

Smart Surveillance of Harmful Highway Vegetation Using Deep Learning

Nitturu Asha Jyothi¹, Janapati Venkata Sriveda Manaswi², Janapati Venkata Sai Chandra Koustubh³,
Vartha Venu⁴, Dasareddygaru Jaswanth Reddy⁵

¹*Department of Artificial Intelligence and Data Science*

^{2,3,4,5}*Department of Computer Science and Engineering, Gitam University,
Bangalore Campus, India – 561203*

Abstract—The rapid spread of harmful and invasive roadside vegetation poses significant risks to highway infrastructure, road safety, and ecological balance. Traditional vegetation monitoring methods rely on manual inspection, remote sensing, or citizen-reported observations, which often suffer from limited coverage, high operational costs, and delayed response times. This study presents a Smart Surveillance System for Harmful Highway Vegetation using deep learning and edge computing to enable automated, geotagged detection from roadside video streams. The proposed system integrates a transformer-based deep learning model with a CPU-optimized edge inference pipeline, GPS-based geolocation tagging, and a web-based monitoring platform that separates administrative control from field-level access. By combining computer vision with spatial mapping, the system aims to provide scalable and efficient vegetation monitoring along transport corridors.

The methodology involves frame sampling from vehicle-mounted video capture, image tiling for high-resolution processing, deep learning-based binary classification, and threshold-based detection logging. Each detection is associated with geographic coordinates and visualized through an interactive web interface using geospatial mapping tools. Experimental evaluation conducted on Indian highway footage produced stable detection outputs with consistent confidence scores and reliable GPS tagging. The results demonstrate the feasibility of deploying a lightweight, edge-based vegetation surveillance framework capable of near real-time monitoring without GPU dependency. The proposed system contributes toward intelligent infrastructure management and supports future expansion into multi-class vegetation identification and large-scale deployment across transportation networks.

Index Terms—Highway Vegetation Monitoring, Invasive Plant Detection, Deep Learning, Edge

Computing, Roadside Surveillance, Geospatial Mapping, Computer Vision, Transformer-Based Models, GPS Geotagging.

I. INTRODUCTION

The rapid expansion of highway networks has significantly increased ecological disturbances along transportation corridors, creating favorable conditions for the spread of harmful and invasive vegetation. Invasive plant species are recognized as major contributors to biodiversity loss, ecosystem imbalance, and substantial economic damage worldwide. Highways function as critical dispersal pathways due to continuous vehicular movement, soil disruption, and altered environmental conditions. Early identification and continuous monitoring of harmful roadside vegetation are therefore essential to mitigate ecological degradation and ensure infrastructure safety.

Conventional monitoring approaches such as manual field surveys, satellite-based remote sensing, and citizen science initiatives provide valuable information but suffer from practical limitations. Manual surveys are labor-intensive and unsuitable for large-scale monitoring. Remote sensing techniques require high-resolution imagery and often lack the capability to reliably distinguish specific vegetation types at ground level. Citizen science data may introduce spatial and taxonomic biases, affecting reliability. These constraints emphasize the necessity for automated, scalable, and cost-effective surveillance systems capable of operating in dynamic roadside environments.

Recent advances in deep learning have demonstrated remarkable performance in image classification and object detection tasks across domains such as medical imaging, traffic monitoring, and environmental analytics. Convolutional Neural Networks (CNNs) and transformer-based architectures have shown strong capability in extracting hierarchical spatial features from complex visual data. Applications of deep learning in image-based detection systems, including infrastructure monitoring and environmental assessment, have proven its robustness in handling real-world variability [10], [11]. Such models are capable of learning discriminative features from high-resolution images and generalizing across varied backgrounds and illumination conditions.

In roadside vegetation monitoring, vehicle-mounted camera systems combined with deep learning-based inference offer a promising solution for large-scale detection. By processing video frames in real time or near real time, harmful vegetation can be identified without manual inspection. However, challenges remain in ensuring computational efficiency, especially when deploying models on resource-constrained devices without GPU acceleration. Edge computing frameworks address this issue by enabling optimized CPU-based inference while maintaining acceptable processing speed and reliability.

In this work, a Smart Surveillance of Harmful Highway Vegetation system is proposed, integrating deep learning-based detection, edge computing, GPS-based geotagging, and web-based visualization. The system processes roadside video frames using a tiling and preprocessing strategy, performs classification using a pre-trained transformer-based model, and logs geospatial detections in structured format. A backend server manages survey data, while an interactive web interface enables visualization of detection points on geospatial maps.

By combining computer vision, edge inference, and geospatial monitoring within a unified framework, the proposed system provides a scalable and practical approach for automated highway vegetation surveillance. This integration supports environmental authorities and infrastructure agencies in making

informed decisions for timely vegetation control and ecological preservation.

