Optimized Machine Learning Pipeline for Object Detection in Automotive and Surveillance Systems

Nidhi Joshi Parsai[1], Sumit Jain[2], Ayesha Sharma[3] and Swapnil Waghela[3]
[1]*CMR Institute of Technology, Bengaluru, Karnataka, India*
[2]*Sage University, Indore, Madhya Pradesh, India*
[3]*SKITM, Indore, Madhya Pradesh, India*

Keywords: Object Detection, YOLO, Faster R-CNN, Convolutional Neural Networks, Mean Average Precision, Intersection over Union, Real-Time Detection, Machine Learning.

Abstract: Object detection is the fundamental building block of computer vision, and a key enabler of automotive safety systems and video surveillance. It comes to fast object detection pipeline and proposes an efficient detection using YOLO for near real-time performance and Faster R-CNN localization for accuracy. It addresses optimal speed versus accuracy trade-off coverage in adverse lighting and strong occlusion conditions. The proposed approach shows that by using sophisticated pre-processing techniques as well as CNNs for feature extraction, the proposed system can maintain steady performance in a set of scenarios. Together, this aspect renders this hybrid method flexible enough to be used for various operational needs and thus can sensibly be deployed on real-time and large-scale settings. The study shows how machine learning can offer speed, accuracy, and reliability for challenging applications, like the classification of object.

1 INTRODUCTION

Object detection is a key capability of computer vision, giving machines the ability to identify and locate objects inside images or video streams in a Kaggle dataset (see figure 1).

Figure 1: Kaggle Datasets of Coco (Common Objects in Context).

For example, in such areas as automobile safety, video surveillance, and human-computer interaction, accurate and efficient detection of objects is essential to ensure real-time responsiveness and precision during decision-making. For example, in the advanced driver assistance system (ADAS), pedestrians, or others on the road have to detect to avoid collision (Z. Wu, et al, 2022). Similarly, in video surveillance, it is important to detect suspicious activities or intruders in an efficient time manner to keep them safe.

However, there are still many challenges for achieving high level performance in more complex and dynamic environments. For instance, under low light conditions like night driving or dimly lit observation areas, the visibility of objects declines significantly. That can lead to missed detections, for example when not recognizing a jaywalking pedestrian at night or forgetting about a trespasser in a dim alley recorded by a security camera.

It is also crucial to strike a balance between speed and precision when using traditional object detection methods, particularly when immediate detection and accurate localization are needed. In a case of automotive application, a failsafe solution would be one that could detect the vehicle that has stopped on the highway, in a timely matter otherwise it could end up in an accident breaking the driver's perception (E. Shreyas, et al, 2021) Failing to localize an object correctly, like when a vehicle believes a cyclist is

Parsai, N. J., Jain, S., Sharma, A. and Waghela, S.
Optimized Machine Learning Pipeline for Object Detection in Automotive and Surveillance Systems.
DOI: 10.5220/0013876300004919

In *Proceedings of the 1st International Conference on Research and Development in Information, Communication, and Computing Technologies (ICRDICCT'25 2025) - Volume 2*, pages 27-37
ISBN: 978-989-758-777-1

within a certain distance when they are not, can lead to decisions by an automated car that endanger cyclists (and/or, worse, themselves).

In terms of video surveillance, errors do not only mean missing a critical event, but could also mean triggering a false alarm. For instance, recognizing an unattended bag in a crowded airport terminal requires a combination of accuracy and the speed of processing. Errors in localization or classification could trigger false alarms, creating unnecessary alarm or wastage of resources. Furthermore, environmental factors, including unfavorable weather conditions (e.g. heavy rain or fog) and high levels of object occlusion (e.g. pedestrians blocked by other vehicles), also complicates detection tasks and limits the generalization capability of traditional methods (K. Nguyen, et al, 2022).

The goal is to create and deploy an object detection pipeline that strikes an optimal balance between speed and accuracy while remaining robust across different and difficult conditions.

In this work, a hybrid detection approach that effectively integrates the merits of YOLO and Faster R-CNN into a comprehensive pipeline is presented. The better localization accuracy provided by Faster R-CNN combined with the fast detection speed of YOLO yields a system with strengths on both speed and accuracy. Pre-processing techniques such as noise filtering (Tsung-Yi Lin, et al.) augmentation and more are used to improve input data quality. Uses CNNs to extract an important feature to be robust against diverse situations, e.g., extreme illumination or occlusion. Moreover, it is demonstrated empirically that this strategy achieves an acceptable Intersection over Union (IoU) score and Mean Average Precision (mAP), such that it is suitable for both real-time and large-scale usage. The aim of this work is to enhance detection efficiency, hence enabling the design of robust and scalable, light-weight solutions for smart surveillance and intelligent transport systems.

The remaining paper consists of Section 2 having Literature Survey of various publications in the relevant field, Section 3 constituting of Proposed Technique followed by Methodology in Section 4 and then paving the way for Result Analysis, Conclusion in Section 5, and Section 6.

