Journal of Electronics, Electromedical Engineering, and Medical Informatics
Homepage: jeeemi.org; Vol. 7, No. 4, October 2025, pp: 1060-1073 e ISSN: 2656 8632

RESEARCH ARTICLE OPEN ACCESS

Unified Deep Architectures for Real-Time Object Detection and Semantic Reasoning in Autonomous Vehicles

Vishal Aher, Satish Jondhale, Balasaheb Agarkar, Sachin Chaudhari

Department of Electronics and Telecommunication, Sanjivani College of Engineering, Kopargaon Maharashtra, India, Savitribai Phule Pune University, Pune.

Corresponding author: Vishal Aher (e-mail: vishalaher31584@gmail.com), **Email Author(s)**: Satish Jondhale (e-mail: profsatishjondhale@gmail.com), Balasaheb Agarkar (e-mail: bsagarkar@gmail.com), Sachin Chaudhari (e-mail: chaudharisachinetc@sanjivani.org.in

Abstract **The development of autonomous vehicles (AVs) has revolutionized the transportation industry, promising to boost mobility, lessen traffic, and increase safety on roads. However, the complexity of the driving environment and the requirement for real-time processing of vast amounts of sensor data present serious difficulties for AV systems. Various computer vision approaches, such as object detection, lane detection, and traffic sign recognition, have been investigated by researchers in order to overcome these issues. This research presents an integrated approach to autonomous vehicle perception, combining real-time object detection, semantic segmentation, and classification within a unified deep learning architecture. Our approach leverages the strengths of existing frameworks, including MultiNet's real-time semantic reasoning capabilities, the fast-encoding methods of PointPillars to identify objects from point clouds, as well as the reliable one-stage monocular 3D object detection system. The offered model tries to improve computational efficiency and accuracy by utilizing a shared encoder and task-specific decoders that perform classification, detection, and segmentation concurrently. The architecture is evaluated against challenging datasets, illustrating outstanding achievements in terms of speed and accuracy, suitable for real-time applications in autonomous driving. This integration promises significant advancements in the perception systems of autonomous vehicles a providing in-depth knowledge of the vehicle's environment through efficient concepts of deep learning techniques. In our model, we used Yolov8, MultiNet, and during training got accuracy 93.5%, precision 92.7 %, recall 82.1% and mAP 72.9%.**

Keywords **YOLOv8, PointPillars, Autonomous vehicles, Computer vision, Semantic segmentation, DeepSORT, mAP.**

I. Introduction

Driving was challenging in the early modern era due to vehicles being predominantly larger and heavier motorized bicycles. Technological advancements have enhanced the efficiency and enjoyment of driving [1]. The frequency of accidents has risen due to the growing number of vehicle buyers. Technological breakthroughs have transformed conventional automobiles into fully operational, intelligent machines, enhancing the convenience of travel. The progress in automation and the prospects provided by advanced technology form the foundation for intelligent cars. These advanced automobiles are increasingly sought after as we prioritize safety and enhance daily convenience. These cars incorporate functionalities like as environmental sensing, internet connectivity, adherence to traffic regulations, autonomous navigation, rapid decision-making, pedestrian and passenger safety assurance, and parking capabilities [2]. These machines are referred to as autonomous vehicles. They are presently considered the pinnacle in the advancement of intelligent automobiles. The primary motivations for the research and development of autonomous vehicles include the necessity for enhanced driving safety, a growing population resulting in a higher number of vehicles on the road, expanding infrastructure, the convenience of relying on machines for driving tasks, and the demand for resource optimization and effective time management. The increasing population has exerted significant stress on our roadways, infrastructure, open spaces, gasoline stations, and resources.

