Japan Bilingual Publishing Co.

Transportation Development Research

https://ojs.bilpub.com/index.php/tdr

ARTICLE

AI-Based Train Localization Using Railway Infrastructure Object Detection

Vladimir A. Fedorov [1,2*]

[1]*Department of Electronic Engineering, Ural Federal University, Yekaterinburg 620002, Russia*

[2]*Department of Research and Development for Integrated Security Systems, NPO SAUT LLC, Yekaterinburg 620027, Russia*

ABSTRACT

This paper presents an Artificial Intelligence (AI)-based approach to train localization through the detection of railway infrastructure objects using convolutional neural networks. The proposed system identifies key visual landmarks such as traffic lights, level crossings, tunnels, bridges, and passenger platforms directly from live video streams captured by onboard cameras during train operation. This enables accurate and continuous localization without relying on satellite navigation systems or additional trackside infrastructure. The object detection model is based on the You Only Look Once (YOLOv11) architecture. It is trained using high-performance Graphics Processing Unit (GPU) resources and subsequently converted and optimized for deployment on the energy-efficient RK3588 neural processing unit (NPU). The system achieves a mean average precision of mAP@0.5:0.95 = 0.52 and operates in real time at approximately 35 frames per second, meeting the practical requirements for onboard applications. Compared to traditional Global Navigation Satellite System (GNSS)-based solutions, the proposed method is inherently resilient to signal jamming and spoofing while significantly reducing infrastructure costs. Its low power consumption and high-speed inference make it especially well-suited for integration into modern railway systems operating at higher automation levels. The results confirm the feasibility of this AI-driven approach as a scalable and robust solution for train localization in diverse operational conditions.

Keywords: Railway Automation; Train Localization; Railway Infrastructure; Object Detection; Convolutional Neural Networks; YOLOv11; NPU

*CORRESPONDING AUTHOR:

Vladimir A. Fedorov, Department of Electronic Engineering, Ural Federal University, Yekaterinburg 620002, Russia; Department of Research and Development for Integrated Security Systems, NPO SAUT LLC, Yekaterinburg 620027, Russia; Email: va.fedorov@outlook.com

ARTICLE INFO

Received: 12 June 2024 | Revised: 28 August 2024 | Accepted:2 September 2024| Published Online: 20 September 2024

DOI: https://doi.org/10.55121/tdr.v2i2.379

CITATION

Fedorov, V. A., 2024. AI-Based Train Localization Using Railway Infrastructure Object Detection. Transportation Development Research. 2(2): 1–11. DOI: https://doi.org/10.55121/tdr.v2i2.379

1. Introduction

The ongoing development of railway automation, particularly the transition toward Grade of Automation 3 (GoA3) and Grade of Automation 4 (GoA4) [1], imposes increasingly stringent requirements on train control and localization systems. At these higher automation levels, where train operation is performed with minimal or no onboard staff, ensuring precise, reliable, and real-time positioning is critical for maintaining safety and operational efficiency.

Traditionally, Global Navigation Satellite Systems (GNSS), including GPS, GLONASS, Galileo, BeiDou, QZSS, and NavIC, have been employed for train localization due to their wide availability and global or regional coverage. However, applying GNSS in safety-critical applications such as autonomous train operation remains challenging. Issues such as signal blockage in tunnels, urban canyons, and forested areas; multipath effects; susceptibility to jamming and spoofing; reliance on foreign-operated space infrastructure; and limited update rates pose significant limitations to ensuring continuous and trustworthy localization [2].

To overcome these shortcomings, alternative or complementary localization technologies are being actively investigated. Among them, computer vision systems designed to detect railway infrastructure objects have the potential to serve as either a standalone alternative to GNSS or as an augmentation method to enhance the robustness and reliability of train localization. These systems employ onboard cameras in combination with artificial intelligence algorithms to identify and classify characteristic infrastructure components. By recognizing such elements and correlating them with a priori georeferenced data, the train's position can be estimated with high accuracy, independently of external radio signals.

This paper presents a novel vision-based localization approach for railway vehicles, which utilizes convolutional neural networks (CNNs) [3] for the real-time detection of railway infrastructure objects. The proposed system is based on the YOLOv11 (You Only Look Once, version 11) [4] object detection architecture and is capable of identifying critical elements such as signals, level crossings, platforms, tunnels, and bridges from video data captured by a forward-facing onboard camera. The recognized infrastructure components are matched to a pre-mapped database, enabling continuous and dynamic train position estimation.

The proposed method offers several advantages: it increases resilience to GNSS signal loss or degradation, enables operation in GNSS-denied environments, and does not require any additional trackside infrastructure. Consequently, the approach is well-suited for integration into next-generation railway automation systems, providing a scalable and reliable localization solution in support of high levels of automation.