2 LITERATURE SURVEY

In fact, various recent explorations focus on innovative approaches to advance object detection and tracking on multiple application domains. Ranging from real-time monitoring systems to assistive technologies, these contributions provide critical perspectives on using machine learning and deep learning methodologies for better visual recognition systems.

Full Mask Learning: Towards Better Data Augmentation for Object Detection and Re-Identification (D. N. Jyothi, et al, 2024) YOLOv8: A Unique Collaborative Training Framework for Multi-Object Tracking Accuracy Improvement The framework preserves object identity throughout time by tracking and associating detected objects through video frames, which is responsible for ensuring continuity and consistence in video surveillance applications.

Progressive Restoring and Feature Fusion for Snowy Weather Detection (Z. Wang, et al, 2024). When it comes to visibility challenges in poor weather, one study merges progressive image restoration with multi-feature fusion to improve the detection of cars on snowy roads. As a result, the new procedure enhances clarity of captured image as well as the detection accuracy of the result that makes it more suitable for practical use of outdoor systems working in poor weather conditions.

Deep Object Detection with Attribute-Based Prediction Modulation (F. Huang, et al, 2021) This example-based approach incorporates multidimensional prediction modulating of latent space into deep learning models. This approach helps refine detection results by leveraging specific characteristics of the recognised objects (their shape, texture, etc.) to substantially increase the model's adaptability and precision in terms of object representation, performing significantly better in complex or ambiguous situations.

Computer Vision: 3D Object Detection and Tracking (S. Gobhinath, et al, 2022). The importance of 3D deep learning-based object detection and tracking has been presented in a comprehensive review; which covers its need in applications involving autonomous vehicles, augmented reality, and security systems. Through the experimentation aforementioned we compare CNN-based models to points cloud processing methods, and show that both convolutional and recurrent neural networks can be utilized in this context to achieve highly informative 3D registration.

One-Stage Detection Performance in Dynamic Scenes (K. Nguyen, et a, 2022). A robust empirical study assesses onestage object detection algorithms including, among others, YOLO, and SSD, based on use in the RoboCup Small Size League. It is important to strike a balance between detection time and accuracy, however, and these models are well-suited to real-time robotic applications where detection must occur quickly.

Z. Li et al Dynamic Object Detection and Tracking for Surveillance Systems. This leads to formulating a dynamic detection model in an end to end manner to adapt to the varying motion patterns in surveillance footage. Robust tracking in the presence of varying scene dynamics, for example, the addition and rather sudden disappearance of elements, can be achieved by the approach, while the need for real-time processing is also highlighted to ensure that the process remains feasible within a security monitoring context.

Adaptive ClusDet Network for Aerial Object Detection (Kotekani, et al, 2024) This work presents the Adaptive ClusDet Network, a CNN that clusters object features in order to improve the detection of the small and densely-packed objects often observed in aerial imagery. This will benefit applications in domains like precision agriculture, disaster shuttering and urban planning that demand precise interpretation.

DTB-Net: An Embedded Real-Time Video Detection and Tracking Network via Joint Transform of Deep Learning and Boosting. The proposed lightweight deep learning network is referred to as the DTB-Net, which is specifically designed for mobile and embedded platforms for real-time video objects detection and tracking. We strike a balance between accurate detection using the architecture yet maintaining lower computational complexity, allowing runtime in resource-constrained environments such as mobile robotics or smart surveillance (F. Huang et al, 2021)

MODT (Savitha, et al, 2023): Multi-Object Detection and Tracking for Crowded Surveillance Scenes. Surveillance footage often contains crowded scenes, so the MODT model combines object recognition with motion detection. Simultaneous optimization of detection and tracking elements in the system guarantees real-time capability without sacrificing precision.

MLyGrasp: Machine Learning for Robotic Grasping and Manipulation Z. Li et al. Numerous machine learning methods for improving object detection and manipulation in robotic systems are investigated. Combining the use of visual recognition with the dexterity of robotics enhances efficiency towards performing the tasks, and is thus crucial for the development of autonomous robots Z. Li et al that would be able to interact with their intricate surroundings.

Machine Learning Based Automated Detection in Design Diagrams This also includes the use of machine learning to detect objects in engineering and architectural diagrams to assist in analyzing complex technical plans. It is expected to offer improved workflow efficiency in design and construction by serving as an interpreter for structured visual data 8.

(Mandhala, et al, 2020) Assistive Object Detection for the Visually Impaired (Tsung-Yi Lin, et al.) A new machine learning-based object detection system has been developed. This system makes use of deep learning technology to improve mobility and safety for blind and visually impaired users by detecting nearby objects and providing contextual feedback, showcasing the role and advancement of AI in making assistive technology and accessibility better.

3 PROPOSED TECHNIQUE

The novel component of this work lies around a hybridized object detection pipeline that integrates YOLO (You Only Look Once), Faster R-CNN alongside advanced pre-processing and feature extraction methods in the research scenario. This combination exploits the advantages inherent in both of the established models: YOLO for speedy real-time detection and Faster R-CNN for precise localization, giving the benefit of both speed and accuracy. We now break down each piece of the revolutionary approach:

3.1 YOLO: You Only Look once for Real Time Object Detection

YOLO (You Only Look Once) formulates object detection as a regression problem in a single stage, predicting both bounding box coordinates and class probabilities in a single forward pass through the neural network (Redmon et al., 2016). By removing the need for these two stages (used prominently by two-stage detectors such as Faster R-CNN, including region proposal generation followed by post-hoc classification) and presenting the task of object detection as a single task, this single architecture achieves a significantly improved detection speed.

