

# A Survey of AI-Based Railway Track Object Detection and Alert Systems

Dr. Sincy V Thambi, Aysha Nidha, Sneha P, Souparnika M P, Twinkle P Toms  
Department of Artificial Intelligence and Data Science  
Royal College of Engineering and Technology, Akkikkavu, Thrissur, Kerala, India

**Abstract**—Railway safety is affected by the intrusion of people, animals, vehicles, debris, and other foreign objects into railway track areas. Manual monitoring can be delayed and can produce human errors, while automated vision systems face challenges related to track-region localization, small or irregular objects, changing environmental conditions, false alarms, and computational cost. This survey reviews six studies addressing railway foreign-object detection and track monitoring. The reviewed methods include improved U-Net and YOLOv5s, the TOSS camera–LiDAR system, an improved LeaderRank-based monitoring-point selection and hybrid deep-learning pipeline, MSL-YOLO, MACE-Net, and Mask R-CNN-based track segmentation. Reported contributions, outcomes, and limitations are compared with attention to differences in datasets and experimental settings. The survey identifies recurring needs for diverse real-world datasets, reliable danger-zone localization, lightweight inference, consistent benchmarking, and verified end-to-end alert delivery.

**Index Terms**—Railway safety, foreign-object intrusion, object detection, deep learning, YOLO, Mask R-CNN, track segmentation, alert generation.

## I. INTRODUCTION

Railways are a critical transportation system, but objects entering the track area can cause collisions, delays, and risks to people and wildlife. Manual observation depends on continuous attention and timely response. Computer vision and deep learning offer a way to automate parts of this monitoring process, but a detector must identify not only an object but also whether it lies within the active track or danger region.

This survey examines six supplied research papers related to railway foreign-object detection and alerting. The studies address different parts of the problem: track-boundary segmentation, object detection, camera and LiDAR data fusion, monitoring-point selection, temporal video analysis, and track-region masking. The review compares their approaches and reported results, then derives implementation gaps relevant to an AI-based railway track object detection and alert system.

Across these studies, the central technical challenge is not simply to detect an object in a frame, but to determine whether it presents a hazard to railway operation. The studies therefore represent complementary design choices rather than interchangeable solutions. Their reported scores should be interpreted with the original datasets, evaluation protocols, and hardware in mind.

## A. Survey Method and Selection Scope

The synthesis also considers the full operational chain. A detector may identify an object correctly yet still produce an unsafe or unhelpful system if it cannot determine whether the object intersects the track, if the alert is delayed, or if the same harmless object repeatedly triggers warnings. Consequently, model accuracy is treated as one part of system performance rather than the only criterion for railway deployment.

The survey follows a structured, paper-by-paper reading process. For each supplied article, the problem addressed, main algorithmic components, input data, reported outcome, and stated or observable limitations were extracted. The extracted details were then grouped into common themes: track segmentation, object detection, multi-sensor perception, foreground/trajectory analysis, temporal modelling, and alerts. The comparison table summarizes these themes and preserves each paper's own reported metrics; it does not recompute or normalize results across unrelated datasets.

The scope is intentionally limited to the six supplied papers as the core studies. To make the reference list and background discussion consistent with a full survey-paper format, additional relevant works cited inside those six papers are used only as supporting context; they are not treated as additional core papers reviewed in the comparison table. This distinction preserves the requested six-paper scope while acknowledging closely related railway intrusion-detection, risk-assessment, and general object-detection research [7]–[16].

The intended contribution is not to claim a new detector or a universal performance winner. Instead, this survey organizes the literature around the complete safety chain—track localization, object recognition, danger-zone/risk reasoning, alert delivery, and operational validation—and highlights which system-level measures are still inconsistently reported. This scope is narrower than general rail-construction asset detection and broader than detector-only accuracy comparison.

The contribution of this survey is a system-level comparison of six core studies across five connected functions: track-region localization, object recognition, risk or intrusion assessment, alert generation, and deployment evaluation. This perspective is intended to distinguish detector-level performance from the reliability of the complete warning pipeline. It also identifies reporting items that are often missing or difficult to compare across studies, including test-set diversity, hardware-specific latency, false alerts per unit time, and the delay between intrusion and warning.

The review is organized around methods and system functions rather than treating each paper as an unrelated summary. The selected studies are compared in terms of input modality, track-region understanding, object recognition, temporal reasoning, alert generation, reported evaluation, and practical deployment considerations. The discussion distinguishes information explicitly reported by the papers from design implications inferred by comparing them. Since the papers use different datasets and experimental settings, their numerical results are not interpreted as a common benchmark.

For consistency, this survey distinguishes results explicitly reported by an article from analytical observations made when comparing the studies. Where a paper does not provide a directly comparable metric or a complete description of its test conditions, the limitation is stated rather than filling the gap with an assumed value.

## II. REVIEW OF SELECTED STUDIES

The reviewed systems also differ in sensing. A monocular camera provides rich visual appearance but can be affected by lighting, occlusion, and ambiguous distance. LiDAR contributes geometric and distance information, but adds sensor cost, calibration requirements, and point-cloud processing. Video-based temporal methods use information across frames to reduce reliance on a single image, although they must account for processing latency and object movement. These differences influence the most suitable architecture for a given installation [1]–[6].

Before comparing individual approaches, it is useful to distinguish three related tasks. Track localization estimates the region occupied by rails or the railway corridor. Object detection identifies and classifies people, animals, vehicles, debris, or other obstacles, usually with bounding boxes. Intrusion assessment combines the two outputs to decide whether an object overlaps a defined danger region. Segmentation-based methods can represent the track boundary more precisely, while box-based detectors can be faster and simpler to deploy; neither alone guarantees a reliable alert.

### A. Improved U-Net and YOLOv5s for Railway Foreign Objects

Ning, Ding, and Chen [1] present a vision-based approach that first segments railway-track categories using an improved U-Net and constructs a track-boundary model from extracted feature points. The least-squares method fits the railway boundary. For object detection, the authors improve YOLOv5s by adding an SE attention mechanism, an extra prediction layer for small objects, adjusted anchor-box sizes, and Inner-IOU for bounding-box regression. Their improved segmentation network reports 91.8% mIoU, a 3.9% improvement over the earlier model, while the optimized detector reports a 7.4% increase in mAP@0.5 compared with the original YOLOv5s. The method directly addresses track-region identification and small-object detection. Its results are tied to the authors' dataset and should be independently validated before deployment.

The architecture is useful because the two stages answer different questions: segmentation estimates where the track lies,

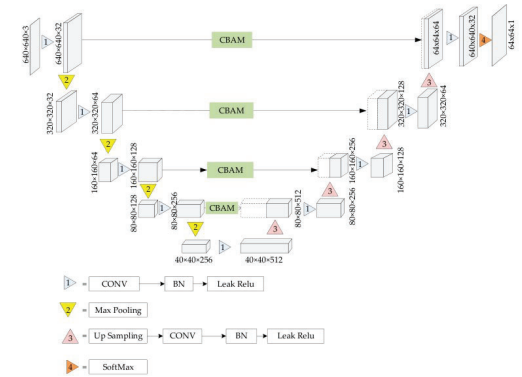


Fig. 1. Improved U-Net architecture used for railway-track segmentation [1].

while object detection identifies the potential obstruction. The boundary model and least-squares fitting provide a geometric representation that can support track-aware filtering. In a practical implementation, the quality of the final decision depends on both stages: a missed or inaccurate track boundary can exclude a real hazard, and a detector error can still generate a false warning. The reported improvement should therefore be read alongside the paper's dataset and evaluation protocol rather than as a universal performance guarantee [1].

