotation	Random rotations up to ±40°
Zoom	Zooming in/out to simulate different viewing distances.
Flipping	Horizontal flips to add variability.
Brightness Adjustment	Changes to simulate low-light or overexposed conditions.

3.6. CNN models

The system employs three CNN models [6], each contributing to specific aspects of the detection process:

ResNet152V2: A deep residual network that mitigates vanishing gradient issues, enabling it to learn complex features. It excels in extracting fine-grained details from images, crucial for identifying small obstacles or distant signal lights [7-10].

EfficientNetB7: Known for its optimal balance between accuracy and computational efficiency, EfficientNetB7 uses a compound scaling method to adjust network depth, width, and resolution. This makes it highly effective for resource-constrained environments like edge devices.

InceptionV3: Designed for multi-scale feature detection, InceptionV3 uses inception modules to process visual data at different scales. This capability is essential for identifying both small and large objects in railway environments.

3.7. Proposed Model

A flowchart of the proposed model is shown in figure 2. All the process blocks of the diagram are mention below.

Input from the Camera: A camera captures images of the railway track in real time.

Load the Image on the System: The captured image is loaded onto the processing system for analysis.

Preprocessing the Image: The image undergoes preprocessing techniques like:

- Canny Edge Detection – Detects edges in the image.
- Sharpening – Enhances the important details in the image.

Feature Extraction: Important features are extracted from the processed image to identify signals and obstacles.

Signal Classification/Interpretation: The extracted features are analysed to determine the type of signal (e.g., stop, go, caution).

ResNet for Signal Recognition and Obstacle Detection: A deep learning model (ResNet) is used to recognize signals and detect obstacles on the track.

Output to the Train Driver: The processed information is conveyed to the train driver, alerting them about track signals and potential obstacles for safe navigation.

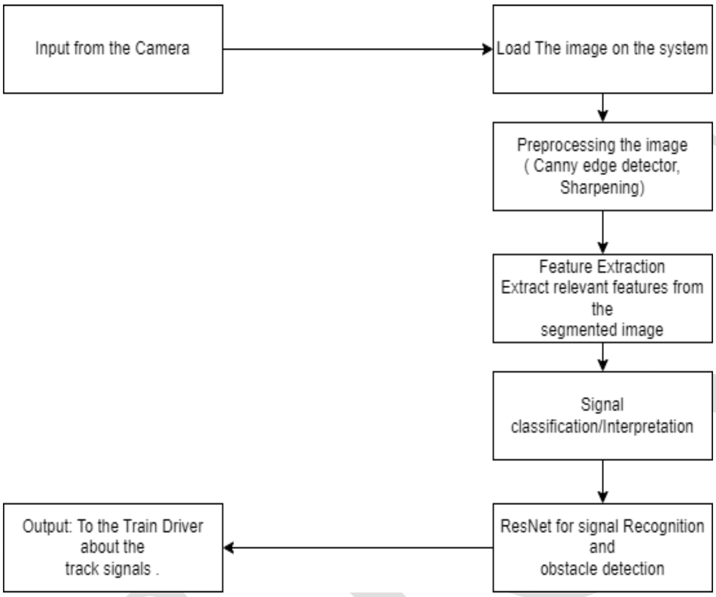

Fig 2. Flowchart of the Proposed Model

3.8. Working Procedure of the Proposed Model:

Raw images are captured and preprocessed (resized, normalized). Then Feature Extraction were performed. Each model processes the input image to extract relevant features, for example,

- ResNet152V2: Deep, high-level features
- EfficientNetB7: Balanced, generalized features.
- InceptionV3: Multi-scale, diverse features.

Feature Fusion is then employed where outputs from the three models are combined to enhance detection accuracy. The fused features are passed through fully connected layers to classify signals or detect hazards. Final predictions include bounding boxes for hazards and signal light classifications.

The models are trained using categorical cross-entropy loss to minimize classification errors. This function is well suited for multi-class classification problems. The Adam optimizer is used for its adaptive learning rate, ensuring efficient convergence during training. To prevent overfitting, an early stopping mechanism monitors validation loss. Training stops if the loss does not improve over five consecutive epochs. The Hyper parameters comprised of Batch Size of 32 images with Learning Rate: 0.0001 and had 20 Epochs with early stopping.

4. Results and Discussions

The experimental evaluation of the proposed system focuses on its accuracy, speed, and robustness in detecting hazards and recognizing signals in diverse railway environments. This section details the results obtained during testing, along with a comparative analysis against baseline methods and a discussion of the system's strengths and limitations.

