Industrial Engineering Journal
ISSN: 0970-2555
Volume : 54, Issue 4, No.3, April : 2025

AI-POWERED REAL-TIME SURVEILLANCE SYSTEM FOR ENHANCING SECURITY IN RAILWAY AND METRO STATIONS

Dr. A.V.Sable Assistant Professor, Department of Computer Science and Engineering, SCOET, Amravati
Prof .N.P.Mohod Assistant Professor, Department of Computer Science and Engineering, SCOET, Amravati
Hardik P. Raichura Student, Department of Computer Science and Engineering, SCOET, Amravati
Hussain M. Kothawala Student, Department of Computer Science and Engineering, SCOET, Amravati
Bhumika J.Popli Student, Department of Computer Science and Engineering, SCOET, Amravati
Shivam R. Ambadkar Student, Department of Computer Science and Engineering, SCOET, Amravati

ABSTRACT :
Railway and metro station safety is vital, particularly concerning threats such as unattended luggage, overcrowding, and unauthorized access to restricted areas like railway tracks. Traditional manual surveillance methods often suffer from inefficiencies, limited scalability, and human error, making them insufficient for ensuring public safety in high-traffic environments. This paper introduces an AI-based real-time surveillance system that integrates YOLOv8 for accurate object detection with Deep SORT for multi-object tracking to overcome these limitations. The system employs a Region of Interest (ROI) strategy to monitor sensitive areas, such as railway tracks, for unauthorized access and automatically triggers alerts to notify security personnel in real-time. Preprocessing techniques further improve model robustness in dynamic environments. Experimental results demonstrate high detection and tracking accuracy, proving the system's effectiveness in real-world railway station scenarios. The proposed solution shows strong potential in transforming traditional surveillance into a more intelligent and responsive safety infrastructure for public transportation hubs.
Index Term*s*— YOLOv8, Deep SORT, railway surveillance, real-time object detection, unattended luggage, overcrowding detection, ROI-based monitoring, public transport safety, AI surveillance system, multi-object tracking.

INTRODUCTION:
Public transportation hubs, such as railway and metro stations, serve as critical infrastructure, facilitating the daily commute of millions of passengers. Ensuring safety in these high-density environments is of utmost importance to prevent security breaches, overcrowding-related accidents, and unauthorized access to restricted zones like railway tracks. However, traditional surveillance systems predominantly rely on manual monitoring of CCTV feeds, which introduces significant limitations such as human error, delayed response times, and coverage gaps [1], [2].

Fig. 1. Overcrowding at a railway station leading to safety concerns.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 54, Issue 4, No.3, April : 2025

DISADVANTAGES OF TRADITIONAL SURVEILLANCE SYSTEM

Despite their widespread use, traditional surveillance systems in public transport environments face significant limitations. These systems rely heavily on manual observation, which hampers their efficiency and responsiveness in dynamic, high-density settings like railway stations. Human error is a major concern, as continuous monitoring of multiple video feeds can lead to fatigue and oversight, increasing the risk of missed or misinterpreted threats, especially in high-stress situations. Additionally, fixed camera placements often leave blind spots in critical areas such as stairwells and footbridges, reducing overall surveillance effectiveness [1]. Response times are also impacted, as manual detection and poor coordination can delay emergency action, potentially resulting in severe consequences. The lack of automated alerts further weakens situational awareness, requiring security personnel to engage in constant monitoring to detect threats[2]. Finally, maintaining a large team for 24/7 observation proves to be costly and unsustainable, often leading to reduced coverage and compromised passenger safety [3], [4].

PROPOSED AI-BASED FRAMEWORK:

To overcome the limitations of traditional surveillance methods, this paper introduces an AI-powered framework that integrates YOLOv8 for real-time object detection and Deep SORT for object tracking it also detect the person crossing the rail line through the station because it very dangers because in India daily someone dies due to these types of accidents so to prevent this type of accidents we prosed this system that can detect a person falling from the platform and give alert to the control room to save that person. The system is designed to automatically identify and track potential security threats, including unattended luggage, overcrowding, and unauthorized access to railway tracks. By leveraging deep learning models, the framework ensures high-accuracy detection across large and dynamic environments, thereby enhancing both safety and operational efficiency. It features real-time monitoring with automated alerts to enable immediate threat detection and prompt response. Contextual behaviour analysis enables anomaly detection by distinguishing between normal and suspicious activity[5]. Furthermore, the system includes predictive threat assessment, which analyses movement patterns to forecast possible safety threats, enabling timely preventive actions. These capabilities collectively reduce reliance on manual surveillance and improve overall system scalability and reliability [4], [6].