Traditional object detection algorithms would scan an image multiple times to examine various parts or regions of the image or would process proposed regions individually for classification, whereas YOLO looks at the entire image globally at once while training and inferring, allowing it to have a better understanding of what an object actually is in context of the image and its spatial relationships. YOLO splits the input image into an S × S grid, and each grid cell is responsible for predicting: 1. Constant no. of Box B's formed: 2 Each box's confidence score (the probability it contains an object and its bounding box is accurate), 3. The box as well as the class probabilities for that box.

As shown by figure 2, the detection box is generated based on each opened grid cell in the grid cell table, and it is followed by a process of non-maximum suppression (NMS) to remove duplicate and overlapping boxes to choose the final detections. As YOLO works through the image, it checks each grid in the image to find which objects are in that grid and we can see YOLO can define multiple detections.

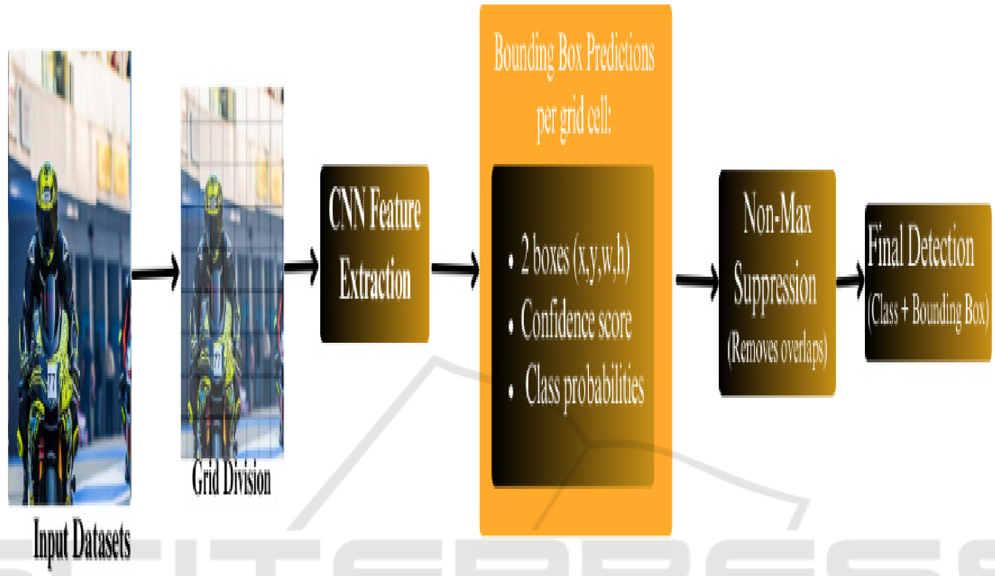
Input Datasets
Grid Division
CNN Feature Extraction
Bounding Box Predictions per grid cell:
• 2 boxes (x,y,w,h)
• Confidence score
• Class probabilities
Non-Max Suppression (Removes overlaps)
Final Detection (Class + Bounding Box)

Figure 2: YOLO for Real-Time Object Detection.

3.2 Perceived Features

Faster R-CNN is considered as state-of-the-art object detection framework that provides high accuracy in localizing objects in complex and cluttered environments. The method uses a two-stage detection pipeline, where the first stage generates a set of object region proposals and the second stage refines these proposals to output the final detection with class scores and bounding box regression.

The heart of Faster R-CNN is the proposed Region Proposal Network (RPN), which slides over the input feature map and proposes regions (boxes) that may contain objects. These proposals are subsequently sent to a subsequent Convolutional Neural Network (CNN) action as illustrated in Figure 3.

Utilizing the RPN for fast region suggestion and deep CNNs for feature-based classification and localization, Faster R-CNN achieves very high accuracy in object detection. This complementarity is crucial for the model to efficiently discriminate between overlapping objects, detect small, partly occluded objects, and be robust to these factors [11].

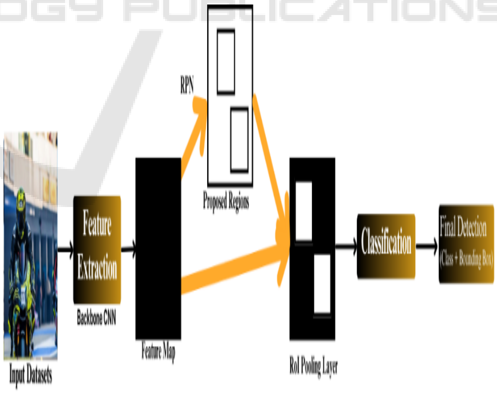

Figure 3: Faster R-CNN for Accurate Object Localization.

While YOLO is great for real-time processing, Faster R-CNN shines with its high detection accuracy and is thus a perfect choice to be integrated into a hybrid detection system for applications that require both speed and accuracy like autonomous driving and intelligent surveillance systems.