II. LITERATURE REVIEW

Over the past decade, significant advancements have been made in vegetation detection and land cover classification through the integration of remote sensing techniques, spectral indices, and deep learning-based semantic segmentation models. Early vegetation monitoring systems primarily relied on spectral reflectance properties, particularly the contrast between red and near-infrared (NIR) bands, to distinguish chlorophyll-rich vegetation from surrounding surfaces. The Normalized Difference Vegetation Index (NDVI), introduced in early satellite-based vegetation studies, became a foundational metric for large-scale vegetation assessment [14]–[16]. NDVI-based approaches demonstrated strong performance in agricultural monitoring, forest mapping, and environmental analysis. However, threshold-based NDVI classification often suffers from sensitivity to illumination variation, shadow artifacts, soil background interference, and fixed parameter dependency, limiting its robustness in heterogeneous environments.

To improve traditional spectral methods, researchers incorporated object-based image analysis (OBIA), morphological filtering, and machine learning classifiers. In a notable study, Ayhan et al. [4] presented a comprehensive evaluation of vegetation detection using both deep learning and conventional approaches. Their work compared three frameworks: DeepLabV3+, a custom CNN model, and an NDVI-based machine learning (NDVI-ML) object-oriented approach. The dataset consisted of high-resolution UAV imagery (10 cm and 20 cm spatial resolution) including RGB and NIR bands. Experimental results showed that the NDVI-ML approach achieved 87.30% accuracy on Kimisala-10 imagery and 85.78% on Kimisala-20 imagery, outperforming the deep learning models under limited training conditions. The authors attributed this to domain shift between training and testing datasets and the limited availability of multi-resolution training data. Their findings highlighted that while deep learning methods offer end-to-end learning capability,

spectral-domain methods remain competitive in scenarios with constrained datasets.

With the evolution of convolutional neural networks (CNNs), vegetation detection shifted toward pixel-wise semantic segmentation. Chen et al. introduced DeepLab and later DeepLabV3+, an encoder–decoder architecture employing Atrous Spatial Pyramid Pooling (ASPP) to capture multi-scale contextual information [25], [42]. DeepLabV3+ demonstrated state-of-the-art performance on benchmark datasets such as PASCAL VOC 2012 [26]. Its ability to combine high-level semantic features with fine spatial resolution made it particularly suitable for land cover segmentation tasks. Subsequent studies applied DeepLabV3+ to remote sensing imagery for crop mapping and vegetation classification [30]–[33]. Du et al. [30] reported that DeepLabV3+ outperformed UNet, PSPNet, and SegNet in crop area mapping using RGB imagery. However, the model’s dependency on large annotated datasets and its restriction to three-channel RGB inputs limit its flexibility when multi-spectral data such as NIR are available.

Several researchers attempted to extend DeepLabV3+ to incorporate additional spectral channels. Huang et al. [35] modified the backbone architecture to process multi-channel inputs by replacing the Xception backbone with ResNet101. Although performance improved with additional features such as point clouds and NIR information, the approach required training from scratch due to incompatibility with RGB pre-trained weights. Transfer learning from RGB to multi-band imagery remains challenging because pre-trained models are predominantly trained on natural image datasets without NIR information [41]. These limitations highlight a computational and architectural bottleneck in adapting high-performing segmentation networks for multi-spectral remote sensing tasks.

In parallel, custom CNN architectures were explored to overcome multi-band limitations. Perez et al. [38] demonstrated the feasibility of CNN-based soil detection using patch-based classification, highlighting that smaller architectures can effectively learn spectral–spatial relationships with fewer parameters. Similarly, Kwan et al. [40] showed that deep learning models can achieve reasonable land cover classification accuracy using limited spectral bands. However, these models often require

sequential training due to memory constraints and may not generalize well across datasets with differing resolutions and imaging conditions.

Beyond semantic segmentation, object detection frameworks were also explored for vegetation analysis. Yamamoto et al. proposed a multi-modal Faster R-CNN architecture that fused RGB images with vegetation indices such as Excess Green (ExG) and CIVE to improve roadside vegetation detection [3]. Their fusion strategy enhanced robustness under varying illumination conditions, demonstrating that integrating handcrafted vegetation indices with deep learning features can improve detection stability. However, Faster R-CNN–based frameworks require GPU acceleration and are computationally intensive, limiting their applicability for edge-based or real-time deployment.

Recent advancements in deep learning introduced attention mechanisms and transformer-based models to capture global contextual relationships. Attention-based feature fusion networks [32] and patch-attention segmentation models [33] have shown improved performance in remote sensing classification tasks. These architectures address limitations of convolutional receptive fields by modeling long-range dependencies. However, transformer-based models significantly increase computational complexity and memory consumption, making them less suitable for deployment on CPU-only embedded systems without aggressive optimization.