For deployment, the segmentation and detection stages should be evaluated separately as well as together. A small boundary error may change whether an object is considered inside the danger zone, even when the object detector itself is correct. Testing should therefore include trackside objects that are close to, but not actually inside, the defined hazard region.

### B. TOSS: Camera–LiDAR Detection and Fuzzy Alerting

Rajeswari et al. [2] introduce TOSS, a track-object detection system combining a camera and LiDAR. LiDAR point-cloud data is converted into a bird's-eye view, preprocessed, clustered, and used to estimate object dimensions and positions. Camera images are denoised using a bilateral filter, and YOLOv8 detects objects in the visual stream. The camera and LiDAR outputs are supplied to a fuzzy system that generates alert messages for the locomotive pilot and nearby control rooms. The paper reports 98.91% overall accuracy and 97.1% mAP. A key strength is the explicit connection between detection and alert generation. However, the design relies on multiple sensors, calibration, data fusion, and fuzzy decision logic, increasing implementation complexity compared with a camera-only prototype.

TOSS is distinctive in making the sensing and warning stages explicit. The camera contributes semantic appearance cues, whereas LiDAR supplies three-dimensional geometry and distance-related information. Combining them can help distinguish a nearby object from an object that appears close in a two-dimensional image. However, fusion introduces engineering requirements: the sensors must be aligned, timestamps should correspond, and missing or noisy measurements need a defined fallback. The fuzzy alert layer is also important because an operational warning needs to convert model outputs into a decision that can be understood and acted upon [2].

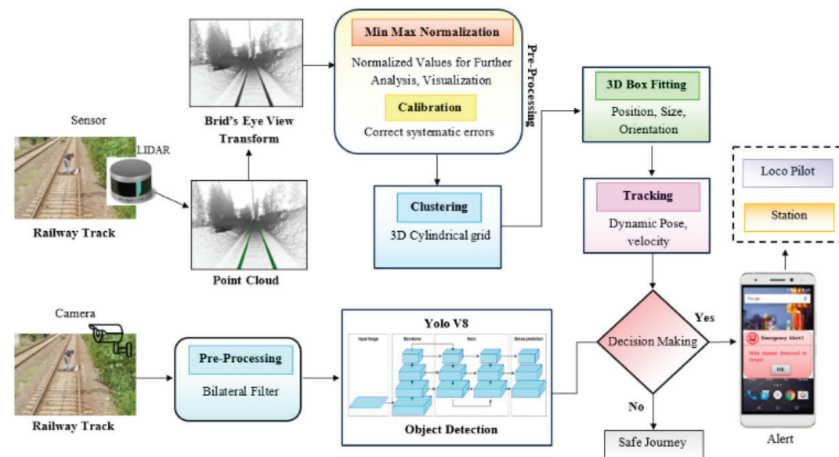


Fig. 2. Overall process of the TOSS camera-LiDAR detection and alert methodology [2].

The practical benefit of fusion depends on reliable time synchronization, coordinate calibration, and handling of missing measurements. If camera and LiDAR observations refer to different moments or coordinate systems, fusion may produce incorrect object positions. A fair assessment should report not only recognition performance but also calibration effort, processing latency, and the behaviour of the system when one sensor is unavailable.

### C. Improved LeaderRank-Based Intrusion Identification

This approach emphasizes where monitoring should be focused as well as how intrusion candidates are identified. The key-point ranking component can support placement of cameras or other monitoring equipment at strategically important locations. The foreground-segmentation and recognition stages then process the scene at those locations. Its value is therefore partly system-level: improving coverage of critical points may be useful even when the recognition pipeline is not identical to a modern end-to-end detector. Evaluation should consider whether the selected points cover the relevant track sections and whether the extracted candidates remain reliable under changing backgrounds [3].

Niu [3] proposes a pipeline for identifying foreign-object intrusion in key railway areas. An improved LeaderRank algorithm identifies key trajectory points that can guide monitoring-equipment placement. An improved Gaussian mixture model segments surveillance images to extract candidate foreground regions, after which a hybrid deep-learning model recognizes intrusion behaviour. A Firefly Algorithm is used to optimize the model. The study's main contribution is combining monitoring-point selection, foreground extraction, and deep-learning recognition. The publisher's abstract reports improved recognition accuracy and reliability but does not provide a single numerical metric in the abstract; therefore, this survey does not assign an unsupported score. The approach is promising for monitoring coverage, although practical performance depends on camera placement, segmentation quality, and validation under diverse track conditions.

The LeaderRank component addresses the allocation of monitoring attention, while foreground extraction and recognition address scene interpretation. These functions should be evaluated separately: effective camera placement does not by itself guarantee accurate object classification, and strong classification does not guarantee that all critical track locations are monitored. The overall benefit depends on both stages working together.

### D. MSL-YOLO for Railway Foreign-Object Intrusion Detection

Niu, Feng, and Hou [4] propose MSL-YOLO, a lightweight detector developed from YOLOv8n for railway foreign-object intrusion detection. The model replaces the SPPF module with a Multi-scale Shared Convolution Module (MSCM), improves the C2f feature-fusion structure using StarBlocks and Efficient Multi-scale Attention (EMA), and uses a Lightweight Shared Convolutional Detection Head (LSCD). These changes are intended to strengthen multi-scale feature extraction while reducing computational cost. The authors evaluate MSL-YOLO on a custom railway intrusion dataset containing 3,311 images and seven object categories, with data augmentation applied to the training set. The paper reports 94.3% mAP@0.5, 2.35 million parameters, 6.7 GFLOPs, 277 FPS, and 3.30 ms latency under the reported experimental setup. The authors also describe failure cases involving extreme lighting, heavy occlusion or overlap, and cluttered backgrounds. Since the dataset is custom-built and hardware and test conditions affect speed metrics, these results should be interpreted in the context of the authors' experimental setup rather than treated as a universal ranking.

The reported MSL-YOLO figures describe the authors' experimental setup and should not be treated as guaranteed performance on a different railway line or camera system. Dataset size, class balance, image resolution, GPU or CPU hardware, and the definition of FPS can all affect the result. For reproducibility, future evaluations should state these conditions and include difficult examples such as partial occlusion and small distant objects.