2. Related Work

Train localization has traditionally relied on a diverse set of techniques, including satellite navigation systems and trackside infrastructure sensors. While each approach offers specific advantages, they also present distinct limitations – particularly in the context of autonomous and semi-autonomous railway operations.

2.1. Global Navigation Satellite Systems (GNSS)

Global Navigation Satellite Systems, such as GPS (United States of America), GLONASS (Russia), Galileo (European Union), BeiDou (China), QZSS (Japan), and NavIC (India), have been widely adopted for positioning and navigation across various transportation sectors, including rail. GNSS offers global coverage and accurate positioning under clear-sky conditions, making it a practical solution for large-scale railway networks.

However, despite their widespread use, GNSS-based systems face several limitations – especially in safety-critical applications such as autonomous train control:

- Signal blockage – GNSS signals can be obstructed in tunnels, cuttings, under bridges, and within urban canyons, leading to temporary loss of positioning [5].
- Multipath effects – Reflections from buildings, terrain, or other surfaces can distort signal paths and result in inaccurate positioning [6].
- Signal jamming – Due to their low power at ground level, GNSS signals are vulnerable to both intentional and unintentional radio frequency interference [7].

- Spoofing attacks – Malicious entities can transmit counterfeit GNSS signals, causing receivers to compute false positions [8].
- Dependency on external infrastructure – GNSS relies on space-based assets that are often operated by foreign governments, which may raise concerns about availability, security, or geopolitical risks.

These limitations have driven research into alternative and complementary localization techniques.

2.2. Wheel Odometry

Wheel odometry is a traditional method for estimating train position by counting wheel rotations and calculating the traveled distance based on the known diameter of the wheel. This technique is widely used in railway applications due to its simplicity, independence from external infrastructure, and low implementation cost.

However, this method is prone to cumulative errors arising from wheel slip or slide, which are particularly common during braking, acceleration, or operation under low-adhesion conditions (e.g., wet or icy rails) [9]. Another major source of error is the gradual reduction in wheel diameter over time due to wear of the wheel tread and, more specifically, the steel tyre (wheel bandage). Since the calculation of distance is directly dependent on the wheel's diameter, even small changes can lead to significant position estimation errors over long distances.

To mitigate this, it is necessary to regularly measure the diameter of the bandage on the wheelsets used for odometry. These measurements allow for timely recalibration of the odometry system, ensuring that distance calculations remain accurate despite mechanical wear. In practice, such measurements are typically carried out during scheduled maintenance using portable measuring tools or stationary systems installed in maintenance depots.

Despite regular corrections, standalone wheel odometry is often insufficient for applications requiring high-precision localization, such as autonomous train operation. Therefore, it is commonly used in conjunction with complementary positioning technologies, including GNSS, inertial navigation systems, and trackside transponders, to enhance accuracy, provide redundancy, and reduce the impact of drift and slippage in challenging operating environments.

2.3. Inertial Navigation Systems (INS)

Inertial Navigation Systems (INS) estimate a vehicle's position, velocity, and orientation by processing signals from accelerometers and gyroscopes through a process known as dead reckoning. These systems operate independently of external signals, making them particularly valuable in environments where GNSS is unavailable, such as tunnels, urban canyons, or areas affected by signal interference. As a result, INS is frequently used as a fallback solution when GNSS data is temporarily inaccessible [10].

One of the key advantages of INS is its ability to provide continuous localization without requiring any external infrastructure. This autonomy makes it highly suitable for integration into onboard train control systems, where uninterrupted position data is essential for ensuring safety and automation continuity. However, the major drawback of inertial navigation is the gradual accumulation of errors over time, often referred to as drift. Since INS calculates position incrementally based on previous measurements, even small sensor inaccuracies can compound, resulting in significant deviations from the actual position unless periodic corrections from an external reference are applied.

To address this issue, advanced positioning systems often employ GNSS/INS integration, where GNSS data is used to regularly correct the drift inherent in INS calculations [11]. This sensor fusion approach significantly enhances localization robustness by combining the high accuracy of GNSS with the continuity of INS. However, in scenarios where GNSS updates are entirely unavailable, such as during long tunnel segments or in the presence of GNSS jamming or spoofing, the effectiveness of this fusion is compromised. Without reliable external corrections, the INS continues to drift, leading to increasing localization errors over time.

Such inaccuracies become particularly problematic in the context of autonomous train operation, where precise positioning is critical for safe braking, station stopping, and track switching. In safety-critical applications, prolonged reliance on uncorrected INS data may result in deviations large enough to compromise operational integrity, highlighting the need for additional redundancy or alternative localization strategies in GNSS-denied environments.

