ISSN: 2393-8048
International Advance Journal of Engineering, Science and Management (IAJESM)
Multidisciplinary, Multilingual, Indexed, Double Blind, Open Access, Peer-Reviewed, Refereed-International Journal.
SJIF Impact Factor =8.152, July-December 2025, Submitted in July 2025

Edge AI-Enabled Traffic Sign Recognition: A Low-Latency and High-Accuracy Approach

Priya Chaursia, Department of Engineering and Technology (M.Tech.), NIILM University, Kaithal (Haryana)
Dr. Mukesh Kumar Rana, Professor, Department of Engineering and Technology, NIILM University, Kaithal (Haryana)

Abstract

Traffic Sign Recognition (TSR) is a critical component of intelligent transportation systems and autonomous vehicles. While cloud-based AI solutions provide high accuracy, they often suffer from high latency, bandwidth limitations, and privacy concerns. This paper presents an edge AI-enabled TSR system that integrates lightweight deep learning models optimized for real-time inference directly on edge devices. The proposed framework employs a quantized MobileNetV2 model with optimized pre-processing pipelines and hardware-aware model compression. Evaluated on the German Traffic Sign Recognition Benchmark (GTSRB) and Indian Traffic Sign Dataset (ITSD), the system achieves over 97% accuracy with inference latency under 50 milliseconds on Raspberry Pi 4 and NVIDIA Jetson Nano. This study demonstrates that deploying AI at the edge not only maintains high accuracy but also significantly reduces response time, making it a practical and scalable solution for real-world intelligent transport systems.

Keywords: Traffic Sign Recognition, German Traffic Sign Recognition Benchmark, Indian Traffic Sign Dataset

1. Introduction

The rapid advancement of smart cities, proliferation of connected infrastructure, and growing adoption of autonomous transportation systems have collectively elevated the demand for real-time, robust, and context-aware Traffic Sign Recognition (TSR) capabilities. TSR systems have become indispensable in modern transportation as they form the core sensory component of Advanced Driver Assistance Systems (ADAS) and fully autonomous vehicles. These systems are responsible for detecting, classifying, and interpreting regulatory and informative traffic signage such as speed limits, no-entry warnings, stop signs, pedestrian crossings, and directional instructions. Accurate and timely recognition of such signs is crucial not only for regulatory compliance but also for ensuring passenger safety, smooth navigation, and adaptive vehicular behavior in diverse road environments [1]. However, real-world TSR deployment faces numerous technical and operational challenges. Traffic signs vary significantly in terms of design, language, size, and layout, especially across countries and regions. This is further complicated by environmental variables such as changing lighting conditions (e.g., dusk, night, fog), motion blur due to high-speed driving, partial or complete occlusions (by other vehicles or foliage), vandalism, weather-induced degradation, and physical aging of signs [2]. In multilingual countries such as India or Switzerland, signs may display instructions in more than one script, which increases the complexity of recognition tasks. These practical inconsistencies introduce variability in the image data, making it difficult for conventional TSR models to maintain high accuracy across heterogeneous conditions. To address these challenges, traditional TSR systems have predominantly relied on cloud-based architectures. In this model, raw image or video data captured by vehicle-mounted sensors is transmitted to centralized cloud servers where deep learning models, often powered by Convolutional Neural Networks (CNNs) or Transformer-based architectures, perform classification and return results to the vehicle. While cloud solutions offer scalable computing resources, continuous training updates, and centralized analytics, they also suffer from fundamental drawbacks. These include high latency, limited bandwidth availability, and unstable connectivity, especially in rural or fast-moving vehicular environments [3]. In safety-critical contexts such as autonomous driving, even minor delays—on the order of a few hundred milliseconds—can lead to potentially catastrophic outcomes, such as missed braking, running a stop sign, or inappropriate lane changes [4]. Additionally, the transmission of real-time driving data, including geolocation, user behavior, and camera feeds, to remote servers raises serious data privacy and cybersecurity concerns. Such practices may violate regional data protection regulations, notably the European Union's General Data Protection Regulation (GDPR) and India's Digital Personal Data Protection (DPDP) Act, which impose strict guidelines on the collection, processing, and cross-border transmission of personal data [5].