#### E. MACE-Net for Foreign-Object Intrusion Detection

Chen et al. [5] propose MACE-Net, a multi-level feature aggregation and context-enhancement network built around YOLOv8. The model integrates GOLD-YOLO for multi-level feature aggregation, DCNv3 for adapting feature sampling to irregular object shapes, CBAM attention for channel and spatial feature refinement, and GIoU loss for improved localization. The authors construct a dataset of 7,200 railway images covering pedestrians, obstacles, animals, and vehicles. MACE-Net reports mAP@0.5 of 83.8% and an F1-score of 83.4%, compared with 78.9% mAP@0.5 for the YOLOv8 baseline. The paper reports 43 frames per second, compared with 47 FPS for the baseline, and notes increased model size and computational complexity. It also reports false positives and false negatives in challenging backgrounds, indicating that improved accuracy does not eliminate the need for robust real-world testing.

MACE-Net focuses on the representation of objects at multiple scales and on contextual feature enhancement. Multi-level aggregation can help preserve both coarse semantic information and finer spatial details; deformable sampling can adapt to irregular shapes; and attention can emphasize informative channels or image regions. These components are relevant to railway scenes where hazards can vary greatly in size and shape. At the same time, additional modules should be justified through controlled ablation tests and inference-time measurements, because a more elaborate architecture is not automatically better for real-time embedded hardware [5].

The design illustrates how feature aggregation, deformable sampling, and attention can target different sources of detection error. Nevertheless, added modules also increase design complexity and may affect inference speed. A useful ablation study would measure the contribution of each component individually, while a field-oriented test would check whether improvements persist across lighting, weather, viewpoints, and object sizes.

#### F. Mask R-CNN for Track Identification and Object Detection

Mahmud et al. [6] use a Mask R-CNN pipeline to detect the railway track and then identify objects inside the detected track region. The system uses a ResNet-based backbone and feature-pyramid features, region proposals, ROI alignment, classification, bounding-box regression, and instance masks. Image-processing stages include RGB conversion, Gaussian filtering, HSV conversion, and blue-mask extraction to select the track as the region of interest. The authors report a mean average precision of 0.9375 and a frame rate of 30 FPS using approximately 300 images derived from GTA V train-view footage at  $512 \times 512$  resolution. The track-region mask can reduce detections of objects beside the active track. Limitations include a small dataset, computationally intensive training, and the need for more varied object categories and weather conditions.

The important design choice in this work is to identify the rail region before deciding which objects are relevant. This reduces the chance that an object beside the track is treated as an immediate obstruction simply because it is visually close to the rails. Instance masks provide a more detailed spatial

representation than rectangular boxes, but the two-stage Mask R-CNN pipeline can require more computation than a lightweight one-stage detector. The paper reports a mean average precision of 0.9375 and a frame rate of 30 frames per second under its experimental conditions; those values should not be assumed for a different camera, dataset, or processing device [6].

Instance masks can support a more precise estimate of the overlap between an object and the railway danger zone than a bounding box alone. This spatial detail comes with greater annotation and computation requirements, so the appropriate choice depends on the safety decision being made. If the application only needs a coarse warning, a lighter detector may be sufficient; if precise track overlap matters, segmentation may be valuable.

#### G. Literature Search Strategy and Study Selection

To improve traceability, the literature search for this survey was structured around railway foreign-object intrusion, track-obstacle detection, railway perception, and alert/risk-assessment systems. Publisher and scholarly records were consulted through IEEE Xplore, ScienceDirect/Elsevier, PubMed Central, MDPI, Springer Nature, PLOS, and indexed search results. Search phrases included “railway track foreign object detection,” “railway obstacle intrusion detection,” “railway track object detection deep learning,” “railway intrusion risk assessment,” “camera LiDAR railway detection,” and “railway object detection survey.” The search was supplemented by backward checking of reference lists from relevant papers and surveys.

Studies were considered relevant when they addressed railway or rail-transit obstacle/foreign-object detection, track or danger-zone localization, railway intrusion risk assessment, rail-specific sensing, or an evaluation method directly applicable to railway safety. Priority was given to peer-reviewed journal articles and established conference papers with identifiable bibliographic records. General object-detection papers were retained only where they explained a method used by the reviewed railway studies. Biomedical or otherwise unrelated applications were excluded from the core evidence set.

Screening was based on title and abstract first, followed by the available full text or publisher record where accessible. For each included study, the review recorded the detection approach, sensing modality, dataset description, reported metrics, alert/risk functionality, and stated limitations. This is a structured narrative survey rather than a registered PRISMA systematic review: a complete reproducible count of all records retrieved and excluded was not available for this manuscript, so no numerical screening flow or claim of exhaustive coverage is made. A submission-ready systematic review should add dated database exports, duplicate-removal counts, full-text exclusion reasons, and a PRISMA flow diagram if the authors complete that process.

To reduce overinterpretation, numerical results are quoted only when visible in the cited source and are not used to rank models across unrelated test sets. Missing hardware details, unavailable datasets, and unreported alert-level measures are treated as reporting limitations rather than inferred values.

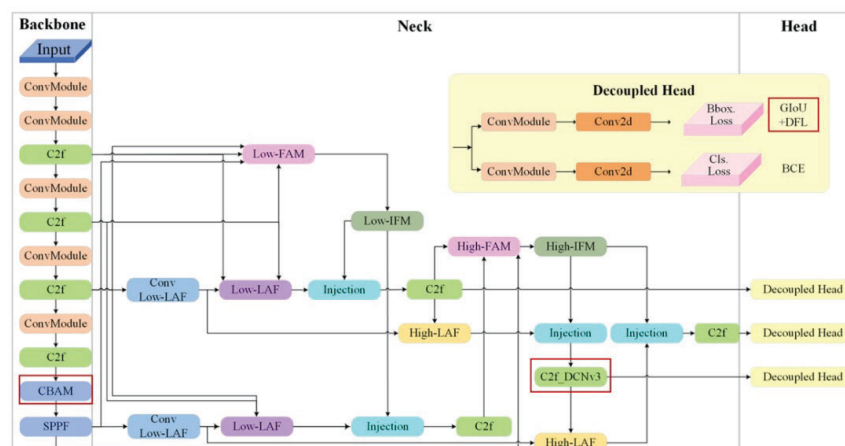


Fig. 3. Multi-level feature aggregation and context-enhancement network (MACE-Net) [5].

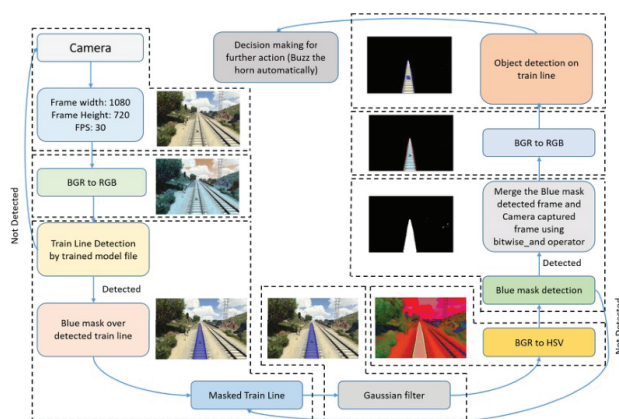


Fig. 4. Block diagram of the proposed Mask R-CNN-based track detection system [6].

