Indo American Journal of Multidisciplinary Research and Review (IAJMRR)
International Peer Reviewed - Refereed Research Journal
ISSN: 2581 - 6292, Impact Factor: 6.885, Website: www.iajmrr.com

RAILWAY TRACK OBSTACLE DETECTION USING CNN FOR ACCIDENT AVOIDANCE

T. Durga*, V. A. Kishore, S. Gowtham** & S. Irfan Sharif****

* Assistant Professor, Department of Electrical and Electronics Engineering, Dhanalakshmi Srinivasan Engineering College (Autonomous), Perambalur, Tamil Nadu, India

** UG Student, Department of Electrical and Electronics Engineering, Dhanalakshmi Srinivasan Engineering College (Autonomous), Perambalur, Tamil Nadu, India

Cite This Article: T. Durga, V. A. Kishore, S. Gowtham & S. Irfan Sharif, "Railway Track Obstacle Detection Using CNN for Accident Avoidance", Indo American Journal of Multidisciplinary Research and Review, Volume 10, Issue 1, January - June, Page Number 116-119, 2026.

Type of Review: Peer Reviewed as per |C|O|P|E| Guidance.

DOI: https://doi.org/10.5281/zenodo.19827936

Abstract:

Human-animal conflict near railway tracks has become a serious safety and ecological concern in many regions. Frequent incidents of wildlife crossing railway lines often result in accidents, loss of animal life, and disruption of railway operations. To address this issue, this article proposes a real-time intelligent monitoring system that integrates deep learning, image processing, and Internet of Things (IoT) technologies. The system is designed to automatically detect animal intrusions using motion-triggered cameras and classify captured images using a Convolutional Neural Network (CNN)-based model. By distinguishing actual threats from false triggers such as moving shadows or environmental disturbances, the proposed approach enhances detection accuracy and ensures timely alerts to railway authorities, thereby improving safety and reducing response time.

1. Introduction:

Animal detection using artificial intelligence has emerged as an important research area in wildlife conservation, smart surveillance, and transportation safety systems. With the increasing expansion of railway networks through forested and rural areas, the risk of wildlife collisions has significantly increased. These incidents not only cause damage to railway infrastructure but also threaten biodiversity and ecosystem balance. This project focuses on developing an automated system that can detect animals near railway tracks in real time and issue immediate alerts. The system leverages machine learning and computer vision techniques to analyze live video feeds and identify potential intrusions. By eliminating the need for constant human monitoring, the solution ensures continuous surveillance and faster decision-making, ultimately reducing accident risks.

2. Artificial Intelligence and Machine Learning Foundations:

- Artificial Intelligence (AI): Artificial Intelligence refers to the development of systems capable of performing tasks that typically require human intelligence. These tasks include reasoning, decision-making, learning from experience, and pattern recognition. In this system, AI is used to analyze environmental data and detect anomalies such as unexpected animal movement near railway tracks.
- Machine Learning (ML): Machine Learning is a subset of AI that enables systems to learn from data without being explicitly programmed. ML algorithms improve their performance over time by identifying patterns in training data. In this project, ML plays a crucial role in classifying images and distinguishing between animals and non-animal objects.

3. Deep Learning and CNN Architecture:

Deep Learning (DL):

Deep Learning is an advanced subset of machine learning that uses multi-layered neural networks to learn complex patterns from large datasets. It is widely used in applications such as autonomous vehicles, medical imaging, and object detection. In this project, deep learning enables the system to recognize animals accurately even in complex environments.

Convolutional Neural Networks (CNN):

CNNs are specialized deep learning models designed for image processing tasks. They automatically extract meaningful features from images and classify them into predefined categories. The CNN architecture used in this system consists of:

- Input Layer: Accepts image data in matrix form.
- Convolutional Layer: Extracts features such as edges, shapes, and textures.

- Pooling Layer: Reduces dimensionality and computational complexity.
- Flatten Layer: Converts feature maps into a single vector.
- Fully Connected Layer: Performs classification based on extracted features.
- Softmax Layer: Produces final probability scores for each class.

This layered structure enables highly accurate image classification and real-time decision-making.