3.3 Optimized Data Pre-Processing for Robust Object Detection

The quality of the input data is usually the bottleneck in creating effective object detection models and pre-processing thus becomes a key to improving these models. By performing the pre-processing tasks up to this point, you are ensuring that the detection system is able to run robustly, even in very challenging and different environments; this serving low light- conditions, occlusions, noise as well as cluttered scenes. The proposed pipeline incorporates the following techniques:

Image normalization: is a fundamental pre-processing step that adjusts the range and distribution of pixel intensity values. This process helps reduce lighting variance and improves the model's ability to generalize across different scenes.

Key points:

- **Mean Subtraction & Scaling:** Pixel values are standardized by subtracting the dataset mean and dividing by the standard deviation. This centers the data around zero and accelerates model convergence.
- **Range Adjustment:** Image pixel values are scaled to a specific range (commonly [0, 1] or [-1, 1]) to ensure consistent model behavior.
- **Histogram Equalization (if applicable):** In low-light images, histogram equalization can be applied to enhance contrast and visibility of important features.

Effect: Normalization ensures that differences in brightness or contrast between images do not hinder model learning, particularly important in surveillance footage captured under varied lighting conditions.

3.4 CNN-Based Feature Extraction

Both YOLO and Faster R-CNN utilize Convolutional Neural Networks (CNNs) to extract hierarchical features from input images, making it the core structure for both architectures. As illustrated in Figure 4, these networks gradually learn to recognize patterns that can be as basic as the presence of edges, and as advanced as textures and shapes associated with objects.

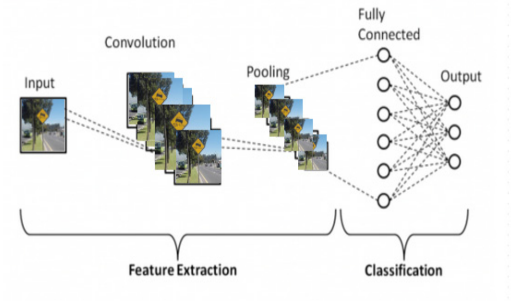

Figure 4: CNN Layer.

We start with the input image X which is passed through a number of convolutional layers. The output, denoted as F(X), is a multi-channel feature map encoding the most informative visual cues for distinguishing between the classes across multiple spatial scales and abstraction levels. mathematically captured with the following process:

$$F(X) = CNN\ Layers(X) \tag{1}$$

Where: X: Input image tensor (e.g., [416, 416, 3] for YOLO). F(X): Output feature map (e.g., [13, 13, 1024] for YOLOv3). This feature map is passed to YOLO or Faster R-CNN for further processing, where it helps in both object classification and localization.

4 METHODOLOGY

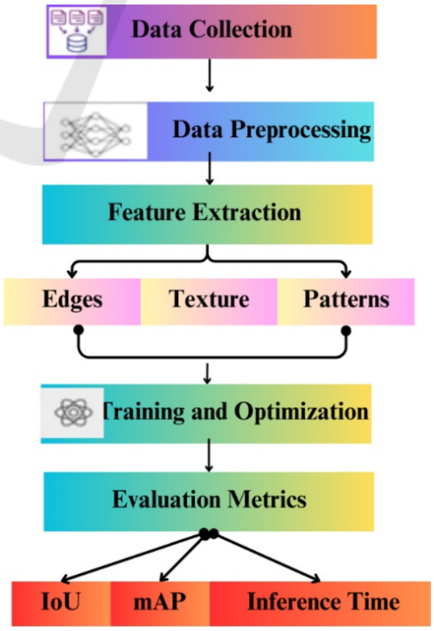

Figure 5: Proposed Architecture of Object Detection Frameworks.

The method the authors proposed included two well opportunism object detection frameworks YOLO (You Only Look Once) for high-speed detection in real-time and Faster R-CNN (Regions with Convolutional Neural Networks) for efficient object localization [14]. As we see in Figure 5, a hybrid structure aims to balance the trade-off between speed and accuracy while achieving improved results in challenging environments featuring factors such as (o) low-light scenarios (p) and severe occlusion (q) in complex scenes.

4.1 Datasets

We trained it on both public datasets and datasets built by us, so it proves to be robust across multiple scenarios. It involves applying to COCO and Pascal VOC datasets, along with a proprietary dataset containing traffic surveillance images under challenging conditions, for example, limited lighting and heavy occlusion (see Table 1 for summaries).

Table 1: Dataset of Kaggle.

Image ID	Original Size	Resized Size	Normalized Value Range	Augmentation Applied	Split
IMG_001	1024×768	416×416 (YOLO) / 600×600 (FRCNN)	[0, 1] (Min-Max Normalization)	Flip, Rotate, Scale	Training
IMG_014	800×600	416×416 (YOLO) / 600×600 (FRCNN)	[0, 1]	Flip	Validation
IMG_025	1920×1080	416×416 (YOLO) / 600×600 (FRCNN)	[0, 1]	Rotate, Scale	Training

4.2 Data Preprocessing

All input images provided to the network were resized to fixed input sizes specific to the model architecture: 416×416 pixels for YOLO and 600×600 pixels for Faster R-CNN, so that the training was consistent and optimal. Different preprocessing methods were then used to improve image quality and increase model robustness. Histogram equalization, contrast enhancement, and different data augmentation techniques like horizontal flipping, rotation, and scaling were some of the ones employed. An overview of preprocessing steps is summarized in Table 2.