Recent related work reinforces the need to compare systems at both detector and operational levels. A 2024 study in Heliyon converts urban rail-transit foreign-object detection into a foreground-segmentation problem and reports 95.8% detection accuracy on its collected subway-platform data [30]. A 2025 MSL-YOLO study focuses on railway foreign-object intrusion detection [4], while a 2025 Expert Systems with Applications paper combines MSIA-YOLOv8 with DALNet for obstacle intrusion detection and risk assessment and reports 97.3% mAP and 137 FPS under its own experimental conditions [31]. These headline results are not directly comparable because datasets, class definitions, image sizes, and hardware differ. In addition, a 2025 systematic review/meta-analysis addresses small-object detection in rail construction, including dataset and model-complexity considerations [32]; its focus on rail assets such as clips, bolts, and nuts differs from safety-critical foreign-object intrusion, so it is relevant background rather than a direct substitute for railway hazard-detection evidence.

### III. COMPARATIVE ANALYSIS

#### A. Supporting Literature and Technical Context

Several closely related studies provide context for the six core papers. Ning et al. [7] combine railway foreign-object detection with risk assessment, emphasizing that a detection result must be interpreted in relation to the danger posed by an object. Wang et al. [8] investigate a deep generative, semi-supervised approach to rail foreign-object detection, a direction relevant when annotated railway images are limited. Zhang et al. [9] address real-time foreign-object detection in complex open railway environments, where changing backgrounds and scene variability make robust recognition difficult.

Other railway-specific studies explore changes to detector architectures and intrusion-warning logic. Meng et al. [10] describe SDRC-YOLO for railway foreign-object intrusion scenarios. Zhang et al. [11] connect YOLO-based detection with risk assessment in an obstacle-intrusion warning mechanism. Wang et al. [12] review deep-learning-based railway perimeter intrusion detection, while Song et al. [13] investigate attention-aggregated semantic segmentation for railway foreign-object perception. Cao et al. [14] survey machine-vision railway intrusion detection and discuss associated challenges and perspectives. These supporting works indicate that detection, localization, risk interpretation, and alerting are related but distinct system functions. Railway operation and surveillance studies also discuss autonomous-train deployment, operational obstacle detection, railway monitoring frameworks, and broader rail-network video analytics [17], [18], [25], [27], [28].

The wider computer-vision literature also clarifies the roles of the architectures used in the reviewed railway studies. Mask R-CNN combines object detection with instance-level masks [15], which makes it suitable when the railway region must be represented spatially rather than only by a bounding box. SSD is a single-shot object detector [16] and is often discussed as a speed-oriented alternative to proposal-based detectors. These general methods are cited here to explain the technical background; their results should not be interpreted as railway-specific

performance unless evaluated on railway data.

The supporting literature should not be ranked directly against the six core papers unless datasets, object classes, hardware, and evaluation protocols are aligned. Its role in this survey is to show the broader design space: semi-supervised learning can address annotation constraints; attention and feature aggregation can improve feature representation; risk assessment can prioritize hazards; and segmentation can help determine whether an object lies within the track danger region. A complete railway warning system should combine these functions only when the available data and implementation resources justify the added complexity.

#### B. Comparison by Functional Capability

No single approach is best under every constraint. A camera-only model may be suitable when cost and ease of installation dominate; sensor fusion may be justified where reliable distance information is important; a segmentation-aware design is attractive when the system must distinguish the track from its surroundings; and temporal modelling may be valuable when sequential evidence improves confidence. The appropriate selection depends on available hardware, required response time, environmental conditions, the types of hazards to detect, and the cost of false alarms versus missed detections.

The papers contribute at different points in the detection chain. The improved U-Net/YOLO approach [1] and Mask R-CNN system [6] explicitly combine track-region understanding with object recognition. TOSS [2] emphasizes complementary camera and LiDAR sensing and includes fuzzy alerting. The LeaderRank work [3] addresses key-point selection and intrusion processing, making monitoring coverage part of the design. MSL-YOLO [4] focuses on lightweight multi-scale detection and inference speed, while MACE-Net [5] concentrates on feature aggregation and context enhancement.

A meaningful comparison should consider the full pipeline: input sensing, track-region localization, object recognition, intrusion reasoning, and alert delivery. A method that performs well at one stage may still be unsuitable if it omits another stage required for an operational warning. The table is therefore a structured summary of reported approaches, not a universal ranking of the models.

Table I summarizes the six papers using methods and results stated in the supplied sources. Metrics differ in definition, dataset composition, and experimental conditions; values are therefore descriptive rather than a fair ranking across papers.

When interpreting Table I, differences in dataset composition and metric definitions must be kept in view. Accuracy, mean average precision, intersection-over-union, and frame rate measure different properties; a higher value in one column does not establish overall superiority. The table should be read alongside the qualitative limitations and the original papers' experimental conditions.

#### IV. COMMON LIMITATIONS AND RESEARCH GAPS

The reviewed literature demonstrates progress in object detection, track segmentation, and alert generation, but several recurring

gaps remain.

- Dataset diversity and generalization: some studies use custom or small datasets, synthetic imagery, or limited test conditions. More varied real-world data is needed across day/night scenes, weather, track curvature, object distance, and camera viewpoints.
- Track-aware decision-making: object detection alone does not always determine whether an object is inside the active danger zone. Track-boundary segmentation or polygon masks can reduce unnecessary alerts for nearby objects.
- False alarms and missed detections: complex backgrounds, vegetation, distant small objects, occlusion, and poor illumination remain challenging. Railway safety requires careful evaluation of both false positives and false negatives.
- Computational efficiency: higher-performing architectures may require more parameters and computation. A practical system should measure latency and frame rate on the intended hardware, not only on a research workstation.
- End-to-end alert verification: TOSS explicitly describes a fuzzy alert mechanism, while other studies mainly emphasize detection. An implementation should test the complete path from detection through danger-zone assessment to a timely, understandable alert.
- Comparable evaluation: the studies use different datasets and metrics, making headline values such as accuracy, mAP, and FPS unsuitable for direct ranking. A common test set and consistent metrics would make comparisons more meaningful.

Operational robustness and alert management: a field system must define how detections from consecutive frames are combined, how confidence thresholds are selected, when an alert is repeated or cleared, and what happens when the camera is obstructed or the model stops responding. These details affect usability and reliability even though they may not be captured by object-detection metrics. A fail-safe response and event logging should be designed at system level and tested under realistic operating conditions.

Class imbalance and rare hazards: dangerous intrusions may be much less frequent than normal track scenes. A model can therefore achieve a seemingly strong aggregate score while failing on uncommon but safety-critical cases. Evaluation should report per-class precision and recall, confusion matrices, and results for rare object categories where the dataset supports them. Synthetic or augmented samples can help training, but final testing should include independently collected real scenes.

These gaps are connected. Limited data diversity can increase false alarms under new conditions, while rare hazardous events make recall difficult to estimate reliably. Track-aware reasoning can reduce irrelevant detections, but it also depends on accurate track localization. Research should therefore report component-level performance and end-to-end alert outcomes rather than optimizing a single detector score in isolation.