4. Literature Review:

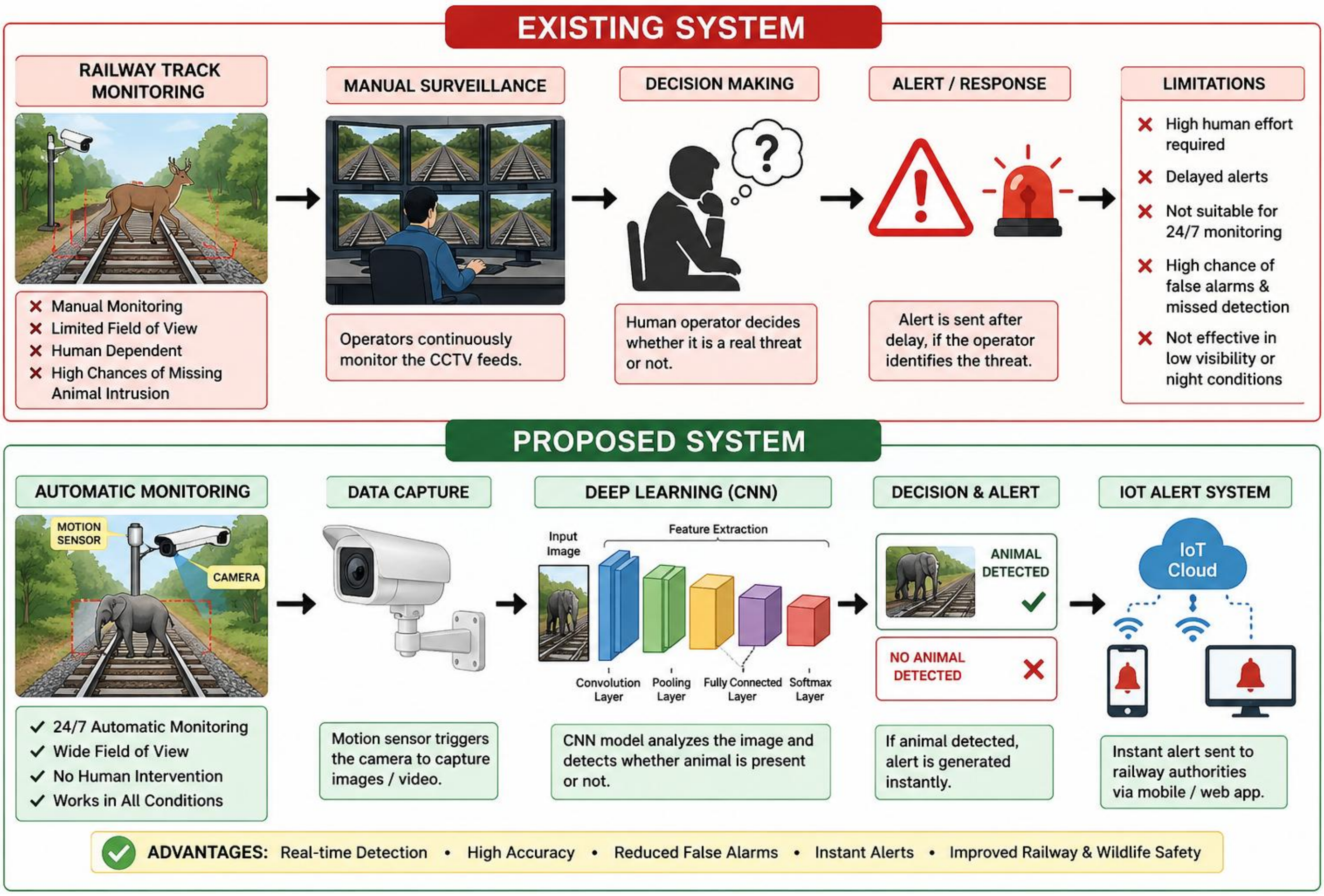
Several research contributions have influenced the development of this system. Sabnis et al. (2019) proposed object detection techniques for unmanned railway crossings using SSD models. Raffik and Hemanath (2021) explored IoT-based communication systems for transmitting obstacle alerts to railway officials, significantly improving safety response times.

Mamat and Othman (2023) demonstrated the effectiveness of YOLOv5 in animal detection, achieving high accuracy in agricultural environments. Similarly, Gandhi and Gupta (2022) compared multiple CNN-based models and concluded that optimized Faster R-CNN architectures provide a balanced trade-off between speed and accuracy for real-time applications.

These studies highlight the growing importance of deep learning-based detection systems in safety-critical environments.

5. Existing System:

Traditional animal detection methods rely heavily on manual surveillance or basic motion detection systems. These approaches often suffer from high false alarm rates and lack adaptability to complex environments. Moreover, many systems fail to operate efficiently in real-time due to computational limitations.



6. Proposed System:

The proposed system introduces a CNN-based automated detection framework integrated with IoT communication. Unlike conventional methods, this system uses high-resolution cameras and deep learning models to accurately detect animals near railway tracks. The model is trained to differentiate between actual animal presence and irrelevant disturbances such as moving vegetation or lighting changes.