2.4. Radio-Frequency Identification (RFID)

RFID-based localization employs tags, either passive, powered by the reader's interrogation signal, or active, equipped with their own power supply, strategically affixed alongside or between the rails, while RFID readers are installed on the train's undercarriage or near its leading bogie [12]. As the train moves, the onboard reader periodically energizes passive tags (or polls active tags), receives their unique identifiers, and thereby determines that it has passed a known reference point. By correlating each tag's ID with its pre-surveyed track location, the system can update the train's position with sub-meter accuracy at each tag crossing.

Because RFID does not rely on satellite signals, it is immune to GNSS-specific vulnerabilities such as multipath interference in urban canyons, signal blockage in tunnels, or intentional jamming and spoofing. This resilience, combined with the fine granularity of tag spacing, makes RFID localization attractive for safety-critical applications where reliable position updates are essential.

Despite these benefits, widespread adoption of RFID localization faces significant practical challenges. Deploying thousands of tags along hundreds of kilometers of track demands substantial upfront investment in hardware, installation labor, and track-side power or maintenance access for active tags. Even passive tags, while lower-cost per unit, require careful mounting, periodic inspection, and replacement when damaged or degraded. As a result, the total cost of ownership can be prohibitive for large networks.

Furthermore, RFID systems inherently produce discrete position updates only at tag locations. Between tags, the train must rely on onboard odometry or inertial sensors, which accumulate drift until the next tag is read. Consequently, the effective localization resolution and continuity are bounded by tag spacing: closer spacing yields more frequent updates but further raises infrastructure costs. This trade-off between accuracy, update rate, and deployment expense remains a key limitation when considering RFID for continuous train positioning over long mainline routes.

2.5. Balises

Balises play a central role in many modern train control systems by acting as fixed reference points along the track. In the European Train Control System (ETCS), for example, Eurobalises are mounted between the rails at predetermined locations and serve as passive transponders: when a train passes over a balise, the onboard antenna energizes it and the balise then transmits stored data back to the train's onboard unit [13]. That data typically includes the balise's exact geographic reference, line identification, permissible speed, and any temporary speed restrictions or movement authorities. By periodically "reading" these waypoints, the train continually refreshes its understanding of its location with meter-level accuracy, which is essential for enforcing braking curves, maintaining safe separation, and enabling higher levels of automation.

Because balises do not rely on continuous radio links or satellite signals, they offer exceptional reliability and immunity to electromagnetic interference, attributes that are vital in safety-critical railway environments. Their passive design also means they have no onboard power source and very low failure rates, contributing to system robustness and safety certification.

However, this fixed-point approach also introduces inherent limitations. Position information is only updated when the train crosses a balise, resulting in discrete location "jumps" rather than a continuous position track. Between balises, the train must rely on less precise odometry or inertial measurements, which can accumulate error over distance. Moreover, the installation and upkeep of balises, and the associated track-side infrastructure such as mounting assemblies, cables, and data concentrators, add capital and operational expenses. Regular inspection, cleaning, and occasional replacement of balise units are required to ensure long-term reliability, which increases life-cycle costs compared to purely onboard or wireless positioning solutions.

2.6. AI-Based Methods

Recent advancements in computer vision and artificial intelligence (AI) have introduced new opportunities for train localization. AI-based approaches, particularly those employing CNNs, can identify and interpret railway infrastructure elements from visual data captured by onboard cameras.

Techniques such as object detection using the YOLO (You Only Look Once) family of models have demonstrated high performance in detecting railway signals, switches, signs, level crossings, and obstacles [14,15]. These methods are especially attractive because they rely on visual features of existing infrastructure and can operate effectively in GNSS-denied environments, such as tunnels or urban canyons.

In contrast to inertial navigation systems (INS), which suffer from accumulated drift over time and require frequent correction from external sources, AI-based visual methods do not accumulate positional errors in the same way. Each detection of a known infrastructure element provides a new, independent cue for localization.

Furthermore, unlike RFID-based systems and balise-based solutions, which require widespread and costly trackside equipment installation and maintenance along the entire rail network, AI-based systems only require onboard sensors and computing hardware. This makes them more scalable and cost-effective, particularly for retrofitting existing rolling stock.

Another key advantage is the abundance and diversity of railway infrastructure objects, such as kilometer and hectometer posts, signals, level crossings, bridges, tunnels, and platforms, which are spatially distributed along the tracks. These landmarks serve as frequent and reliable reference points, enabling the continuous update of the train's estimated position using visual data captured from onboard cameras.