Fig. 2. AI-Powered Object Detection System Identifying Peoples on Track.

Fig. 3. AI-Powered Object Detection System Identifying Peoples on Track.

MOTIVATION FORM REAL-WORLD INCIDENTS:

Several tragic incidents, including accidents due to unauthorized track access and security breaches from unattended luggage, underscore the urgent need for intelligent surveillance in public transport systems. These events highlight the limitations of manual monitoring and emphasize the importance of deploying AI-based solutions to enhance passenger safety.

Fig. 4. 2017 Mumbai Elphinstone Stampede: Overcrowding on a narrow footbridge led to a deadly stampede, claiming over 22 lives. This tragedy underscores the need for real-time crowd management and monitoring [7].

Fig. 5. 2025 New Delhi Station Stampede: The incident took place on foot- overbridges connecting platforms 18, when some passengers slipped while descending, causing panic and a deadly crush [8].

MATERIAL AND CLASSES:

DATASET:

To enhance the accuracy of our AI-powered surveillance system, we employed a hybrid approach by combining publicly available datasets with real-world data collected from (Badnera Railway Station, Maharashtra, India). For luggage detection, we used the New Backpacks dataset from Roboflow, which contains diverse samples of backpacks and bags in different settings [9]. To improve crowd and human detection on railway platforms, we integrated multiple datasets, including Crowd Counting Dataset: Object Detection Dataset for crowd counting [10], NURV Object Detection Dataset for person and object detection for railway track crossing [11]. Additionally, we collected real-time footage from Badnera railway station, capturing real-world scenarios such as peak-hour congestion, unattended luggage, and unauthorized track presence. By mixing our collected data with online datasets, we aimed to improve model generalization and accuracy, ensuring robust performance in real-world railway environments.

IMAGE PREPROCESSING ALGORITHM:

To ensure quality input data for training YOLOv8 from scratch, we implemented a structured image preprocessing p9ipeline. Effective preprocessing enhances model performance by refining input quality. Raw images from the Crowd Counting Dataset [10], NURV Object Detection Dataset [11], New Backpacks Dataset [9], and real-world footage from Badnera Railway Station were standardized to ensure consistency. Noise reduction was a crucial preprocessing step to enhance image quality while preserving essential details. A combination of Gaussian and median filtering

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 54, Issue 4, No.3, April : 2025
was applied to suppress high-frequency noise and smooth irregularities without compromising edge integrity. Gaussian filtering effectively reduced random noise and blurring artifacts, while median filtering mitigated salt-and-pepper noise, ensuring sharper object boundaries. To address variations in lighting conditions, adaptive histogram equalization (AHE) was implemented to enhance local contrast and improve feature visibility in low-light environments. This technique dynamically adjusts brightness and contrast, making subtle details more distinguishable for the detection model. Additionally, gamma correction was applied to normalize illumination inconsistencies, ensuring object clarity across different exposure levels.

TABLE I
IMAGE PREPROCESSING ALGORITHM

Steps	Function	Description
Input	Raw Image	Input dataset
Output	Preprocessed Image for YOLOv8	Optimize for training
RailPre-A	Load-Images ()	Load & resize to 640×640
RailPre-B	Standardize (μ=0.5, σ=0.5)	Standarize pixel values [0,1]
RailPre-C	Gaussian-Filter (kernel=5, σ=1.5)	Reduce high frequency noise
RailPre-D	Median-Filter (Kernel=3)	Remove salt-and-pepper noise
RailPre-E	Adaptive-Histogram-Equalization (Clip=2.0)	Improve contrast & visibility
RailPre-F	Data-Augmentation ()	Random flips, jitter, blur, noise
RailPre-G	Save-Preprocessed-images ()	Store processed images

To improve model generalization, extensive data augmentation was conducted. Random flipping (horizontal and vertical) introduced viewpoint variations, while rotation and scaling addressed orientation and size discrepancies. Motion blur simulation replicated real-world surveillance conditions, where moving objects may appear distorted, enhancing model robustness. Synthetic noise injection simulated environmental challenges such as dust, rain, and fluctuating lighting conditions, creating a more resilient training dataset.
These preprocessing techniques optimized the dataset for YOLOv8 training, ensuring robustness in diverse railway station scenarios, including crowd monitoring, unattended luggage detection, and