TABLE I  
COMPARISON OF THE SIX REVIEWED STUDIES

| Paper                       | Core method                                      | Input / dataset                                           | Reported result                                                       | Main limitation                                                                 |
|-----------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------|---------------------------------------------------------------------------------|
| [1] Ning et al. (2024)      | Improved U-Net + improved YOLOv5s                | Railway image dataset                                     | mIoU 91.8%; +7.4% mAP@0.5 vs baseline YOLOv5s                         | Needs validation beyond the authors' dataset                                    |
| [2] Rajeswari et al. (2024) | YOLOv8 + camera/LiDAR + fuzzy logic              | Camera and LiDAR data                                     | 98.91% accuracy; 97.1% mAP                                            | Sensor calibration and fusion complexity                                        |
| [3] Niu (2025)              | Improved LeaderRank + GMM + hybrid deep learning | Track-monitoring video/images                             | Reports improved recognition; no single headline score in abstract    | Performance depends on monitoring placement and segmentation                    |
| [4] Niu, Feng & Hou (2025)  | MSL-YOLO: YOLOv8n + MSCM + C2f-StarEMA + LSCD    | Custom railway intrusion dataset: 3,311 images; 7 classes | 94.3% mAP@0.5; 2.35M parameters; 6.7 GFLOPs; 277 FPS; 3.30 ms latency | Custom dataset; failures under extreme lighting, occlusion/overlap, and clutter |
| [5] Chen et al. (2025)      | MACE-Net: GOLD-YOLO, DConv3, CBAM, GIoU          | 7,200 railway images                                      | mAP@0.5 83.8%; F1 83.4%; 43 FPS                                       | Higher computation than baseline YOLOv8                                         |
| [6] Mahmud et al. (2023)    | Mask R-CNN + track-region mask                   | ≈300 GTA V-derived images, 512 × 512                      | mAP 93.75%; 30 FPS                                                    | Small dataset; intensive training; more weather diversity needed                |

*Note:* Results are transcribed from the cited papers and are not directly comparable because datasets, object classes, hardware, and evaluation protocols differ. Reported speed and accuracy values should be interpreted only within each study's experimental conditions.

## V. SUGGESTED DIRECTION FOR THE PROPOSED SYSTEM

Based on the reviewed studies, a feasible prototype can use a camera-based input pipeline with image resizing and normalization, track-region localization, object detection, danger-zone checking, and alert generation. A YOLO-family detector is a reasonable starting point when inference speed and implementation simplicity are priorities. A segmentation-based track mask can be added if detections outside the active track region cause frequent false alarms. LiDAR and fuzzy fusion, as used by TOSS [2], may be considered only if the project has the hardware and time required for calibration and multimodal integration.

A practical evaluation should report precision, recall, F1-score, mAP@0.5, inference speed or latency, and the number of false alerts. Testing should include objects inside the track, objects beside the track, partially intruding objects, small and distant objects, low-light scenes, and different weather conditions. Alert delivery should be tested separately from model accuracy so that a correct detection is not assumed to guarantee successful notification.

A reproducible test protocol should separate training, validation, and test footage by recording session or location where possible, reducing the risk that near-duplicate frames appear in multiple splits. Test cases should include objects fully inside the track area, objects outside it, partial intrusions, small distant hazards, occlusion, shadows, rain or low illumination where available, and scenes with no hazard. Each reported result should state the dataset size, object categories, image or video input, hardware, confidence threshold, and evaluation metric definition.

The evaluation can be organized into complementary test groups: (i) object recognition, using precision, recall, F1-score, and mAP; (ii) track localization, using an appropriate segmentation score such as IoU or mIoU where masks are available; (iii) real-time operation, measuring end-to-end latency and sustained frame rate on the target hardware; and (iv) alert quality, recording false alerts, missed intrusions, time from first visible

intrusion to warning, and stability of alerts across consecutive frames. These groups should be reported separately so that a strong detector score does not hide a weakness in the end-to-end system.

For a prototype, the test plan should be fixed before model tuning. Training, validation, and test sets should be separated by recording session or location where possible, and the final test set should remain untouched during model selection. This makes the reported results more credible and reduces the chance that near-duplicate video frames inflate performance estimates.

## VI. FUTURE RESEARCH DIRECTION

### A. Evaluation Protocol and Reporting Requirements

A fair evaluation should separate image-level recognition from system-level safety performance. For detection, precision, recall, F1-score, and mean average precision should be reported with the confidence threshold and IoU criterion stated. For track segmentation, IoU or mean IoU can be reported where ground-truth masks are available. For real-time operation, the paper should state the hardware, input resolution, average latency, sustained frames per second, and whether preprocessing and alert delivery are included in the timing. This separation prevents a fast detector from appearing end-to-end real-time when the complete alert pipeline is slower.

Test data should represent both positive and negative situations. Positive examples should include objects fully inside the danger zone, partial intrusions, small distant hazards, and partially occluded objects. Negative examples should include objects near but outside the track, trackside vegetation, shadows, reflections, and ordinary railway infrastructure. Where the dataset permits, results should be reported by class and scene condition rather than only as one aggregate score. Rare hazards deserve separate attention because overall accuracy can hide a poor recall for uncommon but dangerous objects.

Data splitting is another important requirement. Frames ex-

tracted from the same video are highly similar, so random frame-level splitting can place near-duplicate images in training and testing sets and inflate performance. A stronger protocol separates footage by recording session, camera, route, or location when possible. The final report should describe the number of images or clips, the classes annotated, the split strategy, annotation rules, and any augmentation used. These details make results easier to reproduce and help readers judge whether a model is likely to generalize to a new railway environment.

Alert quality should be measured independently from detector quality. Useful system-level measures include false alerts per unit time, missed intrusions, time between first visible intrusion and warning, alert stability across consecutive frames, and recovery after camera or network interruption. The reviewed papers do not all report these measures, so they are proposed evaluation criteria rather than results attributed to the papers. In safety-related applications, an understandable and timely alert is a necessary part of system performance, not merely an optional interface feature.

A further direction is to make alerts actionable. A warning can include the estimated hazard class, approximate position relative to the track, confidence, and time of detection, while avoiding unnecessary repeated notifications. Logs can support later review of missed events and false alarms. Any deployment-oriented study should also define how the system behaves during camera failure, network interruption, or uncertain predictions. These are design recommendations arising from the comparative review; they are not claims that every reviewed paper has already implemented these functions.

Future work should prioritize robust field validation over architecture complexity alone. A common benchmark containing diverse locations, seasons, weather, viewpoints, and object categories would support fairer comparison. A modular pipeline that combines track segmentation, object detection, and a transparent intrusion rule may be easier to validate than an opaque single score. Where sensor fusion is used, calibration and synchronization should be measured as part of the experiment. Where temporal models are used, the gain in missed-detection reduction should be compared against added latency.

Future systems should also be designed for maintainability. Model updates, camera movement, seasonal changes, and changes in trackside infrastructure can cause performance drift after deployment. A practical monitoring plan should define when to review alerts, how to collect difficult cases safely, and how to revalidate a model before a revised version is used in an operational setting.