Another key challenge identified in the literature is dataset imbalance and domain shift. Ayhan and Kwan [50] discussed how imbalanced class distributions bias the loss function toward dominant classes, degrading minority class performance. Additionally, Zhang et al. [20] emphasized that deep learning models often exhibit significant performance degradation when training and testing data differ in resolution, sensor characteristics, or environmental conditions. This domain sensitivity is particularly relevant in roadside vegetation monitoring, where lighting, weather, and camera positioning vary continuously.

Despite the impressive progress in vegetation segmentation accuracy, most existing studies focus on offline UAV or satellite imagery processing rather than real-time video streams. Furthermore, computational requirements remain high, with most

semantic segmentation frameworks relying on GPU acceleration for inference. Few studies address CPU-optimized inference, thread tuning, batch processing strategies, or memory-efficient logging mechanisms required for edge deployment.

From the reviewed literature, several methodological gaps are evident. First, many high-accuracy vegetation detection methods depend on NIR imagery, limiting deployment in low-cost RGB-only systems. Second, deep learning models demonstrate sensitivity to dataset distribution shifts and require extensive annotated training data. Third, integration between computer vision inference and geospatial logging is rarely addressed in vegetation monitoring research. Fourth, limited work explores real-time, CPU-based inference pipelines suitable for roadside monitoring under constrained hardware environments.

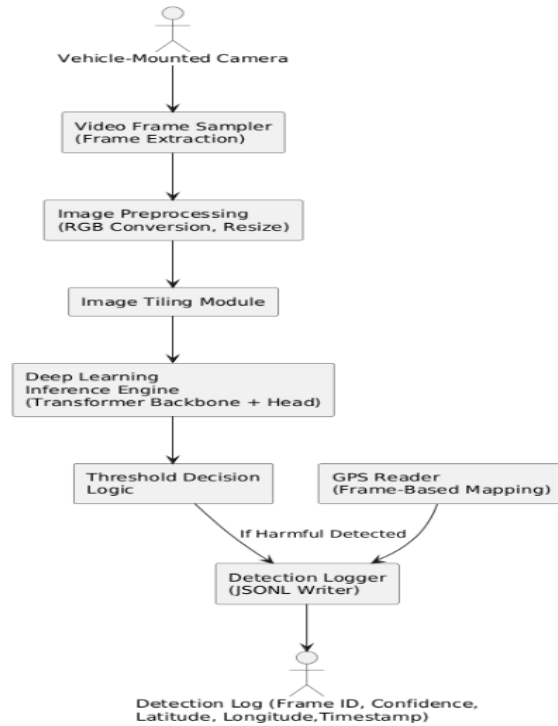
Overall, the literature shows a clear progression from spectral index thresholding to deep learning-based semantic segmentation and multi-modal fusion techniques. While architectures such as DeepLabV3+ [25] and CNN-based frameworks [38] have improved segmentation accuracy, challenges related to computational efficiency, multi-spectral integration, and deployment scalability remain unresolved. The need for a resource-efficient, edge-compatible vegetation detection framework that integrates geospatial tagging and stable real-time inference represents a critical research gap. The present study builds upon these findings by proposing an optimized edge-based inference pipeline tailored for roadside harmful vegetation detection, addressing both algorithmic performance and deployment feasibility.

III. METHODOLOGY

This study proposes a complete edge-to-cloud intelligent surveillance framework for the detection and geospatial monitoring of harmful highway vegetation using deep learning. The methodology integrates computer vision-based inference, GPS-based geotagging, structured logging, backend data management, and web-based geospatial visualization into a unified operational pipeline.

The primary objective is to design a computationally efficient, CPU-optimized system capable of performing real-time or near real-time roadside vegetation detection under practical deployment

constraints such as limited hardware resources and long-duration field operation. The architectural design ensures that data flows seamlessly from roadside video capture to administrative geospatial visualization without manual intervention, thereby enabling scalable and structured highway monitoring.



The overall system architecture consists of five interconnected components: a roadside data acquisition module, an edge-based inference pipeline, a geospatial logging framework, a backend server and database management layer, and a web-based monitoring interface. Each component is carefully designed to ensure reliability, computational stability, and end-to-end traceability of detection events.

A. Roadside Data Acquisition

The first stage of the methodology involves acquiring roadside video data using a vehicle-mounted RGB camera system. The camera is positioned strategically to capture continuous footage of highway vegetation along transport corridors. Since harmful or invasive vegetation can appear as small clusters distributed irregularly along roadside segments, the system records video at sufficiently high resolution to preserve fine-grained visual details.