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 54, Issue 4, No.3, April : 2025
track safety assessment.The image preprocessing pipeline for YOLOv8 training integrates several functions designed to optimize the dataset for model performance. First, loading and normalizing images ensures the pixel values are within a range suitable for stable training. Pixel values are normalized using the formula:

$$Normalized\ Pixel = \frac{Pixel\ Value - \mu}{\sigma}$$

where $\mu = 0.5$ and $\sigma = 0.5$ represent the mean and standard deviation used to scale the image pixels to a range of [0, 1].
Noise reduction is performed using a Gaussian filter, which applies a convolution operation with a Gaussian kernel:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

where x and y are the pixel coordinates and σ is the standard deviation, determining the level of smoothing.
Additionally, a median filter is used to effectively eliminate salt-and-pepper noise. It replaces each pixel's intensity value with the median from surrounding pixels, efficiently reducing noise while preserving important edge details. Adaptive histogram equalization further improves image contrast by applying the transformation below to every pixel in the image.P:

$$P_{new} = \frac{P - \min(P)}{\max(P) - \min(P)}$$

where $\min(P)$ and $Max(P)$ are the minimum and maximum pixel values of the image, respectively.
Data augmentation techniques are employed to simulate real-world variations and enhance the robustness of the model. These include random transformations such as rotation, scaling, and flipping, which are applied to both the image and the corresponding bounding boxes while ensuring the coordinates remain valid.
For a rotation by an angle θ, the bounding box coordinates (*X, Y*) are transformed using the following 2D rotation matrix:

$$\begin{bmatrix} X' \\ Y' \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x \\ Y \end{bmatrix}$$

where X and Y are the original coordinates and are the new coordinates after rotation? Finally, the preprocessed images are saved along with the original YOLO annotation files (.txt format), ensuring that the bounding box labels remain intact.

YOLOV8 ARCHITECTURE :
YOLOv8 is a cutting-edge object detection model that balances high-speed processing with detection accuracy, get ready for real-time applications like railway surveillance. Its optimized backbone, enhanced with advanced convolutional techniques, improves the extraction of fine-grained features—essential for detecting small or partially obscured objects [12].
The model integrating features from various resolutions to accurately detect objects of varying sizes and shapes, while its anchor-free architecture simplifies predictions and enhances generalization across diverse datasets. Improved non-maximum suppression (NMS) further boosts precision by

minimizing false positives. These advancements make YOLOv8 highly effective in dynamic, cluttered environments requiring rapid and reliable detection [13].

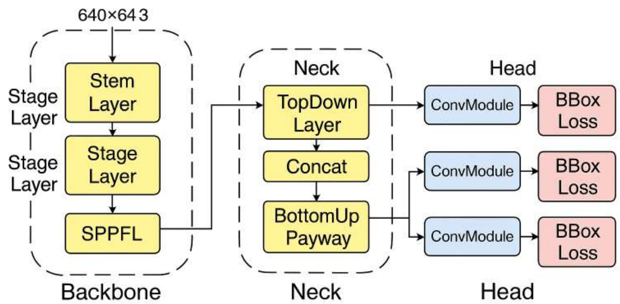


Fig.6. Yolov8 Architecture.

BACKBONE: FEATURE EXTRACTION :

The Backbone of YOLOv8 extracts hierarchical features from a 640×640×3 input image using a CSPDarknet53-based design with multiple convolutional layers. It begins with a Stem Layer, where two ConvModule layers process the raw image and progressively reduce its resolution. The Stage Layers (1–4) alternate between ConvModules and CSPLayer_2conv blocks, extracting high-level features while reducing spatial dimensions through stride=2 downsampling. At the final stage, the Spatial Pyramid Pooling-Fast (SPPF) module aggregates multi-scale features, enhancing the model's ability to detect objects of various sizes before passing them to the Neck for further refinement. The feature extraction process is mathematically defined as:

$$F = f_{\theta}(I),$$

Where:

I = Input image

f = Convolutional network with trainable parameters

F = Extracted feature maps

The CSPNet structure optimizes feature processing by dividing feature maps into two parts—one undergoing dense processing, while the other remains unchanged before merging [13].