## VII. ADDITIONAL COMPARATIVE DISCUSSION

### A. Track Localization and Danger-Zone Reasoning

An important distinction across the reviewed literature is whether a model detects an object anywhere in an image or decides that the object occupies a railway danger zone. The distinction is central to operational usefulness. A person or animal visible beside the railway may not require an immediate train warning,

whereas a small object partly entering the active track may be safety-critical. The Mask R-CNN study addresses this issue by identifying the track region before detecting objects within it [6]. The improved U-Net and YOLOv5s method similarly combines track segmentation and object detection [1]. These designs suggest that the track boundary should be treated as a functional part of the decision pipeline rather than only as visual context.

Track localization can be represented by a polygon, pixel mask, rail-line estimate, or semantic segmentation map. Each representation has different implementation costs. A polygon or mask can model curved track regions more closely than a fixed rectangular region of interest, but it also requires reliable localization under perspective changes, shadows, vegetation, and partial occlusion. Rail detection and railway-environment segmentation studies cited in the supplied papers provide useful technical context for this problem [19], [20], [21]. Their findings should not be treated as directly equivalent to foreign-object detection results because track segmentation and obstacle recognition are different tasks.

The danger zone should be defined explicitly for the intended application rather than assumed to be identical in every scene. Its geometry may depend on camera perspective, rail spacing, and the required safety margin. A system should also define how it handles uncertain boundaries and objects that partially overlap the zone, because these cases can determine whether an alert is triggered.

### B. Sensor Selection and Multimodal Perception

Camera-only systems have a practical advantage because they can reuse video streams and provide colour, texture, and semantic information. Their weaknesses include sensitivity to illumination, weather, camera vibration, and objects that occupy only a few pixels. TOSS addresses some of these limitations by combining camera images with LiDAR-derived geometric information [2]. LiDAR can provide distance and shape cues, but its integration introduces hardware cost, calibration, synchronization, and point-cloud processing requirements. Work on high-precision LiDAR for railway obstacle detection, cited in the supplied deep-learning study, reinforces the relevance of distance-aware perception [22].

The sensor decision should therefore follow the intended operating environment. A low-cost prototype may begin with a camera and track-aware object detector. A deployment requiring robust distance estimates may justify camera-LiDAR fusion. In either case, the evaluation should report the conditions under which the system succeeds and fails. Sensor fusion should not be assumed to improve every metric automatically: its benefit depends on calibration quality, synchronization, sensor coverage, and the reliability of each input stream.

Sensor selection should be based on a documented trade-off between installation cost, reliability, environmental tolerance, and the information needed by the warning logic. A camera-only baseline is useful for establishing performance and cost, while a multimodal design should demonstrate that the additional

sensor improves safety-relevant outcomes enough to justify its calibration and maintenance burden.

### C. Detector Design: Accuracy, Speed, and Data Requirements

The supplied studies cover one-stage detectors, segmentation-aware networks, sensor-fusion pipelines, and methods that use temporal or contextual information. YOLO-family models are attractive for real-time detection, while Mask R-CNN provides instance masks and can better represent irregular object shapes and the active rail region [5], [6]. TOSS adds an alert-decision stage, and the LeaderRank-based method places emphasis on identifying important monitoring points [2], [3]. These are not interchangeable contributions: one improves visual recognition, another combines sensing and warning, and another helps allocate monitoring resources.

A method should not be selected from a single reported accuracy value. The Mask R-CNN study reports a mean average precision of 0.9375 and a frame rate of 30 frames per second on its own dataset and setup [6]. Such a result is informative for that experiment, but it cannot be ranked directly against results from studies using different datasets, image sizes, class definitions, train/test splits, and hardware. Railway detection literature also explores lightweight networks, few-shot learning, and multi-mode feature enhancement [19], [20], [23], [24]. These directions are particularly relevant where the number of labelled intrusion examples is limited or inference must run on edge hardware.

Reported speed should be compared only when the hardware, image size, batch size, and timing procedure are known. In particular, frame rate measured on a high-end GPU may not reflect performance on an edge computer installed near the track. A deployment report should include end-to-end latency, including preprocessing and alert generation, rather than timing only the neural-network forward pass.

### D. Dataset Quality, Generalization, and Rare Events

Dataset quality is a common dependency across all architectures. The Mask R-CNN study describes a custom dataset of approximately 300 images, including realistic video-game footage, and an 80/20 training–testing split [6]. Synthetic or simulated imagery can help generate varied scenes, but performance on such data does not by itself establish reliability on real railway video. A stronger evaluation should include footage from different railway corridors, camera positions, seasons, lighting conditions, weather, and track curvature. It should also include rare but important cases such as partially occluded animals, small debris, people near the track edge, and objects that enter the track only briefly.

Semisupervised and few-shot approaches are relevant because dangerous intrusions are comparatively rare and expensive to annotate. A deep generative semisupervised approach for rail foreign-object detection and a few-shot intrusion method are among the related studies referenced by the supplied papers [8], [24]. Such approaches may reduce dependence on large labelled datasets, but they still require careful validation on unseen routes and genuinely independent video sessions. Splitting adjacent

frames from one video across training and testing can inflate measured performance because those frames may be nearly duplicates.

Rare-event learning should be evaluated with care because overall accuracy can remain high even when the model misses the few hazardous examples. Precision and recall should be reported by class, with the number of positive test examples stated where possible. Data augmentation and synthetic samples may help, but they should not replace validation on genuine, independently collected railway footage.

### E. Alert Reliability and Operational Integration

A railway safety system is not complete when it draws a bounding box. It must determine whether an object is inside or approaching the danger zone, estimate the persistence of the detection, and communicate a timely warning. TOSS explicitly incorporates a fuzzy-based alerting approach [2], while other reviewed papers emphasize object recognition or track-region detection more strongly. Related railway surveillance work in the supplied papers also discusses safety monitoring, video analytics, and risk-oriented warning [7], [11], [26]–[29]. Together, these works support treating the alert logic as a separately testable subsystem.

System-level measures should include false alerts per hour, missed intrusions, warning latency, alert stability across consecutive frames, and performance during temporary camera or sensor failure. A confidence threshold that improves precision may reduce recall, so threshold selection should reflect the relative consequences of false warnings and missed hazards. The system should also define how an alert is cleared, how duplicate warnings are suppressed, and what happens when the track detector is uncertain. These behaviours should be documented and tested rather than left implicit in the detector implementation.

Operational evaluation should record the complete sequence from first visible intrusion to warning delivery. Repeated detections of the same object should be grouped into a single event where appropriate, and the alert logic should be tested for unstable on/off behaviour. Human operators also need understandable alerts that communicate the detected class, approximate location, confidence, and reason for escalation.

### F. Recommended Benchmark and Reporting Practice

Future studies would benefit from a shared evaluation protocol that separates object detection, track localization, and end-to-end warning quality. Detection can be reported using precision, recall, F1-score, and mAP with the IoU threshold stated. Track masks can be evaluated using IoU or mean IoU. Real-time behaviour should include latency and frame rate on specified hardware, while operational safety should include false alerts per unit time and missed intrusions. Confidence intervals or repeated runs can help show whether small differences are meaningful.