However, processing every frame of high-resolution video introduces significant computational overhead. To balance detection sensitivity with computational feasibility, a controlled frame sampling mechanism is implemented. Let the recorded video stream be represented as:

$$V = \{F_1, F_2, F_3, \dots, F_n\}$$

Instead of processing all frames, a configurable sampling rate (r) (frames per second) is applied such that the processed frame set becomes:

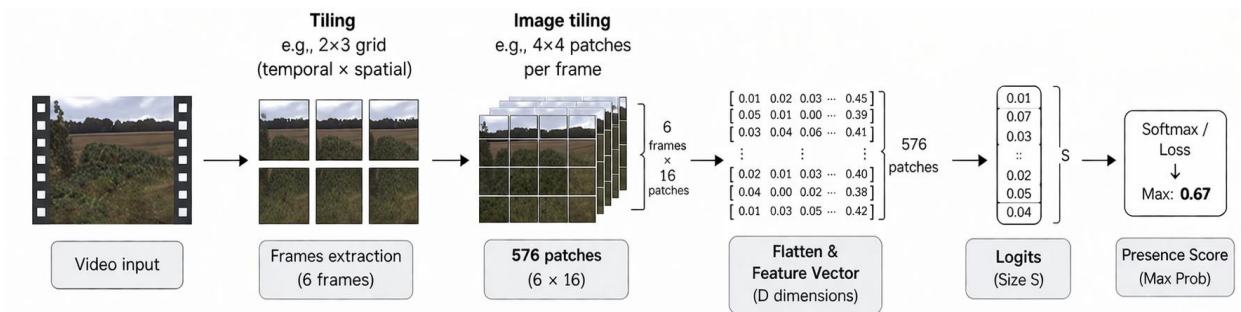
$$F_{\text{sampled}} = \{F_i \mid i = k \times r\}$$

where (k) is a positive integer. This sampling strategy reduces redundant processing while maintaining sufficient spatial continuity along the surveyed highway stretch.

B. Edge-Based Inference Pipeline

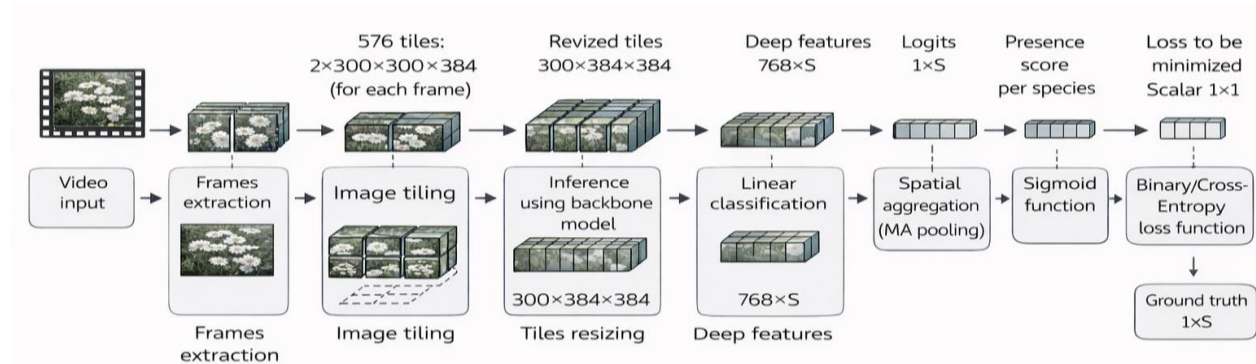
The core of the methodology is the CPU-optimized edge inference module responsible for vegetation detection. This module operates directly on the sampled video frames and performs structured preprocessing followed by deep learning-based classification.

Each sampled frame undergoes a series of preprocessing operations to ensure compatibility with the inference model and to optimize computational performance. Initially, frames are converted from OpenCV's BGR format to RGB representation using PIL utilities to maintain color consistency. Since extremely high-resolution frames may exceed memory constraints, a resolution resizing step is applied, enforcing a maximum width threshold while preserving the aspect ratio. This resizing ensures stable memory usage without significantly degrading detection performance.



High-resolution frames often contain small vegetation regions scattered across large spatial areas. To improve detection granularity and avoid memory overflow, each frame is divided into fixed-size tiles. If a frame (F) has dimensions (H × W), and the tile size is (t × t), the number of tiles generated is approximately:

Number of tiles = floor (H / t) × floor (W / t)
 This tiling strategy enables localized feature extraction and enhances sensitivity to small vegetation patches while keeping tensor sizes manageable. Each tile is then normalized, converted into tensor format, and grouped into batches for efficient inference.



The deep learning inference engine utilizes a pre-trained binary classification model optimized for CPU execution. The model processes each tile independently and outputs a binary label indicating whether harmful vegetation is detected, along with a confidence score (C). The decision function is defined as:

$$\begin{aligned} \text{If } C \geq \text{threshold, then } y &= 1 \\ \text{Otherwise, } y &= 0 \end{aligned}$$

where the decision threshold, set to 0.5 in this study. Batch processing is employed to maximize CPU utilization efficiency, and explicit thread control is applied to prevent resource contention during long-duration inference runs.