When integrated with prior knowledge of track topology or map-matching techniques, AI-based localization can either enhance traditional methods or serve as a standalone solution for train position estimation in the context of autonomous or semi-autonomous railway operations.

3. Materials and Methods

Considering the advantages offered by AI-based localization methods, such as robustness to signal interference, independence from GNSS, and minimal reliance on external infrastructure, this work proposes a train localization approach based on the detection of railway infrastructure elements using convolutional neural networks.

3.1. Dataset Description

In order to effectively train convolutional neural networks for object detection tasks, a large and diverse dataset of labeled images is essential. The quality, variability, and annotation accuracy of the dataset directly influence the performance, generalization capability, and reliability of the resulting model, especially in safety-critical domains such as railway transportation.

To train the convolutional neural network, a custom dataset of railway infrastructure objects was manually created. The dataset was constructed from video recordings captured by a forward-facing camera mounted on a train during regular trips under various weather conditions and at different times of the day. This approach ensured a wide variety of source images, enhancing the robustness and generalization ability of the trained model.

All images in the dataset have a resolution of at least Full HD (1920×1080 pixels), providing sufficient detail for accurate object detection. **Figure 1** illustrates several representative images from the dataset.

Figure 1. Example images from the dataset.

Each image was manually annotated with bounding boxes around key railway infrastructure elements, including passenger platforms, level crossings, traffic lights, tunnels, bridges, and other relevant features.

The final dataset consists of more than 20,000 annotated images, making it a substantial resource for training and evaluating deep learning models in the context of railway infrastructure detection and train localization.

3.2. YOLOv11 Architecture

To detect objects in railway infrastructure, we use the YOLOv11 model as the convolutional neural network. YOLOv11 represents an improved version of the previous YOLO series and is specifically designed to provide high detection accuracy while maintaining real-time inference speed.

The architecture of YOLOv11 consists of three main components:

- Backbone – responsible for extracting visual features from the input image,
- Neck – aggregates and refines features at multiple scales,
- Head – performs object detection based on the processed features.

The composition and interaction of these components are illustrated in **Figure 2**.