The test set should contain both positive and negative examples. Positive cases should include full and partial intrusions, small distant objects, different object classes, and occlusions. Negative cases should include objects outside the danger zone, trackside vegetation, shadows, signs, and structures that may

trigger false detections. A location- or recording-session-based split is preferable to random frame-level splitting. This protocol would allow researchers to compare architectures more fairly and to distinguish a strong image detector from a dependable railway warning system.

A benchmark should document the recording location, view-point, weather and illumination conditions, object categories, annotation rules, and split strategy. It should include difficult negative examples as well as true intrusions, because a system that detects objects everywhere may generate too many warnings to be useful. Reporting these details makes it easier for other researchers to reproduce results and assess generalization.

### VIII. CONCLUSION

This survey reviewed six studies related to AI-based railway track object detection and alerting. The sources explore complementary techniques: improved U-Net and YOLOv5s, camera-LiDAR fusion with fuzzy alerts, monitoring-point selection and hybrid recognition, MSL-YOLO for lightweight multi-scale detection, MACE-Net, and Mask R-CNN-based track segmentation. The comparison suggests that no single architecture is best for every operating environment. Reliable deployment requires track-aware decisions, representative data, robustness to lighting and occlusion, computational efficiency, and validation of the complete alerting pipeline. Future work should use shared benchmarks and report both detection metrics and end-to-end warning performance.

The comparison indicates that reliable railway safety requires more than high object-detection accuracy: track-aware decision-making, environmental robustness, timely alerts, and reproducible field evaluation are equally important. Future work should validate the complete pipeline on diverse real railway footage and report detector, track-localization, and end-to-end alert metrics separately. Supporting references provide wider context, but claims about a specific method should be checked against the original source before being presented as established evidence.

The main conclusion is that railway intrusion detection should be assessed as a safety-oriented system rather than only as an image-recognition task. Model selection should follow the operating constraints, available data, sensor configuration, response-time requirement, and consequences of false alarms and missed hazards. The evidence reviewed here does not establish a universally superior model because the studies use different datasets and experimental protocols. Before operational use, any prototype requires independent field validation, documented failure analysis, and a clearly defined human response procedure.

### REFERENCES

- [1] S. Ning, F. Ding, and B. Chen, "Research on the Method of Foreign Object Detection for Railway Tracks Based on Deep Learning," *Sensors*, vol. 24, no. 14, Art. no. 4483, Jul. 2024, doi: 10.3390/s24144483.
- [2] D. Rajeswari, S. Rajendran, A. Arivarasi, A. Govindasamy, and A. Ahilan, "TOSS: Deep Learning-Based Track Object Detection Using Smart Sensor," *IEEE Sensors Journal*, vol. 24, no. 22, pp. 37678–37686, Nov. 2024, doi: 10.1109/JSEN.2024.3447730.
- [3] L. Niu, "Automatic Identification Method of Foreign Body Intrusion in Railway Transportation Track Based on Improved LeaderRank Identification of Key Points," *Egyptian Informatics Journal*, vol. 30, Art. no. 100682, Jun. 2025, doi: 10.1016/j.eij.2025.100682.
- [4] H. Niu, D. Feng, and T. Hou, "Research on foreign object intrusion detection in railway tracks based on MSL-YOLO," *Journal of Engineering and Applied Science*, vol. 72, Art. no. 136, 2025, doi: 10.1186/s44147-025-00708-7.
- [5] X. Chen, Y. Tian, M. Li, B. Lv, S. Zhang, Z. Qu, J. Wu, and S. Cheng, "Automatic Detection of Foreign Object Intrusion Along Railway Tracks Based on MACE-Net," *PLOS ONE*, vol. 20, no. 8, Art. no. e0329303, Aug. 2025, doi: 10.1371/journal.pone.0329303.
- [6] I. Mahmud, M. M. Kabir, J. Shin, C. Mistry, Y. Tomioka, and M. F. Mridha, "Advancing Wildlife Protection: Mask R-CNN for Rail Track Identification and Unwanted Object Detection," *IEEE Access*, vol. 11, pp. 99519–99534, 2023, doi: 10.1109/ACCESS.2023.3313253.
- [7] S. Ning, R. Guo, P. Guo, L. Xiong, and B. Chen, "Research on Foreign Object Intrusion Detection for Railway Tracks Utilizing Risk Assessment and YOLO Detection," *IEEE Access*, vol. 12, pp. 175926–175939, 2024, doi: 10.1109/ACCESS.2024.3504010.
- [8] T. Wang, Z. Zhang, and K. L. Tsui, "A Deep Generative Approach for Rail Foreign Object Detections via Semisupervised Learning," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 1, pp. 459–468, 2023, doi: 10.1109/TII.2022.3149931.
- [9] B. Zhang, Q. Yang, F. Chen, and D. Gao, "A Real-Time Foreign Object Detection Method Based on Deep Learning in Complex Open Railway Environments," *Journal of Real-Time Image Processing*, vol. 21, no. 5, Art. no. 166, 2024, doi: 10.1007/s11554-024-01548-z.
- [10] C. Meng, Z. Wang, L. Shi, Y. Gao, Y. Tao, and L. Wei, "SDRC-YOLO: A Novel Foreign Object Intrusion Detection Algorithm in Railway Scenarios," *Electronics*, vol. 12, no. 5, Art. no. 1256, 2023, doi: 10.3390/electronics12051256.
- [11] Z. Zhang, P. Chen, Y. Huang, L. Dai, F. Xu, and H. Hu, "Railway Obstacle Intrusion Warning Mechanism Integrating YOLO-Based Detection and Risk Assessment," *Journal of Industrial Information Integration*, vol. 38, Art. no. 100571, 2024, doi: 10.1016/j.jii.2024.100571.
- [12] J. Wang, H. Zhai, Y. Yang, N. Xu, H. Li, and D. Fu, "A Review of Intrusion Detection for Railway Perimeter Using Deep Learning-Based Methods," *IEEE Access*, vol. 12, pp. 184142–184157, 2024, doi: 10.1109/ACCESS.2024.3510746.
- [13] X. Song, H. Song, H. Wang, Z. Zhang, and H. Dong, "Deep Learning-Based Railway Foreign Object Intrusion Intelligent Perception Using Attention-Aggregated Semantic Segmentation," *IEEE/ASME Transactions on Mechatronics*, vol. 30, no. 4, pp. 2609–2619, 2025, doi: 10.1109/TMECH.2024.3468620.
- [14] Z. Cao, Y. Qin, L. Jia, Z. Xie, Y. Gao, Y. Wang, P. Li, and Z. Yu, "Railway Intrusion Detection Based on Machine Vision: A Survey, Challenges, and Perspectives," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 7, pp. 6427–6448, Jul. 2024, doi: 10.1109/TITS.2024.3412170.
- [15] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2980–2988.
- [16] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single Shot MultiBox Detector," in *Computer Vision – ECCV 2016*, 2016, pp. 21–37.
- [17] P. Singh, M. Dulebenets, J. Pasha, D. Ernesto, S. Gonzalez, Y. Lau, and R. Kampmann, "Deployment of Autonomous Trains in Rail Transportation: Current Trends and Existing Challenges," *IEEE Access*, vol. 9, pp. 91427–91461, 2021.
- [18] M. Simonović et al., "Toward the Enhancement of Rail Sustainability: Demonstration of a Holistic Approach to Obstacle Detection in Operational Railway Environments," *Sustainability*, vol. 16, Art. no. 2613, 2024.
- [19] L. Guan, L. Jia, Z. Xie, and C. Yin, "A Lightweight Framework for Obstacle Detection in the Railway Image Based on Fast Region Proposal and Improved YOLO-Tiny Network," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–16, 2022.