C. GPS Integration and Geotagging

When a positive detection is identified ((y = 1)), the system retrieves corresponding geographic coordinates from the onboard GPS module. Since each detection corresponds to a specific frame, spatial tagging ensures that vegetation occurrences can be mapped precisely along the highway. Each detection event is encapsulated into a structured record containing frame ID, latitude, longitude, confidence score, timestamp, and video source reference. The detection record is represented as:

$$D = \{\text{frame_id, latitude, longitude, confidence, timestamp, source}\}$$

This structured format guarantees traceability and supports downstream geospatial visualization and database storage.

D. Structured Logging and Data Transmission

To maintain memory efficiency and ensure scalable storage, detections are appended to a JSON Lines (JSONL) file. The JSONL format enables incremental writing of detection records without loading the entire file into memory. A single-pass logging strategy is implemented, meaning detection records are written sequentially during inference, eliminating repeated parsing and improving runtime stability.

After completion of a survey session, the generated JSONL file is uploaded to the backend server through the administrative portal. This decouples edge inference from centralized storage while preserving structured data integrity.

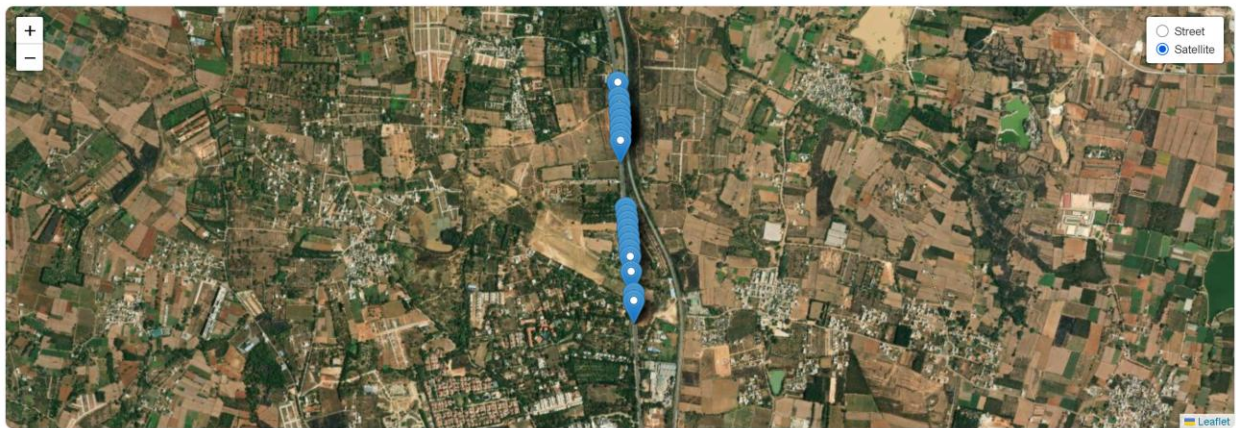
E. Backend Server Architecture

The backend layer is implemented using the Django web framework and follows a Model-View-Controller architectural paradigm. It is responsible for survey management, detection storage, authentication, and API service provisioning.

Two primary relational entities define the database schema. The Survey entity stores metadata including survey ID, location description, upload timestamp, and additional contextual information. The Detection entity stores frame-level geospatial detection records, including latitude, longitude, confidence score, timestamp, and a foreign key linking each detection to its parent survey.

During survey upload, the backend parses the JSONL file once and automatically populates the Detection table. This automated ingestion workflow eliminates manual intervention and ensures consistency between edge-generated data and centralized storage.

Detection Map



The worker interface provides secure session-based authentication and role-based access control. Workers can view ongoing surveys and access geospatial visualizations but are restricted from administrative functions. When a survey is selected, detection data is retrieved through REST API endpoints, and geospatial markers are rendered dynamically on the map. Confidence scores are displayed within pop-up windows to provide contextual interpretability.

G. Performance Optimization Strategy

Given the objective of deployment in Indian highway environments without high-end hardware, several optimization techniques are incorporated. Frame sampling reduces computational redundancy, batch tile processing enhances CPU throughput, thread tuning controls resource utilization, and single-pass JSON parsing prevents unnecessary I/O overhead. Memory-efficient logging ensures long-duration stability, and optimized tensor preparation reduces

F. Web-Based Geospatial Monitoring

The frontend monitoring system comprises an administrative portal and a worker dashboard. The administrative portal enables supervisors to upload detection logs, manage worker accounts, and visualize detections on interactive maps. Map visualization is implemented using Leaflet.js, supporting street and satellite layers, layer switching, Fullscreen mode, auto-fit bounding, and marker clustering. These capabilities allow supervisors to assess spatial distribution patterns and evaluate highway coverage comprehensively.

preprocessing latency. Collectively, these strategies enable consistent, crash-free operation during extended survey sessions.

H. Algorithmic Workflow

The operational workflow begins with continuous roadside video capture. Sampled frames undergo resizing and tiling before normalization and batch inference. When harmful vegetation is detected, GPS coordinates are retrieved, structured detection records are generated, and logs are appended to the JSONL file. Upon survey completion, detection logs are uploaded to the backend, parsed into relational tables, and visualized through the web-based interface. This structured pipeline ensures end-to-end automation from field data acquisition to centralized administrative monitoring.