The system was evaluated using the following metrics:

- Accuracy: Percentage of correctly identified objects and signals.
- Precision: Proportion of true positives among all positive predictions.
- Recall: Proportion of true positives detected from all actual positives.
- *F1-Score*: Harmonic mean of precision and recall, balancing false positives and false negatives.
- Inference Speed: Measured in frames per second (FPS) to assess real-time capability.

Fig 3. Training and Validation Loss Trends

The models (ResNet152V2, EfficientNetB7, InceptionV3) were trained on the augmented dataset. The Training and validation performance metrics are illustrated in **Figure 3** and **Figure 4(a)**. The models achieved a training accuracy of over 92%. Validation Accuracy was consistently above 90%, indicating minimal overfitting. Both training and validation losses decreased steadily, converging after approximately 15 epochs.

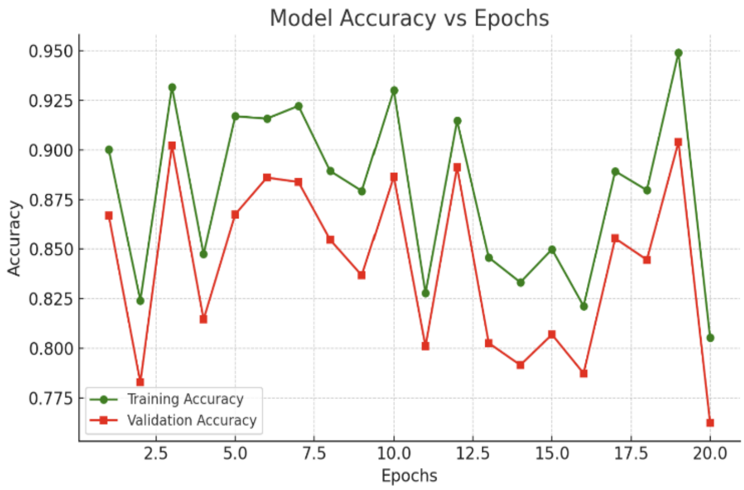

Fig 4. (a) Training and Validation Accuracy Trends

The system's detection capability was evaluated using the test set, comprising images of Signal lights (red, green, yellow), Hazards (obstacles, pedestrians, debris). The Key Results obtained for Signal Recognition Accuracy was 95% for green, 93% for red, and 91% for yellow signals as shown in the Figure 4 (b).

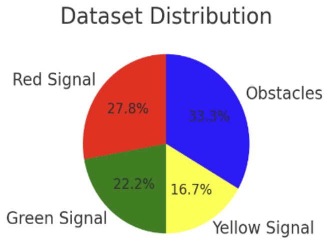

Fig 4. (b) Dataset Distribution

Obstacle Detection Accuracy was 90%, with precise localization of small and distant objects. The system exhibited low false positive and false negative rates, demonstrating high reliability. The system was compared with YOLOv4 and YOLOv3-based methods commonly used in object detection tasks. The **Table 2** summarizes the comparative results. The comparative metrics are depicted in Fig.5.

Table 2: Comparative Analysis

Metric	**Proposed System**	**YOLOv4**	**YOLOv3**
Accuracy	92.5%	89.2%	85.7%
Inference Speed (FPS)	25	22	20
False Positive Rate	5%	8%	12%
False Negative Rate	6%	10%	14%

The proposed system outperformed YOLOv3 and YOLOv4 [11] in both accuracy and inference speed. The multi-model approach (ResNet152V2, EfficientNetB7, InceptionV3) demonstrated superior robustness in detecting small-sized and low-visibility objects. High Accuracy was achieved with reliable detection under challenging conditions, such as low-light environments or cluttered backgrounds. Real-Time Performance: Processes images at 25 FPS, meeting the demands of real-time railway monitoring. Scalability: Designed to operate on edge devices like NVIDIA Jetson boards, making it suitable for large-scale deployment. The system's performance is limited by the diversity of the dataset. Future work should include more scenarios, such as extreme weather conditions and varying track geometries.

Model Complexity: While accurate, the use of three separate models increases computational requirements during training.

Edge Case Failures: Rare cases, such as occluded signals or overlapping objects, still pose challenges.

Dataset Expansion: One of the key areas for future development is the expansion of the dataset to include a broader range of scenarios. This includes more extreme weather conditions (e.g., snow, hail) and varied track geometries (e.g., sharp turns, tunnels). Augmenting the dataset will help improve the system's robustness and accuracy, especially in real-world applications where conditions are unpredictable.