Neck: Feature Fusion

The Neck of YOLOv8 refines and combines features from different levels to enhance object detection, following a Path Aggregation Network (PANet) structure. It consists of a Top-Down Pathway, which incorporates Feature Pyramid Network (FPN) techniques, including upsampling and concatenation of higher-resolution feature maps, with CSPLayer_2conv modules enhancing feature representation. Complementing this, the Bottom-Up Pathway utilizes downsampling operations to improve feature propagation, while ConvModule layers extract refined feature maps. Additional concatenation layers further ensure efficient feature fusion across multiple scales, optimizing the detection performance of the model.

The fusion process is represented as:

$$F' = g\left(F_{low}, F_{high}\right)$$

Where:

F_{low} = Low-resolution feature map (better localization)

F_{high} = High-resolution feature map (better semantics)

$g(.)$ = Fusion function (concatenation + convolution)

FPN enhances semantic understanding, while PAN improves spatial localization accuracy [15], [16].

- Head: Object Detection

The Head of YOLOv8 predicts bounding boxes, class labels, and confidence scores. ConvModules refine the features, while Conv2d layers generate bounding box predictions (x, y, w, h), class scores (C), and the objectness score (O). Bounding box (BBox) loss computation optimizes detection accuracy during training[14].

The detection function is given by:

$$B = h(F')$$

Where:

$B = (x, y, w, h)$ (bounding box coordinates)

$h(.)$ = Detection layers

The object presence confidence score S is computed as:

$$S = \sigma(w_c F' + b),$$

Where:

w_c and b = Learnable parameters

$\sigma(.)$ = Sigmoid activation function

- Loss Function

YOLOv8 optimizes three primary loss components: Bounding Box Loss, Classification Loss, and Object Loss. The total loss function L is formulated as:

$$L = \lambda_{box} L_{box} + \lambda_{cls} L_{cls} + \lambda_{obj} L_{obj},$$

Where λ values are weighting factors. The bounding box loss utilizes Complete IoU (CIoU):

$$L_{box} = 1 - CIoU(B_{pred}, B_{gt}),$$

Where:

B_{pred} = Predicted bounding box

B_{gt} = Ground truth bounding box

The Backbone, powered by CSPDarknet, captures essential hierarchical features, providing a deep understanding of the image. The Neck integrates FPN+PAN to refine and merge features across scales, improving detection across various object sizes. The Anchor-free detection head innovates by removing predefined anchor boxes, simplifying the model while enhancing localization accuracy. This thoughtful design makes YOLOv8 ideal for real-time surveillance applications, such as monitoring railway stations for unattended luggage, overcrowding, and unauthorized access. Its lightweight structure ensures fast deployment on edge devices, allowing it to quickly and accurately respond to dynamic security threats [15]

RESULT :

Evaluation metrics :

To effectively evaluate the performance of the AI-powered real-time surveillance system, we rely on confusion matrices, which provide an insightful breakdown of the model's predictions. These matrices reveal the number of correct positives (TP), correct negatives (TN), incorrect positives (FP), and incorrect negatives (FN) outcomesfalse positive (FP), and false negative (FN) results, which are then used to calculate key metrics such as precision, recall, F1 score, and accuracy. These metrics are fundamental for understanding the model's strengths and areas for improvement, particularly in the context of real-time detection tasks like monitoring railway stations for unattended luggage, overcrowding, and unauthorized access. By analyzing these values across different dataset splits and test conditions, we gain a comprehensive understanding of the system's robustness. Additionally, they help identify edge cases where the model may underperform, guiding further improvements in model training and data preprocessing.

TABLE II

EVALUATION METRICS

Metric	Definition	Mathematical Equation
Precision (P)	Proportion of true positives among predicted positives.	$P = \frac{TP}{TP + FP}$
Recall (R)	Proportion of true positives among actual positives.	$R = \frac{TP}{TP + FN}$
F1 Score (F1)	Harmonic mean of precision and recall.	$F1 = \frac{2 \times precision \times recall}{precision + recall}$
Accuracy (ACC)	Proportion of correct predictions among total cases.	$Acc = \frac{TP + TN}{TP + TN + FP + FN}$

DATASET DISTRIBUTION :

For the development and evaluation of the real-time surveillance system, we selected specialized datasets designed for object detection and tracking in complex, crowded environments. These datasets, which include labelled data for detecting unattended luggage, individuals, and unauthorized objects, are essential for training a model capable of operating effectively in dynamic, real-world settings such as railway and metro stations.