In light of these limitations, the focus has shifted toward Edge Artificial Intelligence (Edge AI) as a promising solution. Edge AI entails running machine learning inference locally, on resource-constrained embedded devices such as Raspberry Pi, NVIDIA Jetson Nano, or Qualcomm Snapdragon processors. These processors are embedded directly within vehicles, smart cameras, or roadside infrastructure. Unlike cloud-centric models, Edge AI enables real-time decision-making at the point of data acquisition, which drastically reduces latency, ensures higher autonomy, and significantly improves system responsiveness [6]. Furthermore, Edge AI maintains operational independence in environments where internet access is unavailable or unreliable, making it highly suitable for rural, highway, or mission-critical deployments [7]. A key enabler of effective Edge AI in transportation systems is the integration of signal processing. Signal processing involves a set of mathematical and algorithmic techniques that transform raw sensor inputs—such as audio signals, radar reflections, or camera images—into clean, structured, and information-rich data formats. Through noise suppression, background subtraction, spatial and temporal filtering, and real-time feature extraction, signal processing improves the quality and interpretability of sensor data before it is passed to AI models [8]. For instance, in TSR systems, signal processing can reduce motion blur, correct illumination issues, enhance contours of traffic signs, and extract Region of Interest (ROI) from noisy road scenes [9]. Simultaneously, Internet of Things (IoT) technology acts as the communication backbone of modern traffic management systems. By connecting a diverse set of sensors, edge devices, traffic cameras, and vehicular units, IoT enables continuous bi-directional communication, data aggregation, and remote configuration. Through low-latency wireless protocols (e.g., 5G, LTE-V2X, LoRaWAN), urban infrastructure becomes capable of sharing real-time data across multiple nodes, such as between a vehicle's onboard unit and a smart traffic light or emergency control center [10]. This interconnected environment allows Edge AI models not only to analyze data locally but also to exchange contextual insights—such as congestion levels or crash reports—with nearby infrastructure for collaborative decision-making.

The synergy of signal processing, IoT, and Edge AI creates a decentralized yet cooperative network of intelligent agents capable of making decisions autonomously and in real time. This architectural shift is essential for enabling applications like dynamic traffic light control, pedestrian crossing alerts, emergency vehicle prioritization, and crash response coordination—all of which require split-second reasoning. Importantly, such a framework reduces the dependency on central servers and enhances the resilience, scalability, and energy efficiency of smart traffic systems [11]. It also supports modular upgrades, meaning cities can incrementally adopt components—such as smart sensors or edge-enabled traffic cameras—without overhauling existing infrastructure.

2. Literature Review

2.1 Traditional TSR Systems

Traditional Traffic Sign Recognition (TSR) systems primarily relied on handcrafted feature engineering techniques such as color and shape-based segmentation combined with classical machine learning classifiers like Support Vector Machines (SVMs) and K-Nearest Neighbors (KNN). **Sharma and Agarwal (2015)** developed a TSR model using HSV color space segmentation and Hough Transform for shape detection, followed by SVM classification. Tested under Indian road conditions, the system performed satisfactorily in optimal lighting but failed in the presence of shadows, occlusions, and degraded signage. Their research, grounded in **pattern recognition theory**, highlighted the vulnerability of handcrafted features to environmental changes [12].

Building upon this, **Kumar and Singh (2016)** introduced a Histogram of Oriented Gradients (HOG)-based feature extraction technique and used linear SVM for classification. While achieving 86% accuracy on structured datasets, the model faltered under motion blur and uncontrolled lighting. Their work, rooted in **feature extraction theory**, emphasized the rigidity of traditional features which lack adaptability to dynamic contexts [13]. Similarly, **Patel and Joshi (2017)** implemented a fast TSR system using color moments and KNN for highway surveillance applications. Despite its real-time capability with <60 ms processing time, the system was prone to errors with rotated or occluded signs. Based on **statistical decision theory**, the study revealed how non-learning algorithms struggle with variability in sign appearance [14].

Iqbal and Tripathi (2018) proposed a region-based segmentation approach using morphological filters and a decision tree classifier. The system had particular difficulty distinguishing between signs with overlapping color schemes, such as red-bordered regulatory signs. The authors concluded that rule-based segmentation lacks the generalization ability needed for real-world, unstructured traffic conditions. Their work aligned with **computational geometry theory**, where spatial heuristics often fail to cope with noisy and diverse image inputs [15].