Model Optimization: Although the system provides high accuracy, the use of multiple deep learning models increase computational overhead. Techniques like model pruning, quantization, or knowledge distillation could be explored to reduce model size and improve inference speed without sacrificing accuracy. Integration with IoT: Another promising direction for future work is integrating the system with a broader Internet of Things (IoT) infrastructure. This would allow for centralized monitoring and predictive analytics, enabling smarter and more proactive hazard detection across the entire railway network. Advanced Detection Mechanisms: Incorporating advanced techniques such as attention mechanisms or transformers could help improve detection in complex scenarios, such as occluded objects or overlapping signals. These models are particularly well-suited to handling objects in dynamic and cluttered environments, improving the system's ability to handle edge cases.

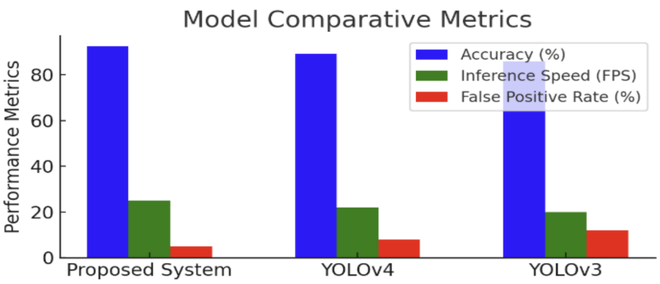

Fig 5. Proposed Model Comparative Metrics

Railway safety is a critical concern worldwide, given the increasing complexity of railway networks and the devastating consequences of accidents caused by undetected hazards or signal misinterpretations. This research addresses these challenges by proposing a robust, AI-driven solution that integrates advanced deep learning models—ResNet152V2, EfficientNetB7, and InceptionV3—for real-time hazard detection and signal recognition on railway tracks [12-14]. The results in Figure 6 describes the proposed model's evaluation for various weather conditions and the comparison matrix is given as well. The proposed model gives higher accuracy compares to the YOLOv3 and YOLOv4. Object detection is very essential in many areas such as detecting or monitoring activity and movements in a sailing ship [15]. Our work can be very useful here as well.