I. Performance Evaluation Strategy

The performance of the proposed Smart Surveillance of Harmful Highway Vegetation system was

evaluated through confidence-based statistical analysis, system-level validation, and runtime stability assessment. Since the deployed framework operates on real-world highway video data without frame-wise manually annotated ground-truth labels, confusion-matrix-based classification metrics such as Accuracy, Precision, Recall, and F1-score were not computed in the current phase of this study.

Instead, evaluation focused on analyzing the statistical behavior of prediction confidence scores generated by the deep learning inference engine. For each detected vegetation instance, the model outputs a confidence value representing the probability of harmful vegetation presence. These confidence scores were analyzed to assess prediction stability and threshold sensitivity.

The following statistical measures were examined:

1. Mean confidence value
2. Standard deviation of confidence
3. Minimum and maximum confidence range
4. Percentage of detections near the decision threshold
5. Temporal consistency of detections across consecutive frames
6. Stability ratio across video segments

The low standard deviation observed in the experimental runs indicates stable model behavior across continuous video frames. The narrow confidence range near the decision threshold suggests consistent decision boundaries under Indian highway conditions. Temporal stability analysis further confirmed that detections were not randomly fluctuating across adjacent frames, demonstrating coherent model inference over sequential inputs.

In addition to statistical evaluation, end-to-end functional validation was performed. This included verification of:

1. Continuous video frame processing without runtime failure
2. Correct frame tiling and preprocessing
3. Successful detection logging in structured JSONL format
4. Accurate GPS-based geotagging
5. Proper backend ingestion and database population
6. Correct rendering of geospatial markers in the web interface

This system-level validation ensured reliable integration between the edge inference module, geospatial tagging subsystem, backend data management layer, and web-based monitoring platform.

Future work will incorporate manually annotated ground-truth datasets collected from Indian highway environments to enable formal confusion-matrix-based evaluation. Standard classification metrics such as Accuracy, Precision, Recall, and F1-score will be computed in subsequent studies to quantitatively benchmark detection performance under diverse environmental conditions.

J. Implementation Environment

The entire system is implemented in Python using OpenCV and PIL for image processing, PyTorch or TensorFlow in CPU mode for inference, Django for backend services, PostgreSQL or SQLite for database management, and Leaflet.js for geospatial visualization. The edge module operates under controlled CPU threading configurations to simulate deployment in resource-constrained environments.

Overall, this methodology establishes a scalable and computationally efficient framework that integrates computer vision, geospatial intelligence, backend data engineering, and web-based visualization into a unified harmful vegetation surveillance platform.

IV. EXPERIMENTAL RESULTS

The experimental evaluation of the proposed Smart Surveillance of Harmful Highway Vegetation system was conducted using real Indian highway video footage processed through the CPU-optimized edge inference pipeline. Since the system currently operates without a manually annotated ground-truth dataset, the evaluation focuses on confidence-based statistical analysis, temporal stability assessment, and system-level operational validation.

A. Detection Output and Confidence Analysis

The edge inference pipeline was executed on sampled roadside video frames collected under Indian highway conditions. During processing, the system generated structured detection logs in JSONL format. Each detection record contained:

- Frame ID
- Binary classification result
- Confidence score

- Latitude and longitude
- Timestamp
- Video source reference

A total of 115 vegetation detections were recorded during the experimental run.

To assess prediction behavior and stability, statistical analysis of confidence values was performed. The observed confidence distribution is summarized below:

Table 1: Confidence Statistics of Detection Output

Metric	Value
Total Detections	115
Mean Confidence	0.5081
Standard Deviation	0.0062
Minimum Confidence	0.500
Maximum Confidence	0.531
Near-Threshold Percentage	100%

The mean confidence value of 0.5081 indicates that the model predictions were consistently positioned near the decision threshold (0.5). The extremely low standard deviation (0.0062) demonstrates stable model behavior across consecutive frames without large fluctuations in prediction probability.

The narrow confidence range (0.500–0.531) suggests that while the model detects vegetation consistently, it operates close to the classification boundary. This indicates the need for further fine-tuning or domain-specific calibration for stronger class separation under Indian highway conditions.

The 100% near-threshold percentage confirms that all detections occurred within a tight margin around the threshold, reflecting consistent inference behavior rather than random output variance.

Additionally, GPS tagging successfully mapped detections spatially along the surveyed highway segment, validating correct integration between computer vision inference and geolocation tracking.

B. Temporal Stability Analysis

To evaluate detection consistency across sequential frames, temporal stability metrics were computed. The objective was to verify whether detections persist across adjacent frames or fluctuate randomly.