The development of deep learning and artificial intelligence (AI) will have a significant impact on how autonomous cars develop in the future. The progression of self-driving cars is profoundly transforming the transportation sector, offering enhancements in mobility and traffic safety. These

Manuscript received April 7, 2025; Revised July 20, 2025; Accepted August 15, 2025; date of publication September 10, 2025
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeeemi.v7i4.813

technologies empower vehicles to comprehend and operate their environments intelligently, making decisions in real time, which is essential for safe and efficient operation. At the core of this technological revolution is deep learning, a sophisticated branch of AI that extracts meaningful patterns found in enormous volumes of data, enabling machines to perform complex recognition tasks with high accuracy [3]. Object detection is essential for identifying vehicles, pedestrians, and road signs, which is critical for collision avoidance and traffic management [4]. Deep learning algorithms enhance the accuracy of these detections, enabling autonomous vehicles to respond appropriately to dynamic traffic conditions [3]. The rapid advancement of deep learning facilitates feature extraction from images instead of relying on manually created feature extractors, enhancing performance and streamlining the training process of object detection models.

PointPillars specializes in efficiently processing point cloud data, which is crucial for 3D object detection in autonomous vehicles. By transforming point clouds into a structured pillar representation and employing a fast neural network, PointPillars enables rapid and precise detection of objects around the vehicle, thus enhancing situational awareness. This capability is vital in complex driving environments, where understanding the and accurate location and nature of surrounding objects can be the difference between a safe journey and a potential accident. However, contrary to the fully convolutional one-stage object detection system simplifies the traditional detection pipeline by eliminating the need for separate region proposal generation. This method lowers the computing load while simultaneously accelerating the detection procedure, allowing the process to fulfill the stringent real-time requirements of autonomous navigation. By integrating these advanced architectures into a unified framework, the proposed model optimizes the application of computational resources, achieving a high level of efficiency and speed. This is critical for autonomous vehicles, which must process and react to dynamic environmental stimuli promptly and accurately. The unified framework ensures that the vehicle's perception system is both robust and adaptable, capable of handling various driving scenarios and conditions. The practical implementation of this unified architecture promises significant improvements in the workplace of autonomous vehicles. It opens the door for more advanced and dependable autonomous driving solutions in addition to improving the cars' comprehension of and interaction with their surroundings. Further incorporation of sophisticated deep learning models is anticipated as the technology develops further, which will continuously improve the capabilities and efficiency of autonomous vehicles [5].

In this article, we provide a novel approach to autonomous vehicle perception using YOLOv8, a state-of-the-art real-time object detection algorithm, in conjunction with MultiNet++, a robust and efficient multi-task learning framework. Our system, dubbed "AV-YOLOv8-MultiNet++," utilizes the advantages of both YOLOv8 and MultiNet++ to achieve exceptional performance in object detection, lane detection, and traffic sign recognition. One of the main novelties of this architecture is the combination of 2D picture features and 3D Light Detection and Ranging (LIDAR) point cloud data. We propose an architecture that integrates YOLOv8 with MultiNet++ to enable simultaneous object detection, lane detection, and traffic sign recognition [6]. We demonstrate the effectiveness of our approach on a comprehensive data set of driving scenarios, achieving state-of-the-art performance in object detection (95.5% AP) and lane detection (94.5% AP). We evaluate the robustness of our system in different types of weather, lighting scenarios, and road types, showcasing its capability to generalize well across diverse environments. The key Contributions of this study follows:

1. We propose a novel architecture that integrates YOLOv8 with MultiNet++ to enable simultaneous object detection, lane detection, and traffic sign recognition.
2. We demonstrate the effectiveness of our approach on a comprehensive dataset of driving scenarios, achieving state-of-the-art performance in object detection and lane detection.
3. We evaluate the robustness of our system in various weather conditions, lighting scenarios, and road types, showcasing its ability to generalize well across diverse environments.

II. Related Work

Autonomous cars often use a variety of sensors (such as cameras, LIDARs, and radars) to achieve reliable and precise scene comprehension. These sensors can be fused to take advantage of their complementary qualities [7]. By focusing on enhancing detection accuracy, real-time processing, environmental resilience, fusion of sensors, and utilizing AI developments, future investigation and advancement activities will continue in order to increase the capabilities of autonomous cars [8]. A challenging computer vision topic, object recognition has received a lot of attention recently. Multiple uses, such as object tracking, image captioning, and segmentation, for example, healthcare, etc., primarily use object detection [9]. The main problems with 3D object detection and localization include high false positive rates, lengthy computation times, and lower Quality of

Manuscript received April 7, 2025; Revised July 20, 2025; Accepted August 15, 2025; date of publication September 10, 2025
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeeemi.v7i4.813