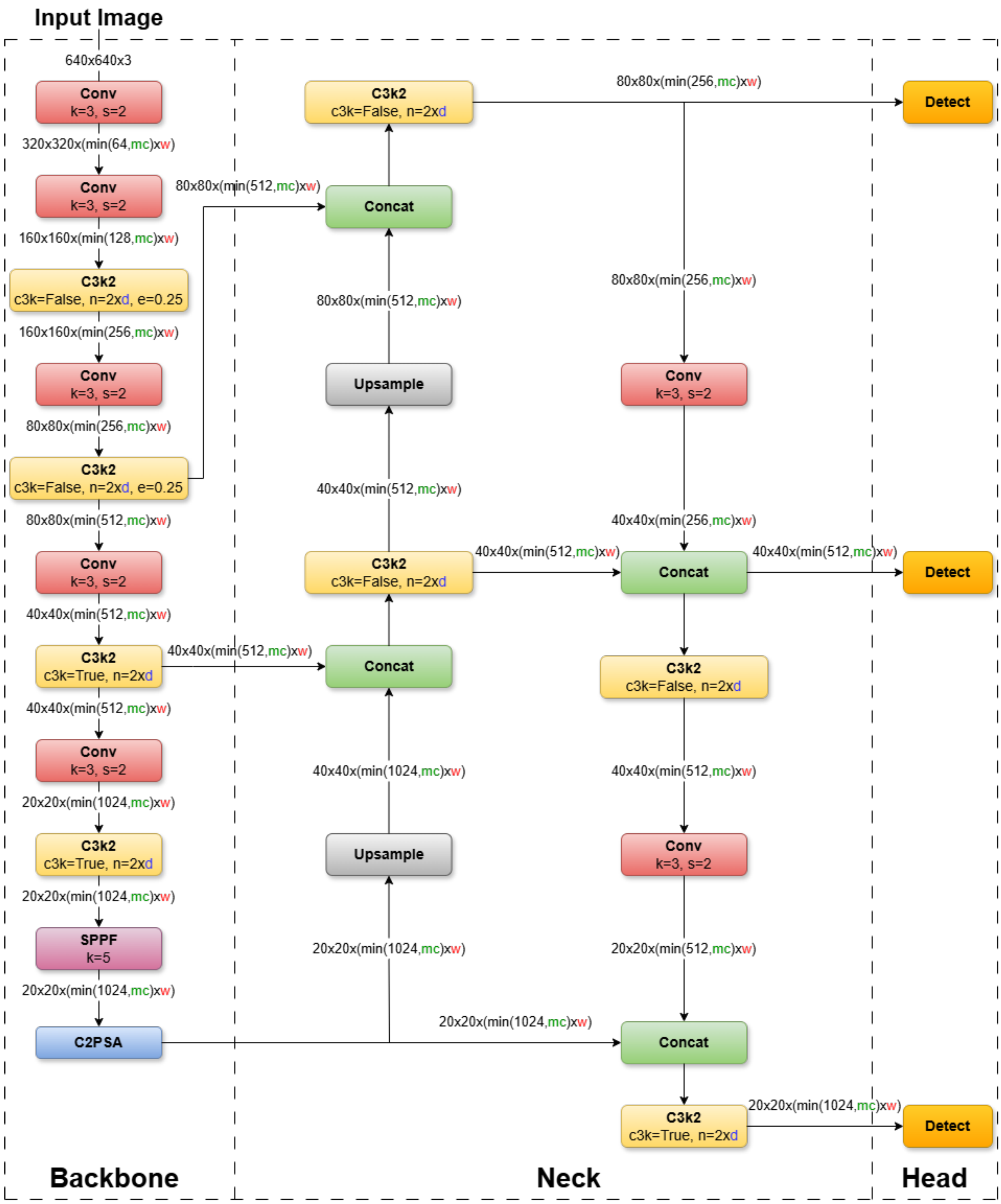

Figure 2. Architecture of YOLOv11.

YOLOv11 is available in five model variants: n (nano), s (small), m (medium), l (large), and x (extra-large) [16]. These variants differ in three key architectural parameters: d (depth multiple), w (width multiple), and mc (maximum number of channels). The parameter d controls the depth of the network by scaling the number of layers; w adjusts the width by modifying the number of channels in each layer; and mc defines the maximum number of channels allowed in any convolutional layer. All model versions and their corresponding parameter values are presented in **Table 1**.

Table 1. Configuration of YOLOv11 model variants according to architectural parameters.

Model Variant	d (Depth Multiple)	w (Width Multiple)	mc (Max Channels)
n (nano)	0.50	0.25	1024
s (small)	0.50	0.50	1024
m (medium)	0.50	1.00	512
l (large)	1.00	1.00	512
x (extra-large)	1.00	1.50	512

YOLOv11 introduces several architectural enhancements over previous versions, aimed at increasing the efficiency and accuracy of object detection, particularly for small and partially occluded objects:

1. C3K2 Block

This block is an evolution of the CSP (Cross Stage Partial) [17] bottleneck structure and serves as the core of the improved backbone. It utilizes compact 3×3 convolutional kernels, which reduce computational load while preserving the model's ability to extract fine-grained features. Unlike earlier structures, C3K2 integrates several internal C3K [18] blocks and merges outputs to maintain rich spatial information with fewer parameters.

2. Spatial Pyramid Pooling Fast (SPFF)

The SPFF module [19] is used in the neck of the architecture to combine features from different scales. It applies multiple max-pooling operations with varying kernel sizes to extract contextual information from both small and large regions of the image. This significantly improves the network's ability to detect objects of different sizes, especially small ones, without sacrificing speed.

3. C2PSA Attention Mechanism

YOLOv11 integrates the C2PSA (Cross Stage Partial with Spatial Attention) block [20] to enhance the model's focus on the most informative areas of an image. It combines spatial attention with partial feature sharing to emphasize important regions, such as small signs or partially obscured objects. This mechanism helps the model to selectively refine feature maps and contributes to higher detection accuracy.

4. Multi-Scale Prediction Head

Similar to previous YOLO models, the head of YOLOv11 generates predictions at three different scales, corresponding to varying levels of spatial resolution. This ensures effective detection of objects ranging from small to large by leveraging feature maps with appropriate detail levels.

Together, these innovations allow YOLOv11 to outperform previous versions in both accuracy and efficiency, making it well-suited for real-time object detection tasks in railway infrastructure environments.

3.3. Training and Deployment

To train the object detection model, we utilize a GPU-based computing environment, specifically the NVIDIA RTX A6000, which provides high computational performance of 38.71 TFLOPS (Tera Floating Point Operations Per Second) and supports the efficient training of deep convolutional neural networks. Training deep models requires processing large volumes of annotated data and performing extensive backpropagation operations, which are highly parallelizable and benefit significantly from GPU acceleration. The use of a GPU during training enables faster convergence and allows for experimenting with different model architectures and hyperparameters within a practical timeframe. Since training is performed in a controlled, stationary environment, power consumption is not a limiting factor, making high-performance GPUs the preferred choice.