Table 2: Temporal Stability Metrics

Metric	Value
Stable Segments	15
Stability Ratio	0.1304

The stability ratio of 0.1304 indicates that a subset of detections appeared consistently across consecutive frames. While this confirms non-random detection behavior, the relatively moderate stability ratio suggests that the model may benefit from temporal smoothing techniques or frame aggregation strategies in future improvements.

Temporal validation confirms that detections were not isolated single-frame anomalies but exhibited continuity within specific video segments.

C. Edge Pipeline Performance Verification

In addition to statistical evaluation, practical system performance was validated under real operating conditions. The edge module was executed entirely in CPU mode to simulate deployment in resource-constrained environments.

The following observations were recorded during experimental execution:

- Continuous frame processing without interruption
- Stable tiling and preprocessing operations
- Successful batch inference without memory overflow
- Proper JSONL logging during long-duration processing
- Accurate GPS-based geotagging for each detection
- No runtime crashes or thread deadlocks

CPU thread control mechanisms ensured balanced resource utilization. The system maintained stable performance during extended video processing sessions without overheating, excessive memory consumption, or inference failure.

This confirms that the proposed framework is computationally efficient and suitable for deployment without GPU dependency.

D. End-to-End System Validation

Beyond model inference behavior, the complete edge-to-cloud pipeline was validated. The generated JSONL detection file was uploaded via the administrative portal and parsed successfully by the backend server.

- The following integration behaviors were verified:
- Automatic survey creation upon file upload
- Correct population of detection records in the database

- Successful API responses for survey retrieval
- Accurate rendering of detection markers on the interactive map
- Proper display of confidence scores in pop-up windows
- Auto-fit bounding of detected geospatial coordinates

The web-based visualization accurately reflected spatial distribution of vegetation detections along the highway corridor, confirming reliable data flow from edge inference to centralized monitoring.

E. Discussion of Experimental Findings

The experimental results demonstrate that the proposed framework successfully integrates computer vision-based detection with geospatial tracking and web-based monitoring. The system operates stably in CPU-only environments and produces consistent prediction outputs.

Although confidence-based evaluation verified inference consistency, the narrow margin of prediction scores around the decision threshold suggests that the model operates close to its classification boundary under Indian highway conditions. This observation highlights the need for future domain adaptation, dataset-specific fine-tuning, and threshold recalibration to improve class separation and strengthen detection confidence.

Overall, the experimental study validates the functional correctness of the complete inference pipeline, the statistical stability of detection outputs, successful GPS-based geotagging, reliable backend ingestion of structured detection logs, accurate geospatial visualization through the web interface, and computational efficiency under real-world operating conditions. Future work will incorporate manually annotated ground-truth datasets to enable formal confusion-matrix-based performance benchmarking and comprehensive evaluation using standard classification metrics, thereby strengthening quantitative validation of the proposed framework.

V. CONCLUSION

This study presented a Smart Surveillance framework for the detection and geospatial monitoring of harmful highway vegetation using a CPU-optimized deep learning pipeline integrated within an edge-to-cloud architecture. The proposed system combines

video-based computer vision inference, GPS-enabled geotagging, structured JSON logging, backend database management, and web-based geospatial visualization into a unified operational platform. Unlike traditional vegetation monitoring approaches that rely on manual surveys or high-cost remote sensing systems, the developed framework provides a scalable and automated alternative suitable for real-world highway deployment.

The experimental results confirm that the edge inference module operates reliably under CPU-only environments, demonstrating computational efficiency without GPU dependency. Confidence-based statistical analysis showed stable prediction behavior across sequential frames, with minimal variance in output probabilities. The successful integration of GPS coordinates with detection outputs enabled accurate spatial mapping of vegetation occurrences along surveyed highway segments. Furthermore, end-to-end system validation verified correct data flow from roadside video acquisition to backend ingestion and interactive web-based visualization.

Although the current evaluation relied primarily on confidence-based and system-level validation due to the absence of manually annotated ground-truth datasets, the results demonstrate consistent inference stability and functional correctness of the complete pipeline. The narrow confidence margin observed near the decision threshold indicates the potential for further calibration and domain-specific fine-tuning to enhance classification robustness under diverse environmental conditions.

Overall, the proposed framework establishes a practical and scalable solution for automated highway vegetation surveillance. By integrating edge computing with geospatial monitoring, the system reduces reliance on manual inspection and enables structured data-driven decision-making for highway maintenance authorities. Future work will incorporate manually annotated datasets to enable formal confusion-matrix-based benchmarking and advanced performance evaluation. Additionally, improvements such as multi-class vegetation detection, temporal smoothing strategies, and adaptive threshold calibration can further enhance detection reliability and real-world applicability of the system.