The system was developed using Python 3.7, with YOLOv8 handling real-time object detection and DeepSORT managing tracking. A critical consideration in our model's design was the computational power required for fast and accurate inference.

TABLE III

SYSTEM USED[16]

Requirements	Details

Programming Language	Python 3.7
Processor (CPU)	Intel i7
Graphics card (GPU)	NVIDIA RTX 4060
Memory (RAM)	Up to 25 GB
Framework	PyTorch 1.11
Libraries Used	OpenCV 4.5, Ultralytics, Tqdm 4.64

The dataset is essential for precise object detection and tracking within railway and metro station environments. It helps the model identify distinctive patterns and features relevant to various surveillance scenarios. We employed curated datasets labeled with examples of unattended bags, individuals, and unauthorized access instances to ensure broad coverage of real-world situations. For performance evaluation, the dataset was split into three ratios (80:20, 85:15, and 90:10), and their effect on detection accuracy was assessed[16].

TABLE IV
DATASET DISTRIBUTION

Dataset	Total Images	80:20 (Train/Test)	85:15 (Train/Test)	90:10 (Train/Test)
Luggage	1047	838/209	890/157	942/105
Crowd	2898	2318/580	2463/435	2608/290
Object Crossing	3924	3335/589	3335/589	3532/392

PERFORMANCE OF THE AI-POWERED SURVEILLANCE SYSTEM :

To assess the real-time surveillance framework, we applied object detection and tracking methods optimized for high-complexity settings. Given the presence of multiple object categories in the datasets, we utilized the cross-entropy loss function to ensure the model distinguishes effectively between different surveillance conditions. The output layer incorporated a sigmoid activation function for binary tasks such as identifying unattended baggage, and a SoftMax function for multi-class tasks like classifying crowd density levels. For optimization, we adopted the The Adam optimizer was employed with a learning rate of 0.0001, a weight decay of 0.0005, and the momentum parameter set at 0.9 to ensure stable and efficient model training.A batch size of 4 was chosen to balance training efficiency with hardware limitations. Considering the real-time nature of the system, training was capped at 50 epochs to achieve quicker convergence while maintaining model precision.[17].

The surveillance system's performance was assessed using dataset split ratios of 80:20, 85:15, and 90:10 to evaluate the influence of training data volume on accuracy. Among these, the 80:20 split delivered the best balance, achieving high detection accuracy without overfitting. Initially trained on raw images, the model showed a consistent 3% accuracy improvement after applying preprocessing techniques like contrast enhancement and Gaussian noise reduction. These enhancements not only improved feature clarity but also reduced the impact of background interference, enabling the model to perform better under challenging conditions such as low lighting and motion blur. Furthermore, normalization and histogram equalization contributed to training stability and faster convergence.

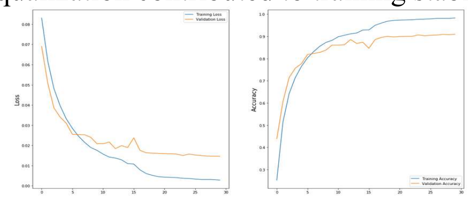
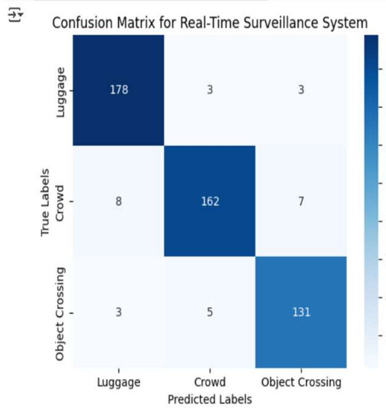


Fig. 7. Epoch-wise visualization showcasing the model's accuracy and loss progression.