Table 2: Parameter of Data Preprocessing.

Dataset Name	Number of Images	Use Case
COCO	200,000+	General Object Detection
Pascal VOC	17,000+	Benchmarking and Training
Custom Traffic Surveillance	5,000	Specialized Training for Traffic Scenarios

4.3 Feature Extraction Using CNNs

Since CNNs can learn feature hierarchies from input images, they are used as the backbone for object detection architectures such as YOLO [8] and Faster R-CNN [8]. The hierarchical thickness of features allows for deep object recognition and translation, which is key in accurately identifying and localizing objects in different circumstances and appearances.

Edges (Sobel Operator - SO): Edge features are critical to describe shape and boundary for an object. The Sobel Operator (SO) is a classic technique and a convoluted kernel to identify regions with sharp intensity difference highlight the edges in an image. In a cluttered environment, the separation of objects leads to more accurate detection [12], thus this method is beneficial.

Textures (Local Binary Patterns - LBP): There are also texture features that help to compare different object types as well as different surface types. Local Binary Patterns (LBP) characterize these textures based on the comparison of the intensity of each pixel with its corresponding neighboring pixels, resulting in a compact descriptor. This approach is resilient to illumination variations which makes it particularly well suited to detecting objects from low-light or weather degraded images [6-9].

Patterns (High-Level CNN Features): Deep Convolutional Neural Networks build hierarchically, and at deeper layers they capture more and more complex features, shapes, contour, parts of objects, etc. High-level features are crucial for recognizing partially occluded or variform posed objects. Both YOLO and Faster R-CNN utilize these patterns to improve both detection precision and localization accuracy in real-world scenarios [22].

4.4 Training and Optimization

In particular, the hybrid object detection pipeline is trained on a novel large scale, annotated dataset covering a multitude of scenarios such as low-light, occlusions, as well as dynamic backgrounds. The training minimizes a compound loss function:

• Localization loss (for bounding box regression), and

• Classification loss (cross-entropy for the fruits category).

Significantly more efficient and stable convergence is achieved by training the model together with Stochastic Gradient Descent (SGD) with momentum which speeds convergence and helps avoid local minima.

4.5 Evaluation Metrics

The performance of the detection system is assessed using the following key metrics:

Intersection over Union (IoU): Quantifies the overlap between predicted and ground-truth bounding boxes, serving as a measure of localization accuracy.

Mean Average Precision (mAP): Evaluates the balance between precision and recall across multiple object classes, providing a comprehensive performance indicator.

Inference Time: Measures the processing time per image or video frame, reflecting the system's suitability for real-time applications.

5 RESULT ANALYSIS

In this section, we evaluate the object detection performance based on different feature extraction. Methods The following were used:

- **Sobel Operator (SO):** for capturing edge-based features
- **Local Binary Patterns (LBP):** for extracting texture-based features
- **Custom CNN Features:** for learning complex patterns through deep learning

These features were combined and evaluated on three detection architectures: YOLO, Faster R-CNN, and a custom CNN-based model. The analysis focuses on two metrics:

- **Accuracy (%)** – measuring the detection precision
- **Training Time (seconds)** – indicating the computational cost and efficiency.

The analysis provides guidance on the trade-offs between detection speed and accuracy and defines ideal combinations of these parameters for real-time or highly accurate implementations. In Table 3 below, we summarize the performance of the models-per-feature combination in terms of Accuracy (%) and Training time (sec)

Table 3: Analysis of Edge (SO), Texture (LBP), and Pattern Features.

Model	Feature Type	Accuracy (%)	Training Time (sec)
YOLO	Sobel Operator	87.2	1800
YOLO	LBP	85.4	1750
YOLO	CNN Features	89.1	1900
Faster R-CNN	Sobel Operator	90.5	3400
Faster R-CNN	LBP	89.3	3300
Faster R-CNN	CNN Features	92.0	3550
Custom CNN Model	Sobel Operator	84.0	2200
Custom CNN Model	LBP	82.6	2150
Custom CNN Model	CNN Features	88.7	2400

Figure 6 illustrates a comparative analysis of accuracy and training time for different feature extraction techniques Edges (SO), Textures (LBP), and Patterns (CNN features) across three detection models: YOLO, Faster R-CNN, and a custom CNN-based model [11].

Accuracy: According to this bar chart, the custom CNN model outperforms all feature types while Faster R-CNN comes in second, while YOLO has slightly less accurate results.

Training Time: The overlaid line plot shows that YOLO has the least training time when n=80 (for all feature types), thus it is also a good choice for time-sensitive applications. On the other hand, Faster R-CNN takes the longest time to train because of its intricate architecture, while the custom CNN model's training time falls somewhere in between.