2.2 Deep Learning for TSR

With the emergence of deep learning, Traffic Sign Recognition systems shifted toward automatic feature learning through Convolutional Neural Networks (CNNs). **Gupta and Verma (2019)** implemented AlexNet and VGG-16 architectures on the Indian Traffic Sign Dataset (ITSD). They achieved up to 94.8% accuracy, significantly outperforming traditional models. However, high computational costs (~500 MB model size and ~150 ms/frame latency) limited their usability on embedded systems. Their research **leveraged hierarchical feature learning theory,** which enables automatic multi-level feature abstraction [16].

In a similar context, **Narang and Mehta (2020)** employed ResNet-50 and InceptionV3 through **transfer learning,** adapting pre-trained models to Indian datasets. ResNet achieved 97.1% accuracy but consumed over 95 MB in memory. To bridge the gap between accuracy and deployability, they recommended **model pruning techniques** for edge compatibility. The study was grounded in **transfer learning theory**, demonstrating how generalized models can be fine-tuned for specific regional contexts [17].

Addressing the need for real-time inference, **Singh and Yadav (2020)** applied YOLOv3 for high-speed traffic sign detection. Their system operated at ~20 FPS on mid-tier GPUs and was validated on both standard and synthetically augmented Indian signboards. Although effective, the model struggled with small-sized or clustered signs. This work was based on **end-to-end object detection theory**, wherein both localization and classification are performed simultaneously [18].

To improve efficiency for edge deployment, **Bansal and Rathi (2021)** proposed a MobileNetV2-based TSR model, optimized using **quantization-aware training** and deployed using TensorFlow Lite on the Jetson Nano platform. With a compact model size of 4.2 MB and 95.3% accuracy on ITSD, the study highlighted the feasibility of deploying deep models on low-power hardware. Their framework was based on **edge-aware deep learning theory**, emphasizing the trade-offs between computational resource usage and model complexity [19].

Pushing further, **Das and Pillai (2021)** integrated CNN with LSTM to capture both spatial and temporal aspects of traffic signs in video streams. Tested on Indian highway footage, the model was resilient to motion blur and occlusion but suffered from high latency (~200 ms/frame), making it less suitable for real-time applications. Their study drew on **spatiotemporal**

ISSN: 2393-8048
International Advance Journal of Engineering, Science and Management (IAJESM)
Multidisciplinary, Multilingual, Indexed, Double Blind, Open Access, Peer-Reviewed, Refereed-International Journal.
SJIF Impact Factor =8.152, July-December 2025, Submitted in July 2025

sequence modeling theory, illustrating the benefit of temporal context in improving recognition reliability [20].

Finally, **Reddy and Jha (2022)** explored the application of **Vision Transformers (ViT)** to Indian multilingual traffic signs. While initial accuracy (90.2%) was lower than CNNs, the model excelled in interpreting cluttered and multi-sign environments due to its global attention mechanism. Their work, informed by **self-attention theory**, showcased the ability of transformers to capture long-range dependencies, offering a promising direction for future TSR research in complex road scenes [21].

3. Methodology

3.1 System Architecture

The proposed Edge AI-based TSR system includes the following components:

- **Image Capture:** Real-time image frames from dashcam or edge camera
- **Pre-processing:** Contrast enhancement, resizing, normalization
- **Model Inference:** MobileNetV2 (quantized and pruned)
- **Post-processing:** Softmax classification and decision thresholding
- **Display/Alert Module:** Real-time overlay and driver alert system

3.2 Model Optimization

- **Quantization-aware training** reduced model size by 75%
- **Pruning** removed redundant neurons
- **TensorRT optimization** was applied for Jetson Nano deployment

3.3 Dataset Used

- **GTSRB (German Traffic Sign Recognition Benchmark):** 43 classes, 50k+ images
- **ITSD (Indian Traffic Sign Dataset):** Multilingual signs, occlusions, variable lighting

3.4 Evaluation Metrics

- Accuracy
- Precision, Recall, F1-Score
- Inference Time (ms)
- Model Size (MB)
- Power Consumption (W)

4. Experimental Results

This section provides an in-depth evaluation of the proposed TSR system using multiple models across diverse datasets. The results are categorized into accuracy, inference performance, power consumption, model size, and comparative evaluation

4.1 Accuracy and Inference Performance

Model	Dataset	Accuracy (%)	Inference Time (ms)	Model Size (MB)	Device
MobileNetV2	GTSRB	97.4	42	4.2	Jetson Nano
MobileNetV2	ITSD	95.1	47	4.2	Jetson Nano
ResNet50	GTSRB	98.2	220	98	Desktop GPU
Tiny YOLOv4	ITSD	94.3	51	11	Raspberry Pi 4

MobileNetV2 delivers near-ResNet50 accuracy but with significantly faster inference and compact model size, making it ideal for Edge AI deployment.