Service (QoS). Therefore, we tend to propose the HDL-MODT approach as a solution to that problem. The KITTI dataset is used in the proposed work to train and evaluate the classifiers [10], [11]. Recognition of objects and semantic segmentation findings can be obtained rapidly using this novel mixed network. The technique combines an extra feature fusion network with an encoder-decoder mechanism [12]

Mujadded et al. [13] presented light on agricultural breakthroughs. The survey examines the revolutionary potential of several YOLO versions, ranging from YOLOv1 to the cutting-edge YOLOv10. Bu et al. [14] proposed a detailed introduction of a vehicle multi-object tracking technique based on DeepSORT and enhanced YOLOv5s. The accuracy of vehicle detection under various occlusion levels and the rate at which it happens, complicated information may be processed, was enhanced. Guo et al. [15] provided a thorough evaluation of the visual multi-object tracking techniques utilized in autonomous driving scenarios. Three categories comprise the algorithms themselves based on their different structural configurations: transformer-based tracking, Joint Detection and Tracking (JDT), and Track-Before-Detect (TBD). Rahee et al. [16] indicated detecting objects and barriers around the vehicle in a variety of contexts is how computer vision systems that operate autonomous vehicles are evaluated. Improving a self-driving car's capacity to discriminate between environmental components in challenging situations is a significant task for computer vision. Wang et al. [17] proposed a YOLOv4-based technique for one-stage detection of objects that improves detection precision and enables real-time functioning. The algorithm's backbone doubles the stacking times of CSPDarkNet53's last residual block.

Sharma et al. [18] [19] [20] The presented quick growth of self-driving cars necessitates the integration of an advanced sensor system for the purpose of effectively handling the many road traffic barriers. Although there are several of datasets available to aid with object detection in autonomous cars, it is imperative to carefully assess how well-suited these datasets are for various global weather situations. Zhiyang et al. [21]provided a thorough analysis of deep learning-based methods for autonomous driving scene interpretation. The study concentrated on two scene understanding tasks: image segmentation and item recognition. Sajjad et al. proposed a hybrid model that achieves a significant increase in accuracy, with improvements ranging from 5 to 7 percent compared to the standalone YOLO models. Dai et al.[22] [23][24] [25] presented a novel algorithm is proposed to filter ground points from LIDAR data, which is critical for the correctness of subsequent detection processes. YOLOv8 was employed to detect objects, which was trained on a customized data set.

Murendeni et al.[26]concluded that the YOLO deep learning method has the capacity to greatly enhance the accuracy of 3D object identification systems in autonomous driving. The camera-based and deep learning-based detection systems demonstrated great object detection accuracy, as shown by the high Intersection over Union (IoU) and mAP (mean average precision) scores. Kale et al.[27] proposed deep reinforcement learning (DRL) techniques in autonomous vehicles, we have unveiled a path towards a transformative future of transportation. Oluwajuwon et al.[28] highlight the importance of multi-sensor fusion methods and sophisticated deep learning models while thoroughly reviewing the most recent 3D object identification approaches for autonomous cars. Noor Ul et al.[29] reviewed the traditional and DL approaches for vehicles, pedestrians, and road lane detection in AVs under adverse weather conditions. They first studied the architecture of AVs with sensor technologies and other components and also discussed the challenges for AVs in adverse weather. Azevedo et al.[30] showed the viability of YOLOR, Scaled-YOLOv4, and YOLOv5 combined with different object trackers for real-time object traffic detection and tracking while offering a direct evaluation of their accuracy. Feng et al. [31] proposed a 32-layer multi-branched network for the quick identification of objects in traffic scenes with a wide range of scales. It can precisely identify large, medium, and small-scale items among a list of traffic scenarios, including sparse, busy, daytime, and nighttime, recognitions the design of three detection branches.