In contrast, deployment on board a train requires a different set of hardware considerations. Mobile and embedded systems used in railway environments must adhere to strict requirements related to low power consumption, compact form factor, and robustness to vibration and temperature variations. For this reason, inference is performed on the RK3588 NPU, which delivers up to 6 TOPS (Tera Operations Per Second) of performance while maintaining a low power profile. This enables real-time object detection under operational conditions without exceeding the system's thermal and power budgets.

Thus, the use of a GPU during training ensures high model accuracy through intensive computation, while deployment on an NPU guarantees efficient and reliable inference in the constrained environment of railway operations.

The overall training and deployment pipeline is illustrated in **Figure 3**.

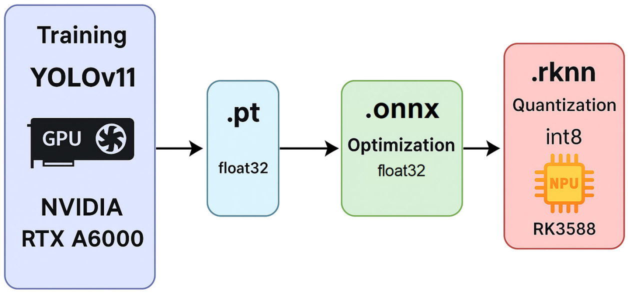

Figure 3. Pipeline of the model training and deployment process.

After completing the training phase, the convolutional neural network model is saved in the «.pt» format (PyTorch) [21], which contains both the learned weights and the network architecture. While this format is well-suited for training and experimentation, it is not optimal for deployment on edge devices. Therefore, the next step involves converting the trained model to the «.onnx» format (Open Neural Network Exchange) [22]. ONNX provides a platform-independent representation of the model and serves as an intermediate stage before final deployment.

To improve inference efficiency, especially on low-power edge devices such as the RK3588 NPU, several optimization steps are applied during the ONNX conversion process:

- Change output node – The model's output node is modified to ensure compatibility with the target inference engine and to simplify post-processing.
- Remove post-processing block – Components related to prediction refinement are excluded from the model, as they are not optimal for quantization and tend to reduce inference speed on embedded hardware.
- Remove DFL (Distribution Function Learning) structure – Although DFL can improve detection accuracy, it significantly increases computational overhead. Removing it reduces latency during inference on the NPU.
- Add score-sum output branch – An additional output branch is introduced to accelerate score aggregation during post-processing, thereby enhancing overall detection speed.

After optimization, the resulting ONNX model is further converted into the «.rknn» format (Rockchip Neural Network) [23] to enable deployment on the RK3588 NPU. This conversion includes integer quantization (int8), which is a key step in adapting the model for efficient inference on edge hardware. Quantization reduces the model's memory footprint, improves execution speed, and increases energy efficiency, all while maintaining a reasonable trade-off between detection accuracy and computational performance.

The RK3588 NPU, with a peak performance of 6 TOPS, is well-suited for real-time deep learning inference, particularly when combined with a quantized model. Applying integer quantization ensures that the model runs efficiently and reliably in real-time detection tasks under constrained computational and power conditions.

4. Results and Discussion

To evaluate the performance of the proposed YOLOv11s model for railway infrastructure object detection, both accuracy and inference speed were assessed using a dedicated test dataset. This dataset included a diverse range of images captured under varying weather conditions, lighting, and perspectives to simulate real-world railway scenarios.

The accuracy of the model was measured using the mean Average Precision (mAP) metric [24], which is widely adopted in object detection tasks. Specifically, we report the mAP@0.5:0.95, which represents the average precision calculated across multiple Intersection over Union (IoU) thresholds, ranging from 0.5 to 0.95 in steps of 0.05 [25]. This stricter evaluation criterion provides a more comprehensive assessment of the model's localization and classification performance.

The YOLOv11s model achieved an mAP@0.5:0.95 of 0.52, indicating that the model is capable of reliably detecting and localizing railway infrastructure elements, such as passenger platforms, traffic lights, level crossings, tunnels, and bridges, even in complex scenes and under challenging conditions. Examples of detected railway infrastructure objects in images are shown in **Figure 4**.

Figure 4. Sample detection results under various environmental conditions.

In addition to accuracy, real-time performance is critical for deployment in railway applications, where rapid detection is essential for timely decision-making. The optimized and quantized model was deployed on the RK3588 NPU, and its inference speed was measured on live video streams with a resolution of 640×640 pixels.

The results show that the model achieves an inference speed of approximately 35 frames per second (FPS). This level of performance meets the requirements for real-time object detection on embedded platforms, ensuring that the system can operate effectively in dynamic railway environments.

The combination of moderate detection accuracy and high inference speed makes the YOLOv11s model a suitable candidate for embedded deployment in railway systems. While the achieved mAP@0.5:0.95 of 0.52 suggests room for further improvement in detection precision, the real-time capability of 35 FPS ensures practical applicability in safety-critical scenarios. Future research may focus on refining the training dataset, incorporating more attention mechanisms, or exploring advanced model compression techniques to further enhance both accuracy and efficiency.