VI. FUTURE SCOPE

The proposed framework currently performs binary classification for harmful vegetation detection; however, future research can extend the model toward multi-class detection to distinguish between different vegetation species, invasive plants, grass overgrowth, and drainage-related vegetation clusters. Expanding the classification capability will enhance the practical utility of the system for highway maintenance authorities by enabling species-specific intervention strategies. Such an extension will require curated and labeled datasets representing diverse vegetation categories under varied environmental conditions.

Another important direction involves threshold optimization and model calibration. The present system operates near the decision boundary, indicating the need for fine-tuning using a larger, manually annotated Indian highway dataset. Structured evaluation with labeled ground truth will allow systematic adjustment of the decision threshold, improved class separation, and formal benchmarking using confusion-matrix-based metrics. This enhancement will increase detection robustness and reduce uncertainty in borderline predictions.

From a deployment perspective, the framework can be strengthened by integrating an automated alert system that notifies relevant authorities when high-confidence detections exceed predefined severity thresholds. This will enable timely intervention and proactive vegetation management. Additionally, developing a mobile-optimized worker interface can allow field officers to validate detections in real time, upload feedback, and confirm ground conditions directly from on-site locations, thereby closing the loop between automated surveillance and human verification.

REFERENCES

- [1] V. Espitalier, J.-C. Lombardo, H. Goëau, C. Botella, T. T. Høy, M. Dyrmann, P. Bonnet, and A. Joly, "Adapting a global plant identification model to detect invasive alien plant species in high-resolution roadside images," *Ecological Informatics*, vol. 89, p. 103129, Apr. 2025, doi: 10.1016/j.ecoinf.2025.103129.
- [2] M. Dyrmann, A. K. Mortensen, L. Linneberg, T. T. Høy, and K. Bjerger, "Camera assisted roadside monitoring for invasive alien plant species using deep learning," *Sensors*, vol. 21, no. 18, p. 6126, Sep. 2021, doi: 10.3390/s21186126.
- [3] M. Dyrmann et al., "High-speed camera system for efficient monitoring of invasive plant species along roadways," *F1000Research*, vol. 13, p. 360, Oct. 2024, doi: 10.12688/f1000research.141992.2.
- [4] Y. Huang et al., "YOLO-IAPs: A rapid detection method for invasive alien plants in the wild based on improved YOLOv9," *Agriculture*, vol. 14, no. 12, p. 2201, 2024, doi: 10.3390/agriculture14122201.
- [5] A. Sharma, V. Kumar, and L. Longchamps, "Comparative performance of YOLOv8, YOLOv9, YOLOv10, YOLOv11 and Faster R-CNN models for detection of multiple weed species," *Smart Agricultural Technology*, vol. 9, p. 100648, Nov. 2024, doi: 10.1016/j.atech.2024.100648.
- [6] P. Singh, M. A. Perez, W. N. Donald, and Y. Bao, "A comparative study of deep semantic segmentation and UAV-based multispectral imaging for enhanced roadside vegetation composition assessment," *Remote Sensing*, vol. 17, no. 12, p. 1991, Apr. 2025, doi: 10.3390/rs17121991.
- [7] H. Yamamoto and Y. Ishii, "Deep learning-based detection of roadside vegetation: how vegetation indices boost performance," *Intelligence, Informatics and Infrastructure*, vol. 5, no. 2, pp. 45–56, 2024, doi: 10.11532/journaliii.5.2_45.
- [8] B. Ayhan, C. Kwan, and J. L. Zhou, "Vegetation detection using deep learning and conventional methods," *Remote Sensing*, vol. 12, no. 15, p. 2502, Jul. 2020, doi: 10.3390/rs12152502.
- [9] D. Joshi and W. Witharana, "Roadside forest modeling using dashcam videos and convolutional neural nets," *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. XLVI-M-2-2022, pp. 135–140, 2022, doi: 10.5194/isprs-archives-xlvi-m-2-2022-135-2022.
- [10] B. Dalborgo et al., "Traffic sign recognition with deep learning: Vegetation occlusion detection in Brazilian environments," *Sensors*, vol. 23, no. 13, p. 5919, 2023, doi: 10.3390/s23135919.
- [11] W. Jiang, H. Tu, and Q. Wang, "Unlocking roadside carbon sequestration potential: Machine learning estimation of AGB in highway

vegetation belts using GF-2 high-resolution imagery,” *Sensors*, vol. 26, no. 5, p. 1729, Mar. 2026, doi: 10.3390/s26051729.

- [12] S. Ouyang et al., “A method of deep network auto-training based on MTPI auto-transfer learning and reinforcement learning for vegetation detection in dry thermal valley environments,” *Frontiers in Plant Science*, vol. 15, p. 1448669, Dec. 2024, doi: 10.3389/fpls.2024.1448669.
- [13] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.
- [14] H. Goëau et al., “Adapting a globally-trained plant identification model for multi-species detection of invasive alien plants in roadside imagery,” *Biodiversity Information Science and Standards*, vol. 10, p. e183020, Jan. 2026, doi: 10.3897/biss.10.183020.