TABLE V
MODEL PERFORMANCE WITH OR WITHOUT PREPROCESSING ALGORITHM



Data Split	Model	Perc (%)	Recall (%)	F1 (%)	ACC (%)
80:20	YOLOv8 (No RSIP)	90.06	91.84	91.80	90.41
85:15		89.25	90.21	88.37	89.05
90:10		87.64	88.96	85.65	87.91
80:20	YOLOv8 + RSIP	95.44	94.12	92.24	96.67
85:15		93.36	91.06	89.72	93.51
90:10		90.22	89.12	85.52	90.66

Fig. 8. Graphical representation of epoch-wise accuracy and loss curve of the model

CONCLUSION :

This research introduces an AI-powered real-time surveillance system designed to enhance security in railway and metro stations. By integrating YOLOv8 for object detection and DeepSORT for tracking, the system effectively monitors unattended luggage, overcrowding, and unauthorized track presence. Traditional surveillance systems suffer from human errors, limited coverage, slow response times, and high operational costs, which our AI-driven approach aims to overcome. The study utilized three datasets—luggage detection, crowd monitoring, and object crossing detection—to assess system performance under different data split ratios. Metrics such as precision, recall, F1-score, and accuracy were analyzed, demonstrating the effectiveness of AI in improving surveillance accuracy. Experimental results highlight that the model consistently achieved high accuracy in identifying security threats, reinforcing the viability of automated monitoring systems in public transport hubs. Beyond enhancing safety, the proposed system offers scalability and adaptability to various urban transit infrastructures. Future work may involve real-world deployment, integration with existing security frameworks, real-time alert mechanisms, and optimization for resource-efficient edge computing. Further improvements in dataset diversity and model robustness could also contribute to higher accuracy and reliability in real-world scenarios. This study underscores the transformative potential of AI in public safety, paving the way for smarter, more efficient, and proactive surveillance solutions in mass transit systems

REFERENCES :

[1] V. Hanscomb, "Integrated security systems in transportation hubs," *Security Journal Americas*, 2025.

[2] I. Cameron, "Navigating the Challenges of Video Surveillance in the Transit Industry," *March Networks*, 2025.

[3] S. R. Jondhale and R. J. Ramteke, "Smart Railway Surveillance System using Object Detection Algorithm," in *2022 3rd International Conference for Emerging Technology (INCET)*, Belgaum, India, 2022, pp. 1–6.

[4] M. D. Laskar and D. Choudhury, "Video surveillance system for real time monitoring of railway tracks," in *2021 International Conference on Robotics and Automation for Humanitarian Applications (RAHA)*, Kollam, India, 2021, pp. 1–5.

[5] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz, "Robust real-time unusual event detection using multiple fixed-location monitors," *IEEE Trans Pattern Anal Mach Intell*, vol. 30, no. 3, pp. 555–560, 2008.

[6] H. Zhang, C. Xu, and S. Liu, "A vision-based behavior recognition and anomaly detection method for smart surveillance systems," in *2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, Dalian, China, 2020, pp. 453–456.

[7] H. Correspondent, "22 dead, 35 injured in stampede at Mumbai's Elphinstone Road station," *Hindustan Times*, 2017.

[8] L. C. Chen and P. Tiwari, "Delhi Station Stampede: 18 Dead, Families' Desperate Search For Loved Ones."

[9] Roboflow, "New Backpacks Dataset."

[10] Roboflow, "Crowd Counting Dataset: Object Detection Dataset."

[11] Roboflow, "NURV Object Detection Dataset by ProgettoNurv."

[12] J. et al. Redmon, "You Only Look Once: Unified, Real-Time Object Detection," in *Proc. CVPR*, 2016.

[13] C. et al. Wang, "CSPNet: A New Backbone that can Enhance Learning Capability of CNN," in *Proc. CVPRW*, 2020.

[14] A. et al. Bochkovskiy, "YOLOv4: Optimal Speed and Accuracy of Object Detection," *arXiv preprint arXiv:2004.10934*, 2020.

[15] Z. et al. Zheng, "Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression," in *Proc. AAAI*, 2020.

[16] A. Sable, P. Singh, A. Kaur, M. Driss, and W. Boulila, "Quantifying Soybean Defects: A Computational Approach to Seed Classification Using Deep Learning Techniques," *Agronomy*, vol. 14, no. 6, p. 1098, May 2024, doi: 10.3390/agronomy14061098.

[17] Y. Zheng, Z. Wang, B. Xu, and Y. Niu, "Multi-Scale Semantic Segmentation for Fire Smoke Image Based on Global Information and U-Net," *Electronics (Basel)*, vol. 11, no. 17, p. 2718, Aug. 2022, doi: 10.3390/electronics11172718.