7. Implementation Methodology:

The system implementation consists of several structured modules:

- Dataset Collection: Large datasets of animal images are collected from CCTV footage and public datasets such as Kaggle to ensure diversity in training data.
- Image Preprocessing: Images are resized, normalized, and labeled to ensure uniform input for the neural network.

- Model Training: The CNN model is trained using labeled datasets to learn distinguishing features of different animal classes.
- Prediction Module: The trained model is used to analyze real-time video frames and classify detected objects.
- Alert System: Once a threat is detected, alerts are generated and transmitted through IoT devices to railway control centers.

8. Software Environment:

- Python Programming: Python is chosen for its simplicity, flexibility, and strong support for machine learning libraries. It allows rapid development of AI-based applications and seamless integration with data processing tools.
- NumPy Library: NumPy is essential for handling multi-dimensional arrays and performing mathematical operations efficiently. It plays a key role in preprocessing image data before feeding it into the CNN model.

9. Computer Vision Using OpenCV:

OpenCV is an open-source computer vision library used for real-time image and video processing. It converts visual data into numerical pixel values that can be analyzed by machine learning models.

The system uses OpenCV to capture live video feeds, process frames, and pass them to the CNN model for classification. Its high performance and low memory usage make it suitable for real-time railway monitoring applications.

10. IoT Integration and Alert Mechanism:

The integration of IoT technology enhances the system's ability to communicate detected threats instantly. When the CNN model identifies an animal near the track, the system sends real-time alerts to railway authorities via cloud-connected IoT modules.

These alerts can be accessed through mobile or web applications, enabling immediate preventive action. This ensures that even remote and forested railway areas are effectively monitored without human intervention.

11. Results and Discussion:

The experimental results demonstrate that the proposed system achieves high accuracy in detecting animals in various environmental conditions. The use of deep learning significantly reduces false positives compared to traditional methods. The system performs efficiently in real-time scenarios, ensuring fast detection and response. By filtering unnecessary alerts, it improves operational reliability and reduces workload on monitoring personnel.

11. Conclusion:

The proposed real-time animal intrusion detection system presents an effective solution for enhancing railway track safety. By combining deep learning, computer vision, and IoT technologies, the system ensures continuous monitoring and accurate detection of wildlife near railway lines. This approach significantly reduces human-animal conflict, minimizes railway accidents, and supports wildlife conservation efforts. The system is scalable, cost-effective, and suitable for deployment in rural and forest railway networks, making it a promising solution for modern intelligent transportation systems.

References:

1. Sabnis, S., et al. (2019). Unmanned Railway Crossing System Using SSD Object Detection. IEEE Conference on Intelligent Transportation Systems (ITSC), pp. 112-117.
2. Raffik, M., & Hemanath, S. (2021). IoT-Based Railway Obstacle Detection and Alert System Using MQTT Protocol. International Journal of Advanced Computer Science and Applications, 12(6), pp. 45-52.
3. Mamat, N., & Othman, R. (2023). YOLOv5-Based Animal Detection in Agricultural Fields. IEEE Access, 11, pp. 89234-89245.
4. Gandhi, R., & Gupta, A. (2022). Performance Comparison of RCNN, Faster RCNN and SSD for Object Detection. Journal of Machine Learning Research Applications, 5(3), pp. 201-210.
5. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779-788.
6. Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(6), pp. 1137-1149.
7. Krizhevsky, A., Sutskever, I., & Hinton, G. (2012). ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems (NeurIPS), pp. 1097-1105.
8. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press, Cambridge, MA, pp. 1-775.
9. Bradski, G. (2000). The OpenCV Library. Dr. Dobb's Journal of Software Tools, 25(11), pp. 120-125.
10. McKinney, W. (2010). Data Structures for Statistical Computing in Python (NumPy & Pandas). Proceedings of the 9th Python in Science Conference (SciPy), pp. 56-61.

11. Chollet, F. (2017). Deep Learning with Python. Manning Publications, pp. 1-384.
12. Russell, S., & Norvig, P. (2021). Artificial Intelligence: A Modern Approach (4th Edition). Pearson, pp. 1-1132.
13. Sharma, A., & Singh, P. (2022). Real-Time Object Detection Using Deep Learning for Smart Surveillance Systems. IEEE Sensors Journal, 22(8), pp. 7432-7441.
14. Kumar, R., & Patel, H. (2021). IoT-Based Smart Monitoring Systems for Railway Safety Applications. International Journal of Internet of Things, 10(2), pp. 89-97.
15. Zhang, L., et al. (2020). Deep Learning for Object Detection in Intelligent Transportation Systems. IEEE Transactions on Intelligent Transportation Systems, 21(10), pp. 1-14.