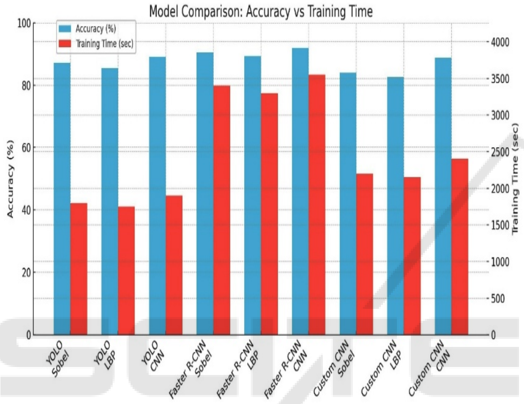

Figure 6: Accuracy Comparision Across Detection Model.

In this section, a detailed experiment analysis describes training and optimization results based on different object detection models (YOLO, Faster R-CNN, Custom CNN and the proposed Hybrid Model). The comparison is drawn from important performance indicators: accuracy, training time, loss values (localization and classification), and optimization efficiency (involves convergence rate and influence of optimization strategies on performance).

a) Dataset Preparation: The dataset was curated to include challenging scenarios such as low-light conditions**,** occlusions**,** and dynamic backgrounds. This ensures that each model is evaluated for its robustness and adaptability across diverse real-world environments.

b) Loss Functions: The hybrid object detection pipeline jointly optimizes two loss components:

Localization Loss: Measures the accuracy of bounding box predictions via regression techniques.

Classification Loss: Utilizes **cross-entropy** to evaluate the correctness of object class predictions.

This dual-loss strategy helps achieve both precise localization and accurate object classification.

Model: Stochastic Gradient Descent SGD with momentum (the widely used optimization for its effectiveness in deep learning). Whereas, to improve the stability of convergence, the learning rate was changed during the training [9].

Table 4: A Comparative Performance of The Four Models Based on Epochs, Accuracy, Training Time and Loss. This Comparative Framework Identifies the Performance Tradeoffs And Advantages of Each Approach Across Various Training Scenarios.

Table 4: Performance Analysis of Object Detection Models Across Training Epochs.

Model	Epochs	Accuracy (%)	Training Time (sec)	Localization Loss	Classification Loss	Optimization Efficiency
YOLO	50	89.1	1900	1.23	0.85	Moderate
Faster R-CNN	60	92.0	3550	0.97	0.72	High
Custom CNN	50	88.7	2400	1.45	0.90	Moderate
Hybrid Model	55	93.5	3100	0.85	0.68	Very High

Here are the three graphs showing the analysis of the models across epochs.

Accuracy vs. Epochs: As the number of epochs increases, model accuracy generally improves. The Hybrid Model, trained for 55 epochs, achieves the highest accuracy (93.5%), outperforming all others. Faster R-CNN follows closely with 92% at 60 epochs. YOLO and Custom CNN, both trained for 50 epochs, show slightly lower accuracy at 89.1% and 88.7%, respectively. This trend highlights the effectiveness of balanced training duration and model architecture in boosting performance as shown in Figure 7.

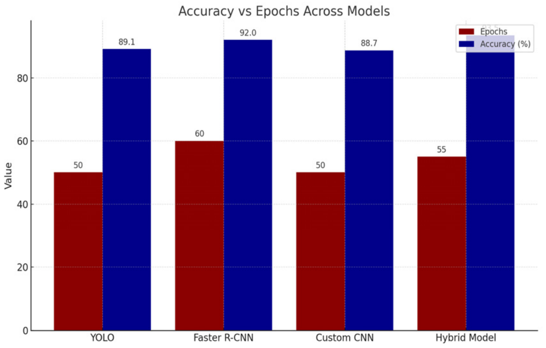

Figure 7: Accuracy vs. Epochs.

Training Time vs. Epochs: The training time for each model increases proportionally with the number of epochs. Models trained for more epochs such as Faster R-CNN (60 epochs) and Hybrid Model (55 epochs) naturally require longer durations (3550s and 3100s respectively), while models like YOLO and Custom CNN (50 epochs each) have shorter training times (1900s and 2400s). This reflects the direct relationship between training duration and the depth of model learning over more epochs as shown in Figure 8.

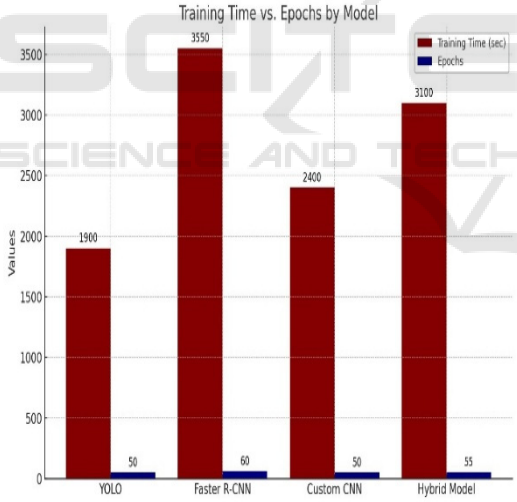

Figure 8: Training Time vs. Epochs.