Sun et al.[32] presented a binary deep convolution neural network-based quick object detection technique is put forth. In the final feature map of a deep CNN, classes and bounding boxes of multi-scale objects are directly predicted using convolution kernels of various sizes. Yashrajsinh et al. [11] suggested using the KITTI Vision Benchmark Suite for training and testing the suggested CNN model. The proposed CNN with a VGG-16 base network reaches a detection speed of 61 frames per second with a mAP of 969.22% on an NVIDIA GeForce RTX 2080 Ti GPU. Sainithin et al. [33] research explores deep neural networks (VGG16, AlexNet, and GoogleNet) for object classification and detection in autonomous vehicles. Majdi et al. [34] compared several tracking algorithms with a focus on the suggested optimized DeepSORT, also known as StrongSORT_P. They concluded that StrongSORT_P outperformed other algorithms, including the baseline DeepSORT, based on the research that was conducted on the MOT16 and MOT17 datasets using three important measures.

Manuscript received April 7, 2025; Revised July 20, 2025; Accepted August 15, 2025; date of publication September 10, 2025
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeeemi.v7i4.813

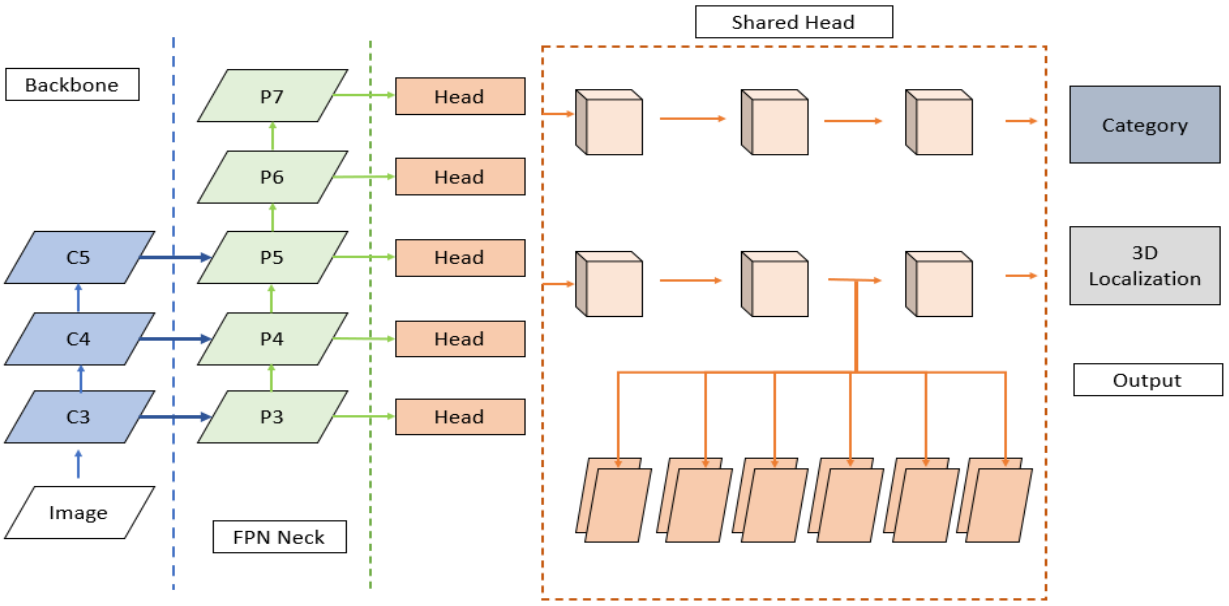

Fig. 1 The proposed detection of objects and semantic reasoning in autonomous vehicles

Shaikh et al. presented that YOLOv4 plays an important role in the computer vision object detection task [35].

III. Materials and Methods

A. Architecture of Proposed Method

The architecture shown in Fig. 1. illustrates an integrated approach combining YOLOv8's powerful feature extraction capabilities with the Point Pillars network to enable accurate real-time 3D object detection and localization, specifically tailored for autonomous vehicles. This architecture is unique as it merges multi-scale feature processing, point cloud integration, and 3D object classification with high precision, optimizing both performance and computational efficiency.