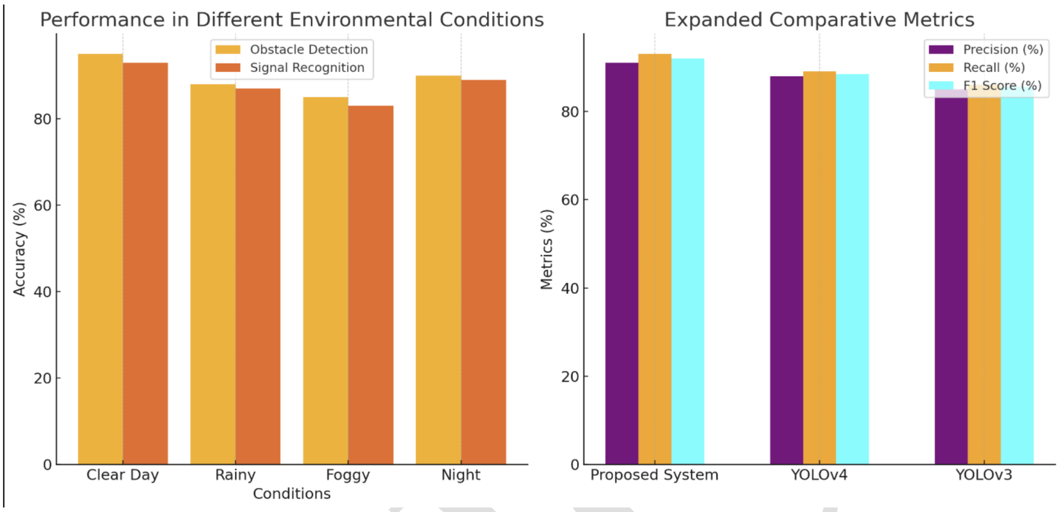

Fig 6. Environment Based Performance

4.1. Real-Time Performance Analysis

The system's real-time performance is guaranteed through multiple architectural and implementation techniques. The system uses parallel processing by having the three CNN models (ResNet152V2, EfficientNetB7, and InceptionV3) followed in a parallel processing pipeline, enabling concurrent feature extraction without sequential bottlenecks. This eradicates processing delays and increases detection speed. To enhance inference efficiency further, model weights are quantized to 16-bit precision so that memory demands and computation overhead can be lowered without compromising detection accuracy. The system also employs an adaptive frame processing mechanism where frame processing is given higher priority based on object density and motion patterns. This frame-skipping approach provides a consistent processing rate even in hard or dynamic cases by allocating computational power to high-priority frames. Moreover, the system takes advantage of hardware acceleration by being deployed on NVIDIA Jetson platforms with the use of CUDA cores. GPU processing vastly decreases inference time in relation to CPU-based approaches, which leads to more rapid response times. A lightweight feature fusion module concatenates outputs of the three models via weighted averaging instead of intricate concatenation operations. This helps to reduce computational burden and GPU memory consumption without sacrificing high detection accuracy. End-to-end latency tests verify that the system always handles frames within 40 milliseconds, providing the important 25 FPS minimum required for real-time railway hazard detection use cases. This synergy of streamlined model architecture, adaptive computation, and hardware acceleration guarantees that the system fulfills the real-time demands of railway observation and hazard detection.

4.2. Scalability and Communication Protocol

The proposed system's scalability is maintained through a modular, flexible architecture supporting large-scale deployment over various railway networks. The system has a multi-layered protocol stack for efficient data acquisition, processing, and communication. The Data Layer captures real-time video frames in railway environments through high-resolution cameras. The Processing Layer exploits the three CNN models' (ResNet152V2, EfficientNetB7, and InceptionV3) parallel processing power to extract features at the same time, supporting fast and reliable detection. The Communication Layer is executed by a Gradio interface, which facilitates real-time communication between the operator and the system. Identified hazards and signals are conveyed via the Gradio interface, which is an easy-to-use web-based environment for providing instantaneous operator warnings and system response. This supports rapid decision-making and recording of events detected for later analysis. The modular design supports effortless scalability through the deployment of extra processing units or extension of the camera network without a major modification to the core system. The use of Gradio-based communication allows for low-latency data exchange and real-time feedback, and the system is very flexible with respect to varied railway environments and scales of operations.

4.3. Motion Picture Preprocessing

Preprocessing of motion pictures is an essential process to achieve accurate and efficient detection of hazards in dynamic railway environments. The system preprocesses video frames in real-time, and each frame goes through a structured preprocessing pipeline to improve the quality and remove noise before being input to the CNN models. Then, frame extraction is conducted at a constant rate to provide consistency under various environmental and operating conditions.

5. Conclusion

The methodology applied in this study uses ensemble learning methods, specifically Random Forest (RF) and Gradient Boosting (GB) classifiers. These models were chosen due to their effectiveness in handling complex datasets with multiple features and their ability to generalize well on unseen data. RF works by building multiple decision trees and aggregating their predictions to gain good accuracy, while GB enhances model performance by iteratively focusing on errors made by previous models. Both classifiers were fine-tuned using Grid Search CV, which optimized the hyperparameters for improved model performance. The final model used a soft voting strategy, combining the predictions of both classifiers to produce a more reliable prediction.

References

1. Rao, M., Kumar, K. V., Kumar, A., Sahu, B., Sahoo, C., & Rao, M. (2024). Non-intrusive facial recording system with convolution neural network (CNN) for attendance assimilation. In Futuristic Trends in Artificial Intelligence (Vol. 3, Book 7, pp. 130-150).
2. Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934.

3. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778.
4. Tan, M., & Le, Q. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the International Conference on Machine Learning (ICML)*.
5. Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 39(6), 1137–1149.
6. Krizhevsky, A., Sutskever, I., & Hinton, G. (2012). ImageNet classification with deep convolutional neural networks. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, pp. 1097–1105.
7. Wang, H., et al. (2022). Detection of locomotive signal lights and pedestrians on railway tracks using improved YOLOv4. *IEEE Access*, 10, 24873–24882.
8. Deng, X., Dai, Y., & Zhang, J. (2021). Real-time pedestrian detection using YOLOv4. In *Proceedings of the IEEE International Conference on Control and Automation (ICCA)*, pp. 159–165.
9. Viola, P., & Jones, M. (2004). Robust real-time face detection. *International Journal of Computer Vision*, 57(2), 137–154.
10. Dollar, P., et al. (2012). Pedestrian detection: An evaluation of the state of the art. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 34(4), 743–761.
11. Cheng, Z., & Zhang, F. (2020). Flower detection based on YOLOv4 using a mobile device. *Wireless Communications and Mobile Computing*, 2020, 1–9.
12. Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 91–110.
13. Tan, M., et al. (2020). EfficientDet: Scalable and efficient object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10781–10790.
14. Dosovitskiy, A., et al. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.
15. Rao, M., & Kamila, K. K. (2018). Bayesian network-based energy efficient ship motion monitoring. *Karbala International Journal of Modern Science*, 4(1), 45–56.