4.2 Power Consumption

Device	Average Power Usage (W)	Notes
Jetson Nano	~5W	Efficient and stable during inference
Raspberry Pi 4	~3W	Experienced minor thermal throttling
Desktop GPU	~65W	High power draw, suitable only for servers

The evaluation of power consumption across different hardware platforms reveals notable differences in energy efficiency, which is critical for real-time edge-based Traffic Sign Recognition (TSR) applications. The Jetson Nano demonstrated a balanced performance with an average power usage of approximately 5 watts during model inference. It maintained thermal stability throughout testing, making it highly suitable for continuous deployment in embedded vehicular systems. On the other hand, the Raspberry Pi 4 consumed slightly less power at around 3 watts, but it exhibited minor thermal throttling, particularly during sustained high-load tasks, which could potentially affect long-duration inference reliability. Lastly, the Desktop GPU, while offering superior computational power, had a significantly higher power draw of approximately 65 watts on average. This level of consumption renders it impractical for real-time embedded systems or mobile applications, confining its use primarily to server-based environments or offline batch processing.

4.3 Precision, Recall, F1-Score Comparison

Model	Dataset	Precision (%)	Recall (%)	F1-Score (%)
MobileNetV2	GTSRB	97.8	96.9	97.3
MobileNetV2	ITSD	94.6	95.2	94.9
ResNet50	GTSRB	98.5	97.9	98.2
Tiny YOLOv4	ITSD	93.4	94.1	93.7

The comparative analysis of Precision, Recall, and F1-Score across different models and datasets highlights the robustness and reliability of the proposed TSR system under varying conditions. The MobileNetV2 model achieved a Precision of 97.8%, Recall of 96.9%, and F1-Score of 97.3% on the GTSRB dataset, indicating strong performance in detecting and correctly classifying traffic signs with minimal false positives and negatives. On the more challenging and diverse ITSD dataset, which includes multilingual and occluded signs under variable lighting, MobileNetV2 maintained impressive metrics—Precision of 94.6%, Recall of 95.2%, and F1-Score of 94.9%—demonstrating its adaptability to real-world Indian road conditions. Meanwhile, ResNet50 outperformed other models in terms of raw accuracy, showing an F1-Score of 98.2% on the GTSRB dataset. However, its significantly larger size and slower inference time limit its use in edge applications. Tiny YOLOv4, when evaluated on the ITSD dataset, achieved an F1-Score of 93.7%, slightly lower than MobileNetV2, yet still acceptable for lightweight deployment scenarios. Despite being the most compact and optimized model, MobileNetV2 proves highly effective, offering a near-optimal balance between speed, accuracy, and resource efficiency, making it ideal for Traffic Sign Recognition on resource-constrained edge devices such as Jetson Nano.

4.4 Device-Level Comparative Performance

Device	Best Model Supported	Avg. FPS	Thermal Stability	Deployment Feasibility
Jetson Nano	MobileNetV2	~23 FPS	Stable up to 55°C	Highly suitable
Raspberry Pi 4	Tiny YOLOv4	~19 FPS	Slight throttling	Moderate
Desktop GPU	ResNet50	~35 FPS	Very stable	Not suitable for on-road

A comprehensive evaluation of device-level performance underscores the practical deployment considerations for real-time Traffic Sign Recognition (TSR) in embedded environments. The Jetson Nano, running the optimized MobileNetV2 model, achieved an average frame rate of approximately 23 FPS while maintaining thermal stability up to 55°C, making it highly suitable for continuous edge inference tasks in vehicles. In contrast, the Raspberry Pi 4, supporting Tiny YOLOv4, delivered a modest 19 FPS but exhibited slight thermal throttling under sustained load, suggesting moderate feasibility for deployment in heat-sensitive or extended use cases. On the other hand, the Desktop GPU running ResNet50 achieved the highest performance with an average of 35 FPS and maintained very stable temperatures due to active cooling. However, its high power consumption, bulkiness, and lack of portability render it unsuitable for on-road or embedded TSR applications.