- [20] T. Ye, Z. Zhang, X. Zhang, and F. Zhou, "Foreign Body Detection in Rail Transit Based on a Multi-Mode Feature-Enhanced Convolutional Neural Network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, pp. 18051–18063, 2022.
- [21] D. Ristić-Durrant, M. Franke, and K. Michels, "A Review of Vision-Based On-Board Obstacle Detection and Distance Estimation in Railways," *Sensors*, vol. 21, Art. no. 3452, 2021.
- [22] Z. Nan et al., "Development of a High-Precision LiDAR System and Improvement of Key Steps for Railway Obstacle Detection Algorithm," *Remote Sensing*, vol. 16, Art. no. 1761, 2024.
- [23] T. Ye, Z. Zheng, X. Li, Z. Zhao, and X. Gao, "An Efficient Few-Shot Object Detection Method for Railway Intrusion via Fine-Tune Approach and Contrastive Learning," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–13, 2023.
- [24] S. Wang, Y. Wang, Y. Chang, R. Zhao, and Y. She, "EBSE-YOLO: High Precision Recognition Algorithm for Small Target Foreign Object Detection," *IEEE Access*, vol. 11, pp. 57951–57964, 2023.
- [25] R. Kapoor, R. Goel, and A. Sharma, "An Intelligent Railway Surveillance Framework Based on Recognition of Object and Railway Track Using Deep Learning," *Multimedia Tools and Applications*, vol. 81, pp. 21083–21109, 2022, doi: 10.1007/s11042-022-12059-z.
- [26] P. Sikora et al., "Artificial Intelligence-Based Surveillance System for Railway Crossing Traffic," *IEEE Sensors Journal*, vol. 21, no. 14, pp. 15515–15526, 2021, doi: 10.1109/JSEN.2020.3031861.
- [27] T. Zhang et al., "Recent Advances in Video Analytics for Rail Network Surveillance for Security, Trespass and Suicide Prevention—A Survey," *Sensors*, vol. 22, no. 12, Art. no. 4324, 2022, doi: 10.3390/s22124324.
- [28] C.-C. Chang, K.-H. Huang, T. K. Lau et al., "Using Deep Learning Model Integration to Build a Smart Railway Traffic Safety Monitoring System," *Scientific Reports*, vol. 15, Art. no. 4224, 2025, doi: 10.1038/s41598-025-88830-7.
- [29] S. Meng, W. Chen, and Y. Jiang, "A Novel Strategy of Two-Stage Cascaded CNN and Overhaul Knowledge Distillation for Fast Railway Foreign Objects Intrusion Detection," *IEEE Access*, vol. 12, pp. 127825–127835, 2024, doi: 10.1109/ACCESS.2024.3457515.
- [30] F. Tan, M. Zhai, and C. Zhai, "Foreign object detection in urban rail transit based on deep differentiation segmentation neural network," *Heliyon*, vol. 10, no. 17, Art. no. e37072, Sep. 2024, doi: 10.1016/j.heliyon.2024.e37072.
- [31] T. Hu, F. Gao, and F. Zhou, "Railway obstacle intrusion detection and risk assessment based on MSIA-YOLOv8 and DALNet," *Expert Systems with Applications*, vol. 299, Part C, Art. no. 130132, 2026, doi: 10.1016/j.eswa.2025.130132.
- [32] H. Mahamivanan, J. Matthews, P. E. D. Love, and F. Nasirzadeh, "Toward accurate detection of small objects in rail construction: A deep learning perspective," *Engineering Applications of Artificial Intelligence*, vol. 160, Part B, Art. no. 111977, Nov. 2025, doi: 10.1016/j.engappai.2025.111977.

## AUTHORS' DETAILS

---

### **Dr. Sincy V. Thambi**

Dr. Sincy V. Thambi is an Associate Professor and Head of the Department of Artificial Intelligence and Data Science at Royal College of Engineering and Technology (RCET), Kerala, India. She received her B.Tech. degree in Computer Science and Engineering from College of Engineering Thalassery in 2012 and her M.Tech. degree in Computational Linguistics from Government Engineering College, Sreekrishnapuram, in 2014. She received her Ph.D. from APJ Abdul Kalam Technological University (KTU), Kerala, for research in Natural Language Processing and Knowledge Graph-based Question Answering. She has seven years of teaching experience and five years of research experience. Her research interests include Natural Language Processing, Knowledge Graphs, Question Answering Systems, Machine Learning, Deep Learning, and Large Language Models, particularly for information retrieval and question answering in low-resource languages such as Malayalam.

---

### **Aysha Nidha**

Aysha Nidha is an undergraduate student pursuing a Bachelor of Technology (B.Tech.) degree in Artificial Intelligence and Data Science at Royal College of Engineering and Technology, Akkikkavu, Thrissur, Kerala, India. Her research interests include deep learning, computer vision, object detection, and railway safety. Her current research focuses on “Automatic Detection of Foreign Object Intrusion Along Railway Tracks Based on MACENet,” which aims to improve railway safety through the application of advanced deep learning and computer vision techniques for detecting foreign objects along railway tracks.

---

### **Sneha P.**

Sneha P. is an undergraduate student pursuing a Bachelor of Technology (B.Tech.) in Artificial Intelligence and Data Science at Royal College of Engineering and Technology, Akkikkavu, Thrissur, Kerala, India. Her academic interests include artificial intelligence, deep learning, computer vision, and intelligent surveillance systems. Her current research focuses on developing an AI-based railway track monitoring system that uses object detection techniques to identify foreign objects on railway tracks and support early risk identification, with the broader goal of enhancing railway safety and operational reliability.

## AUTHORS' DETAILS

---

### **Souparnika M. P.**

Souparnika M. P. is pursuing a Bachelor of Technology (B.Tech.) in Artificial Intelligence and Data Science at Royal College of Engineering and Technology, Akkikkavu, Thrissur, Kerala, India. Her research interests include deep learning, computer vision, object detection, and railway safety. Her ongoing research focuses on developing an automated system for detecting foreign object intrusion on railway tracks using deep learning techniques, with the objective of enhancing railway safety and improving monitoring efficiency.

---

### **Twinkle P. Toms**

Twinkle P. Toms is an undergraduate student in the Department of Artificial Intelligence and Data Science at Royal College of Engineering and Technology, Akkikkavu, Thrissur, Kerala, India. Her academic interests include intelligent computing, machine learning, and computer vision applications. Her project, titled “AI-Based Railway Track Object Detection and Alert System,” explores the use of artificial intelligence and object detection techniques to identify foreign objects on railway tracks. The work aims to facilitate early recognition of potential track hazards and support the development of safer and more reliable railway transportation systems.