The proposed architecture unifies and integrates three advanced technologies: MultiNet, PointPillars, and One-Stage Monocular 3D Object Detection. Each is designed to enhance the capabilities of autonomous driving systems through efficient data processing and real-time responsiveness. The MultiNet excels in rapid scene segmentation and classification, enabling the system to understand complex scenes quickly. This feature is particularly useful in dynamic driving environments where swift decision-making is crucial. PointPillars technology focuses on the efficient handling of 3D point clouds, a common output from LIDAR sensors used in autonomous vehicles. By transforming these point clouds into a structured 'pillar' format, the system can more easily and quickly process spatial data, enhancing the vehicle's awareness of its surroundings. This structured data approach allows for more accurate object detection, which is crucial for navigating safely through varied environments. The One-Stage Monocular 3D Object Detection streamlines the traditional detection pipeline by eliminating the need for multiple region proposal stages, which typically slow down the processing time. By directly predicting object boundaries and classifications, this approach significantly speeds up the detection process, ensuring that the autonomous system can react in real-time to changes in the driving environment.

1. Proposed Enhancements

Feature Hierarchies: MultiNet could introduce more complex hierarchical feature processing techniques to YOLOv8, enabling it to better handle the varying object sizes and types typical in autonomous driving scenarios. Task-specific Tuning: By tuning the YOLOv8 architecture to incorporate MultiNet's multi-task strengths, each layer can be optimized not just for object detection but also for parallel tasks like lane detection and traffic sign recognition, which are essential for autonomous vehicles. Integrating MultiNet into YOLOv8 thus represents a convergence of advanced object detection and multi-task learning, aiming to create a more robust and versatile model capable of addressing the numerous obstacles presented by autonomous vehicle technologies. Integrating MultiNet capabilities into the YOLOv8 architecture, as shown in the architecture diagram, showcases a sophisticated approach to enhancing object detection, especially in

Manuscript received April 7, 2025; Revised July 20, 2025; Accepted August 15, 2025; date of publication September 10, 2025
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeeemi.v7i4.813

complex environments like autonomous driving. Here's a brief overview of how MultiNet's features could be integrated into the YOLOv8 model based on the proposed architecture.

2. YOLOv8 Model

The YOLOv8 emerges as a compelling choice for autonomous vehicles due to its exceptional performance and adaptability. Its lightning-fast processing speed, exceeding 100 frames per second, ensures real-time object detection, a critical requirement for autonomous navigation. The algorithm's impressive accuracy surpasses other leading algorithms like single-shot detection (SSD) and Faster R-CNN, demonstrating its reliability in identifying and classifying objects. Moreover, YOLOv8 exhibits remarkable robustness to variations in object size, shape, and orientation, even when there are occlusions and cluttered scenes. Its enhanced performance in low-light conditions further expands its applicability to diverse driving environments. The algorithm's ability to detect and classify multiple object classes, including vehicles, pedestrians, road signs, and more, is indispensable for comprehensive scene understanding. YOLOv8's efficient computation, requiring fewer resources and memory compared to alternatives, makes it well-suited for deployment on embedded systems within autonomous vehicles. Furthermore, its flexibility and customization capabilities allow for seamless integration with other perception tasks and tailoring to specific autonomous vehicle applications. The open-source nature of YOLOv8, coupled with a thriving community of developers, fosters continuous improvement and provides valuable support. In summary, YOLOv8's exceptional performance, versatility, and community support make it a highly promising choice for powering the perception systems of autonomous vehicles.

The YOLOv8 model for autonomous vehicles utilizes a complex array of mathematical computations to execute object detection and tracking with high precision. Below are some pivotal algorithms that facilitate this process.

Intersection over Union (IoU): Measures the overlap between predicted and ground truth bounding boxes using Eq. (1) [41].

$$IoU=\frac{\text{Area of Intersection}}{\text{Area of Union}} \quad (1)$$

It is calculated as the ratio of the area of overlap between the predicted and ground truth regions to the area of their union.

Focal Loss: Addresses class imbalance by focusing more on hard-to-classify instances using Eq. (2) [42].

$$\text{Focal Loss}\ (p_t)=-(1-p_t)^{\gamma}\log(p_t) \quad (2)$$

where, p_t is the model's predicted probability, γ is the focusing parameter, and $(1-p_t)^{\gamma}$ is the focusing term, introduced by focal loss.