5. Conclusions

This study demonstrates the feasibility and effectiveness of using convolutional neural networks (CNNs) for the detection of railway infrastructure objects as a means of train localization. By detecting key visual landmarks, such as traffic lights, level crossings, kilometer posts, tunnels, bridges, and platforms, the proposed system allows for accurate and continuous train localization without the need for satellite navigation or the installation of additional trackside infrastructure.

The developed solution, based on the YOLOv11s model, achieves a detection accuracy of mAP@0.5:0.95 = 0.52 and a real-time inference speed of approximately 35 frames per second on the energy-efficient RK3588 NPU. These results confirm that the system meets the practical requirements for onboard deployment, especially in scenarios demanding low power consumption and real-time performance.

Unlike GNSS-based localization systems, the AI-driven approach is inherently robust against signal jamming and spoofing and does not require the installation or maintenance of expensive trackside equipment. This makes it particularly attractive for scalable and cost-effective deployment across various railway lines.

Moreover, the entire perception and localization process is performed onboard the train using standard video cameras and deep learning algorithms. This infrastructure-independent strategy aligns well with the goals of higher levels of automation in railway operations, particularly GoA3 and GoA4, and paves the way for more autonomous and intelligent train control systems in the future.

Funding

This work received no external funding.

Institutional Review Board Statement

Not applicable.

Informed Consent Statement

Not applicable.

Data Availability Statement

The dataset created during the current study is not publicly available due to privacy restrictions imposed by NPO SAUT LLC.

Conflicts of Interest

The author declares no conflict of interest.

References

[1] Barruffo, L., Caiazzo, B., Petrillo, A., & Santini, S. (et al., 2024. A GoA4 control architecture for the autonomous driving of high-speed trains over ETCS: Design and experimental validation. IEEE Transactions on Intelligent Transportation Systems. 25(6), 5096–5111. DOI: https://doi.org/10.1109/TITS.2023.3338295

[2] Zidan, J., Adegoke, E.I., Kampert, E., et al., 2020. GNSS vulnerabilities and existing solutions: A review of the literature. IEEE Access. 9, 153960–153976. DOI: https://doi.org/10.1109/ACCESS.2020.2973759

[3] Li, Z., Liu, F., Yang, W., et al., 2021. A survey of convolutional neural networks: Analysis, applications, and prospects. IEEE Transactions on Neural Networks and Learning Systems. 33(12), 6999–7019. DOI: https://doi.org/10.1109/TNNLS.2021.3084827

[4] Khanam, R., Hussain, M., 2024. Yolov11: An overview of the key architectural enhancements. [Preprint]. DOI: https://doi.org/10.48550/arXiv.2410.17725

[5] Jiang, S., Lu, D., Cai, B., 2019. GNSS NLOS Signal Modeling and Quantification Method in Railway Urban Canyon Environment. Proceedings of The 2019 IEEE Intelligent Vehicles Symposium (IV); June 9–12, 2019; Paris, France. pp. 1268–1273. DOI: https://doi.org/10.1109/IVS.2019.8814194

[6] Dong, D., Wang, M., Chen, W., et al., 2016. Mitigation of multipath effect in GNSS short baseline positioning by the multipath hemispherical map. Journal of Geodesy. 90, 255–262. DOI: https://doi.org/10.1007/s00190-015-0870-9

[7] Seferoglu, K.T., Turk, A.S., 2019. Review of Spoofing and Jamming Attack on the Global Navigation Systems Band and Countermeasure. Proceedings of The 2019 9th International Conference on Recent Advances in Space Technologies (RAST); June 11–14, 2019; Istanbul, Turkey. pp. 513–520. DOI: https://doi.org/10.1109/RAST.2019.8767871

[8] Broumandan, A., Siddakatte, R., Lachapelle, G., 2017. An approach to detect GNSS spoofing. IEEE Aerospace and Electronic Systems Magazine. 32(8), 64–75. DOI: https://doi.org/10.1109/MAES.2017.160190

[9] Zhuang, Q., Dong, D., Cui, K., et al., 2015. Detection and Compensation for Coded Odometer Errors Due to Train Wheel Slips and Slides. Proceedings of The Fifth International Conference on Transportation Engineering (ICTE 2015); September 26–27, 2015; Dalian, China. pp. 1640–1648. DOI: https://doi.org/10.1061/9780784479384.207