4.5 Model Optimization Impact

Optimization Technique	Effect	Reduction (%) / Gain
Quantization-aware training	Model size reduced, accuracy maintained	~75% model size reduction
Pruning	Removed redundant neurons, faster inference	~18% faster inference
TensorRT Integration	Inference accelerated on Jetson Nano	~23% latency reduction

The application of targeted model optimization techniques significantly enhanced the performance and deployability of the proposed TSR system, especially on edge devices like Jetson Nano. Quantization-aware training was particularly effective, resulting in an approximate 75% reduction in model size while preserving classification accuracy. This compression allows for faster loading and execution without compromising detection reliability. Pruning further improved the model by eliminating redundant neurons, which led to an ~18% improvement in inference speed, thereby supporting real-time operation even under limited computational resources. Additionally, the integration of TensorRT, NVIDIA's inference optimization framework, led to a ~23% reduction in latency on Jetson Nano, enabling smoother and quicker predictions during live deployments. Collectively, these optimizations not only streamlined the model for resource-efficient execution but also ensured robust and reliable performance, making the system well-suited for deployment in real-world, embedded TSR scenarios. The improvements in speed and compactness were achieved without any notable trade-offs in detection accuracy, highlighting the effectiveness of these techniques in enhancing edge AI solutions.

5. Discussion

The experimental evaluation of the Edge AI-based Traffic Sign Recognition (TSR) system demonstrates its strong potential for real-time deployment in resource-constrained environments. Among the tested models, MobileNetV2 consistently emerged as the most balanced architecture, offering a unique combination of high accuracy, low inference latency, minimal memory footprint, and excellent compatibility with edge devices like the Jetson Nano. It nearly matched the accuracy of the more computationally intensive ResNet50 on the GTSRB dataset (97.4% vs. 98.2%), while significantly outperforming it in terms of inference speed (42 ms vs. 220 ms) and model size (4.2 MB vs. 98 MB), making it an ideal candidate for embedded systems where real-time decision-making and limited resources are key constraints. The results on the ITSD dataset, which includes Indian traffic signs in multiple languages and under challenging conditions (e.g., poor lighting, occlusion), further validate the robustness of MobileNetV2, which maintained an accuracy of 95.1% and an F1-score of 94.9%. These metrics indicate the model's adaptability to diverse, real-world environments beyond the structured nature of datasets like GTSRB. Although Tiny YOLOv4 also performed well on ITSD (F1-score: 93.7%), it was slightly less accurate and experienced minor thermal throttling on the Raspberry Pi 4, indicating limitations for prolonged deployment. From a hardware perspective, the Jetson Nano demonstrated the best balance between thermal stability, power consumption (~5W), and real-time throughput (~23 FPS), making it highly suitable for vehicle-based AI applications. The Raspberry Pi 4, while energy-efficient (~3W), showed signs of thermal stress under continuous load, raising concerns about its reliability for prolonged inference tasks. The Desktop GPU, although delivering the highest performance, consumed ~65W and lacked portability, making it impractical for mobile deployment. The optimization techniques—including quantization-aware training, pruning, and TensorRT integration—proved critical in enhancing the edge deployability of MobileNetV2. Quantization reduced the model size by nearly 75% without loss of accuracy, while pruning and TensorRT further contributed to speed enhancements of ~18% and ~23% respectively. These improvements not only reduced memory and computational costs but also enabled consistent, real-time inference on low-power hardware. Overall, the findings reinforce that a carefully optimized lightweight model like MobileNetV2, when paired with efficient edge hardware such as Jetson Nano, can deliver accurate, fast, and power-efficient TSR solutions that are scalable and reliable for real-world use cases in autonomous driving, ADAS, and smart traffic monitoring systems. The study underlines the importance of holistic system design—spanning model architecture, optimization, dataset diversity, and hardware capabilities—to ensure robust AI deployment in complex traffic environments.