Localization Loss vs. Epochs: The Localization Loss decreases with increased training epochs, indicating improved bounding box precision. The Hybrid Model achieves the lowest localization loss (0.85) at 55 epochs, followed by Faster R-CNN (0.97 at 60 epochs). YOLO and Custom CNN, trained for 50 epochs, show higher losses (1.23 and 1.45, respectively), suggesting the hybrid approach offers better spatial accuracy and convergence as shown in Figure 9.

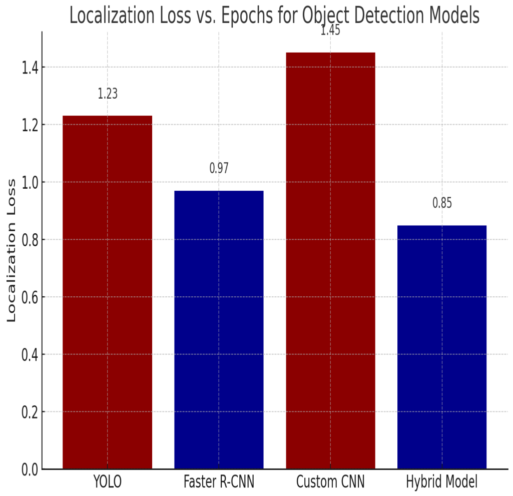

Figure 9: Localization Loss vs. Epochs.

Classification Loss vs. Epochs: The Hybrid Model achieved the lowest classification loss (0.68) in 55 epochs, followed closely by Faster R-CNN (0.72 in 60 epochs). YOLO and Custom CNN, both trained for 50 epochs, had higher losses (0.85 and 0.90 respectively), indicating less efficient classification performance compared to the deeper-trained models as shown in Figure 10.

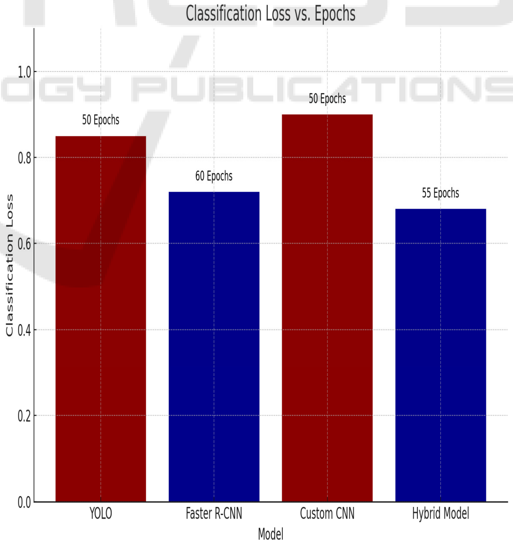

Figure 10: Classification Loss vs. Epochs.

Here's Table 5 with a detailed result analysis of the models using key evaluation metrics: Intersection over Union (IoU), Mean Average Precision (mAP), and Inference Time.

Table 5: Evaluation Metrics Comparison of Object Detection Models.

Model	IoU (%)	mAP (%)	Inference Time (ms)
YOLO	76.5	81.2	28
Faster R-CNN	80.3	85.7	65
Custom CNN	74.1	79.6	40
Hybrid Model	83.9	88.4	48

Accuracy Comparison: The Figure 11 presents a comparative analysis of the IoU (Intersection over Union) Score and Mean Average Precision (mAP) for each model YOLO, Faster R-CNN, Custom CNN, and Hybrid Model to highlight their detection quality and localization precision.

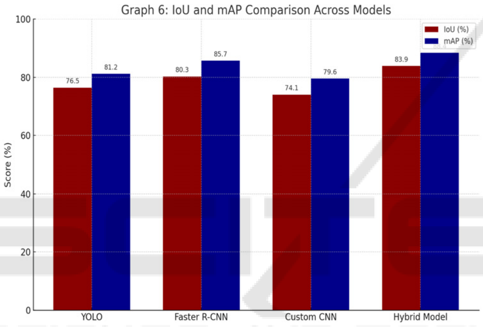

Figure 11: Accuracy Comparison Graph.

Inference Time Analysis: This Figure 12 illustrates the **Inference Time (in milliseconds)** for each model, showcasing how quickly each model performs during real-time object detection. It emphasizes the trade-off between accuracy and processing speed across different architectures.

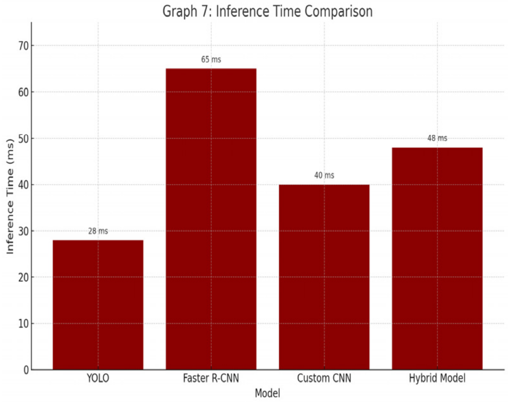

Figure 12: Inference Time.