[10] Chen, Q., Niu, X., Zuo, L., et al., 2018. A railway track geometry measuring trolley system based on aided INS. Sensors. 18(2), 538. DOI: https://doi.org/10.3390/s18020538

[11] He, Y., Li, J., Liu, J., 2023. Research on GNSS INS & GNSS/INS integrated navigation method for autonomous vehicles: A survey. IEEE Access. 11, 79033–79055. DOI: https://doi.org/10.1109/ACCESS.2023.3299290

[12] Malakar, B., Roy, B.K., 2014. Survey of RFID Applications in Railway Industry. Proceedings of The 2014 First International Conference on Automation, Control, Energy and Systems (ACES); February 1–2, 2014; Adisaptagram, India. pp. 1–6. DOI: https://doi.org/10.1109/ACES.2014.6807999

[13] Dhahbi, S., Abbas-Turki, A., Hayat, S., et al., 2011. Study of the High-Speed Trains Positioning System: European Signaling System ERTMS/ETCS. Proceedings of The 2011 4th International Conference on Logistics; May 31–June 3, 2011; Hammamet, Tunisia. pp. 468–473. DOI: https://doi.org/10.1109/LOGISTIQUA.2011.5939444

[14] Fedorov, V.A., 2024. Recognizing Railway Infrastruc-

ture Using CNN and Stereoscopic Vision. Proceedings of The 2024 International Russian Smart Industry Conference (SmartIndustryCon); March 25–29, 2024; Sochi, Russia. pp. 13–18. DOI: https://doi.org/10.1109/SmartIndustryCon61328.2024.10516208
[15] Fedorov, V.A., 2024. Railway infrastructure detection based on YOLOv8 with npu acceleration. Doklady Mathematics. 110(Suppl 1), S42–S48. DOI: https://doi.org/10.1134/S1064562424601951
[16] Jegham, N., Koh, C.Y., Abdelatti, M., et al., 2024. Evaluating the evolution of yolo (you only look once) models: A comprehensive benchmark study of yolo11 and its predecessors. [Preprint]. DOI: https://doi.org/10.48550/arXiv.2411.00201
[17] Dewi, C., Juli Christanto, H., 2022. Combination of deep cross-stage partial network and spatial pyramid pooling for automatic hand detection. Big Data and Cognitive Computing. 6(3), 85. DOI: https://doi.org/10.3390/bdcc6030085
[18] Chen, X., Jiang, N., Yu, Z., et al., 2025. Citrus Leaf Disease Detection Based on Improved YOLO11 with C3K2. Proceedings of The International Conference on Computer Graphics, Artificial Intelligence, and Data Processing (ICCAID 2024); November 8–10, 2025; Guangzhou, China. pp. 746–751. DOI: https://doi.org/10.1109/CEI63587.2024.10871723
[19] He, K., Zhang, X., Ren, S., et al., 2015. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. 37(9), 1904–1916. https://doi.org/10.1109/TPAMI.2015.2389824
[20] Hidayatullah, P., Syakrani, N., Sholahuddin, M.R., et al., 2025. YOLOv8 to YOLO11: A comprehensive architecture in-depth comparative review. [Preprint]. DOI: https://doi.org/10.48550/arXiv.2501.13400
[21] Imambi, S., Prakash, K.B., Kanagachidambaresan, G.R., 2021. PyTorch. In: Prakash, K.B., Kanagachidambaresan, G.R. (eds.). Programming with TensorFlow: Solution for Edge Computing Applications. Springer, Cham, Switzerland. pp. 87–104. DOI: https://doi.org/10.1007/978-3-030-57077-4_10
[22] Jin, T., Bercea, G.T., Le, T.D., et al., 2020. Compiling ONNX neural network models using MLIR. [Preprint]. DOI: https://doi.org/10.48550/arXiv.2008.08272
[23] Ismagilov, R., 2023. Performance Evaluation of the Rockchip Systems-on-Chip Through YOLOv4 Object Detection Model. Proceedings of The 2023 IEEE Ural-Siberian Conference on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT); May 15–17, 2023; Yekaterinburg, Russia. pp. 241–243. DOI: https://doi.org/10.1109/USBEREIT58508.2023.10158842
[24] Uijlings, J.R., Van De Sande, K.E., Gevers, T., et al., 2013. Selective search for object recognition. International Journal of Computer Vision. 104, 154–171. DOI: https://doi.org/10.1007/s11263-013-0620-5
[25] Lin, T.Y., Maire, M., Belongie, S., et al., 2014. Microsoft Coco: Common Objects in Context. Proceedings of The Computer Vision–ECCV 2014: 13th European Conference; September 6–12, 2014; Zurich, Switzerland. pp. 740–755. DOI: https://doi.org/10.1007/978-3-319-10602-1_48