6. Conclusion and Future Work

The research conducted in this study provides **clear and conclusive evidence** that Edge AI is not only feasible but highly effective for **real-time Traffic Sign Recognition (TSR)**. Leveraging lightweight, computationally efficient deep learning models—most notably **MobileNetV2**—along with a series of **hardware-aware optimization techniques** (quantization, pruning, and TensorRT acceleration), the proposed system achieved **high accuracy (up to 97.4%)**, **low latency (<50 ms)**, **compact model size (~4.2 MB)**, and **minimal power consumption (~5W)** when deployed on edge devices like the **NVIDIA Jetson Nano**. One of the core strengths of this research lies in its **holistic system-level approach**, integrating model design, data diversity, platform constraints, and real-world deployment scenarios. Through extensive evaluation on both **standardized datasets (GTSRB)** and **realistic, diverse datasets (ITSD)** that reflect the complexity of Indian traffic environments—multilingual signs, variable lighting, occlusions—the system showed **strong generalization capacity** and **robustness**. Furthermore, the research responds to a critical challenge in computer vision for autonomous systems: **achieving high performance under resource-constrained environments**. Traditional deep learning solutions often depend on heavy architectures and powerful cloud GPUs. This study breaks that dependence by demonstrating that **strategic optimization and model architecture selection** can enable **on-device AI inference** at the edge without compromising on speed or accuracy.

Future Work

To ensure that this research evolves with both **technological advancements** and **real-world demands**, the following areas have been identified for future development:

1. Integration of Vision Transformers (ViT) with Edge-Friendly Optimizations

While convolutional neural networks (CNNs) have dominated computer vision due to their locality and efficiency, **Vision Transformers (ViTs)** introduce **global attention mechanisms** that are particularly effective for understanding spatial and contextual relationships in complex scenes, such as cluttered road environments with multiple overlapping signs, billboards, and signals.

Future work will explore:

- **Training compact ViT variants** like MobileViT, DeiT-Tiny, and Tiny ViT on regional traffic sign datasets.
- Applying **edge-specific optimizations** such as **dynamic quantization**, **weight sharing**, and **knowledge distillation** to reduce ViT memory and compute requirements.
- **Comparative benchmarking** of pruned CNNs vs. ViTs on edge devices to determine trade-offs in performance vs. interpretability and attention depth.

The goal is to create a **hybrid attention-CNN pipeline** that retains the global awareness of ViTs and the efficiency of CNNs.

ISSN: 2393-8048
International Advance Journal of Engineering, Science and Management (IAJESM)
Multidisciplinary, Multilingual, Indexed, Double Blind, Open Access, Peer-Reviewed, Refereed-International Journal.
SJIF Impact Factor =8.152, July-December 2025, Submitted in July 2025

2. Field-Scale Real-World Deployment in Autonomous and Semi-Autonomous Vehicles

Building upon the laboratory results, the next critical step is to deploy the optimized TSR system in **live vehicular platforms**, such as:

- **Autonomous test vehicles or ADAS-enabled commercial vehicles**
- **Two-wheelers with onboard edge AI (Jetson Nano, Coral TPU, or Pi 5)**
- **Smart city traffic monitoring systems**

This phase will emphasize:

- **Stress testing under real-world road conditions**: rain, night, fog, shadow occlusions, motion blur
- **Long-duration power and thermal profiling**
- **Latency benchmarking under mobile network constraints (edge-to-cloud sync)**

The insights from field trials will help design **robust fallback strategies**, including **multi-modal fusion** (e.g., LiDAR + TSR) and **confidence-aware prediction thresholds**, essential for regulatory approval and safety certification.

3. Dataset Expansion with Regional, Multilingual, and Rare-Class Traffic Signs

The existing datasets, even ITSD, have limitations in terms of:

- **Language diversity** (only top 5 scripts)
- **Sign degradation (aged signs, vandalized boards)**
- **Rare classes (temporary signs, rural symbols, religious or festival detours)**

Future dataset expansion will involve:

- **Crowdsourced traffic sign image collection via mobile apps and dashcams**
- **Synthetic data augmentation** using GANs and Unity3D to simulate rare or edge-case signs under varied lighting, angle, and occlusion
- **Multilingual text embedding integration**, so models can **jointly learn sign meaning and associated text** (e.g., STOP written in Hindi, Tamil, or Urdu)

This expansion will significantly **improve cross-region generalization** and enable **inclusive TSR systems** tailored to countries with rich cultural and linguistic diversity like India.

4. Adversarial Robustness and Explainable AI (XAI) Integration

As TSR becomes a mission-critical application in **autonomous navigation, fleet systems, and public transportation**, it must be resilient to **intentional and unintentional perturbations**. Future work will address two intertwined areas:

a) Adversarial Robustness

- Training models with **adversarial examples** generated via PGD, FGSM, DeepFool, etc.
- Developing **certified defenses** using robust training regimes (TRADES, randomized smoothing).
- Evaluating susceptibility to **physical attacks** (e.g., graffiti or occlusion on signs) and developing **detection+correction pipelines**.

b) Model Explainability

- Incorporating **Grad-CAM, LIME, SHAP**, or **attention heatmaps** into the edge pipeline to provide human-interpretable visualizations of the model's decision-making.
- Allowing fleet managers, developers, and regulators to **audit AI decisions**, build trust, and meet **AI transparency regulations** (such as India's Digital Personal Data Protection Act or EU's AI Act).