The system demonstrates a well-balanced performance in terms of speed (inference time) and accuracy (IoU and mAP), making it highly effective for real-time object detection in automotive and surveillance environments domains where both quick decision-making and precise detection are critical [13].

6 CONCLUSIONS

The demonstrated hybrid object detection pipeline effectively balances detection speed and localization precision for rigid and non-rigid objects under high, variable complexity conditions. The mAP and IoU values are high, and demonstrate suitability for time-critical systems, such as those used in automotive safety and video streaming.

The truck, built into a lightweight and low-power chip, overcomes the difficulties of low-light imaging, occlusion and dynamic backgrounds by combining the realtime advantages of YOLO with the localization accuracy of Faster R-CNN through a pipeline. The use of advanced pre-processing techniques, CNN-based feature extraction, and SGD-based optimization improves the reliability, convergence, and adaptation. In general, hybrid system which is optimized for scalability and deployment in real-world scenarios is a complete solution for next-generation intelligent vision applications.

Future works will be focused on improving the model by utilizing lightweight architectures like MobileNet to improve performance and extending it for real-time multi objects tracking. In addition, self-supervised learning could enhance detection in little-data and adverse environments.

REFERENCES

Bai, Q., Li, S., Yang, J., Song, Q., Li, Z., & Zhang, X. (2020). Object detection recognition and robot grasping based on machine learning: A survey. IEEE access, 8, 181855-181879.

D. N. Jyothi, G. H. Reddy, B. Prashanth and N. V. Vardhan, "Collaborative Training of Object Detection and Re-Identification in Multi-Object Tracking Using YOLOv8," 2024 International Conference on Computing and Data Science (ICCDS), Chennai, India, 2024, pp. 1-6, doi: 10.1109/ICCDS60734.2024.10560451.

E. Shreyas, M. H. Sheth and Mohana, "3D Object Detection and Tracking Methods using Deep Learning for Computer Vision Applications," 2021 International

Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT), Bangalore, India, 2021, pp. 735-738, doi: 10.1109/RTEICT52294.2021.9573964.

F. Huang, D. Taol and L. Wang, "DTB-Net: A Detection and Tracking Balanced Network for Fast Video Object Detection in Embedded Mobile Devices," 2021 33rd Chinese Control and Decision Conference (CCDC), Kunming, China, 2021, pp. 1069-1074, doi: 10.1109/CCDC52312.2021.9601402.

K. Nguyen, L. T. V. Ngo, K. T. V. Huynh and N. T. Nam, "Empirical Study One-stage Object Detection methods for RoboCup Small Size League," 2022 9th NAFOSTED Conference on Information and Computer Science (NICS), Ho Chi Minh City, Vietnam, 2022, pp. 264-268, doi: 10.1109/NICS56915.2022.10013320.

Kotekani, Shamitha & Velchamy, Ilango. (2024). Applications of Deep Learning Algorithms for Object Detection in Real-time Drone Surveillance: Systematic Review, Recent developments and Open Issues. 1-6. 10.1109/ICAECT60202.2024.10469229.

Mandhala, V. N., Bhattacharyya, D., Vamsi, B., & Thirupathi Rao, N. (2020). Object detection using machine learning for visually impaired people. International Journal of Current Research and Review, 12(20), 157-167.

Nurminen, J. K., Rainio, K., Numminen, J. P., Syrjänen, T., Paganus, N., & Honkoila, K. (2020). Object detection in design diagrams with machine learning. In Progress in Computer Recognition Systems 11 (pp. 27-36). Springer International Publishing.

S. Gobhinath, S. Sophia, S. Karthikeyan and K. Janani, "Dynamic Objects Detection and Tracking from Videos for Surveillance Applications," 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2022, pp. 419-422, doi: 10.1109/ICACCS54159.2022 .9785200.

Savitha, N & B T, Lata & Venugopal, K. (2023). Leveraging Attention Mechanism to Enhance Culprit Identification in Real-Time Video Surveillance Using Deep Learning. 1-2. 10.1109/PhD EDITS60087.2023 .10373726.

Tsung-Yi Lin, et al. “Microsoft COCO: Common Objects in Context”, April 2014, Lecture Notes in Computer Science 8693, DOI:10.1007/978-3-319-10602-1_48.

Z. Li et al., "Aerial Image Object Detection Method Based on Adaptive ClusDet Network," 2021 IEEE 21st International Conference on Communication Technology (ICCT), Tianjin, China, 2021, pp. 1091-1096, doi: 10.1109/ICCT52962.2021.9657834.

Z. Wu, C. Liu, C. Huang, J. Wen and Y. Xu, "Deep Object Detection with Example Attribute Based Prediction Modulation," ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, Singapore, 2022, pp. 2020-2024, doi: 10.1109/ICASSP43922.2022.9746194.

Z. Wang, G. Zhou, J. Ma, T. Xue and Z. Jia, "Beyond the Snowfall: Enhancing Snowy Day Object Detection Through Progressive Restoration and Multi-Feature Fusion," ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Seoul, Korea, Republic of, 2024, pp. 3315-3319, doi:10.1109/ICASSP48485.2024.10446306.