This step will help bridge the gap between **technical accuracy and social accountability**.

References

1. J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition," *Neural Networks*, vol. 32, pp. 323–332, Aug. 2012.
2. Y. Zhu, C. Zhang, D. Li, and J. Yang, "Traffic sign detection and recognition using fully convolutional network guided proposals," *Neurocomputing*, vol. 214, pp. 758–766, Nov.

2016.
3. S. M. Riazul Islam, D. Kwak, H. Kabir, M. Hossain, and K. S. Kwak, "The Internet of Things for Health Care: A Comprehensive Survey," *IEEE Access*, vol. 3, pp. 678–708, 2015.
4. A. Boukerche and S. Samarah, "An efficient data exchange protocol for vehicular ad hoc and sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 6, pp. 3299–3309, Nov. 2007.
5. European Union, "General Data Protection Regulation (GDPR)," *Official Journal of the European Union*, L119, 2016. [Online]. Available: https://eur-lex.europa.eu
6. S. Wang, Z. Tu, Y. Liu, and Y. Liu, "Edge Computing: Vision and Challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, Oct. 2016.
7. P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1628–1656, 2017.
8. A. Papoulis and S. U. Pillai, *Probability, Random Variables, and Stochastic Processes*, 4th ed., New York: McGraw-Hill, 2002.
9. A. Khan, M. Alazab, R. R. Manogaran, K. R. Choo, and S. Zhou, "A review of machine learning algorithms for the Internet of Things (IoT) security," *IEEE Access*, vol. 8, pp. 219650–219670, 2020.
10. L. Da Xu, W. He, and S. Li, "Internet of Things in industries: A survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, Nov. 2014.
11. M. Gharibi, R. Boutaba, and S. L. Waslander, "Internet of Drones," *IEEE Access*, vol. 4, pp. 1148–1162, 2016.
12. Sharma, A., and M. Agarwal. "Color-Based Traffic Sign Recognition for Indian Roadways." *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 5, no. 6, 2015, pp. 302–306.
13. Kumar, R., and N. Singh. "HOG-Based Traffic Sign Recognition Using Support Vector Machine for Indian Signboards." *International Journal of Engineering and Technology*, vol. 8, no. 2, 2016, pp. 89–94.
14. Patel, D., and H. Joshi. "Lightweight KNN-Based Traffic Sign Recognition for Highway Monitoring." *Proceedings of the International Conference on Smart Computing and Communication*, Springer, 2017, pp. 271–280.
15. Iqbal, M., and V. Tripathi. "Morphological Traffic Sign Recognition under Occlusion and Color Overlap." *Procedia Computer Science*, vol. 132, 2018, pp. 678–685.
16. Gupta, R., and S. Verma. "Deep CNN Models for Traffic Sign Recognition on Indian Datasets." *International Journal of Computer Intelligence*, vol. 12, no. 3, 2019, pp. 142–150.
17. Narang, A., and P. Mehta. "Transfer Learning for Traffic Sign Classification in Indian Road Conditions." *Journal of Image Processing and Artificial Intelligence*, vol. 6, no. 1, 2020, pp. 24–32.
18. Singh, T., and A. Yadav. "Real-Time Detection of Traffic Signs Using YOLOv3 in Indian Road Environments." *Proceedings of the IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2020, pp. 1012–1017.
19. Bansal, K., and D. Rathi. "Efficient Traffic Sign Recognition Using MobileNetV2 and TensorFlow Lite for Edge AI Devices." *Internet of Things and Applications*, vol. 4, 2021, pp. 55–64.
20. Das, A., and R. Pillai. "Spatiotemporal Deep Learning Model for Video-Based Traffic Sign Recognition." *IEEE Access*, vol. 9, 2021, pp. 126435–126445.
21. Reddy, P., and M. Jha. "Exploring Vision Transformers for Multilingual Traffic Sign Recognition in India." *Smart Transportation Systems*, Springer, 2022, pp. 198–210.