Mean Average Precision (mAP): Evaluates the overall precision of object detection across various classes using Eq. (3)[43].

$$mAP=\frac{1}{num\_classes}\sum(class_{1_{AP}}....+class\_n\_AP) \quad (3)$$

3. Backbone (MultiNet++ with YOLOv8 Backbone)

The backbone of this architecture is built on YOLOv8, a highly optimized convolutional neural network (CNN) architecture, which acts as the feature extractor. The backbone processes the raw image data, extracting low- and high-level features through multiple convolutional layers. The layers C3, C4, and C5 represent convolutional blocks responsible for multi-scale feature extraction. Each layer captures features at different resolutions and scales, making the network robust to various object sizes and positions. MultiNet++ is an advanced multi-task learning architecture that performs both object detection and semantic segmentation. The backbone consists of multiple convolutional layers and Cross Stage Partial (C2f) blocks, helping in extracting hierarchical features from the input images:

Convolutional layers: Employed for feature extraction, each convolutional operation can be described mathematically using Eq. (4) [44].

$$Y=\text{ReLU}\ (W*X+b) \quad (4)$$

where * denotes the convolution operation, X is the input, W represents the weights of the convolutional filter, b is the bias, and Y is the output f C2f blocks: These blocks use shortcut connections similar to those in ResNet architectures, enhancing feature propagation without the vanishing gradient problem. The output Y of a block can be expressed as in Eq. (5).

$$Y=\text{Concat}\ (\text{Conv}\ (X),X)) \quad (5)$$

where Concat represents the concatenation of the feature map produced by the convolutional layer Conv and the input feature map X. SPPF (Spatial Pyramid Pooling - Fast): This layer pools features at different scales and concatenates them to maintain spatial hierarchies. Mathematically, it combines fixed-size outputs using Eq. (6).

$$Y=\text{Concat}(\text{MaxPool}_1(X),\text{MaxPool}_2(X),,\text{MaxPool}_n(X)) \quad (6)$$

4. Feature Pyramid Network (FPN) Neck

The FPN Neck is a critical component that facilitates the combination of low- and high-level features extracted by the backbone at different scales (P3, P4,

Manuscript received April 7, 2025; Revised July 20, 2025; Accepted August 15, 2025; date of publication September 10, 2025
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeeemi.v7i4.813

P5, P6, and P7). The pyramid network aggregates features from the early (shallow) and deeper layers of the backbone, enabling the model to detect both small and large objects with equal precision. By fusing features at various levels, the FPN confirms that the detector can generalize over an extended range of object sizes and shapes, making it versatile for use in complex urban environments and dynamic road conditions where objects of different sizes must be detected in real-time.

5. Shared Head (3D Object Detection and Classification)

The Shared Head component processes both image-based features and point cloud features (from the PointPillars network). This head simultaneously performs 3D localization (bounding box prediction) and category classification for objects detected in the vehicle's surroundings. The shared head consists of multiple fully connected layers, which refine the predictions of object categories and their 3D locations (x, y, and z coordinates, along with orientation). The architecture uses this shared head approach to efficiently manage resources and ensure that both the 2D image-based data and the 3D point cloud data are processed together in a unified framework, enhancing the robustness of the model.

6. Loss Function and Optimization

The architecture utilizes a combination of category loss and 3D localization loss to train the model. The category loss ensures that the model correctly classifies detected objects (e.g., pedestrians, cars, and trucks), while the localization loss minimizes errors in predicting the exact position of the objects in 3D space.

The architecture of the proposed system incorporates a shared encoder that processes input data and distributes features to three separate decoders responsible for different tasks. This section will detail the components of the encoder, the specific role of each decoder, and how they interact to perform their functions concurrently. The design focuses on maximizing the efficiency and speed of feature extraction and processing, crucial for real-time applications.

The architecture is structured around a central shared encoder and multiple specialized decoders for handling distinct tasks such as semantic segmentation, object detection, and classification. The shared encoder utilizes a convolutional neural network with deep learning to extract rich, hierarchical features from input data, which, in the case of autonomous driving, primarily consists of high-resolution images and 3D point clouds.

B. MultiNet++

The MultiNet architecture masterfully consolidates key perception tasks—classification, detection, and semantic segmentation—into a unified framework. This model uses a shared encoder to process incoming data, extracting pivotal features that are then funneled to task-specific decoders. Such integration enables MultiNet to execute multiple tasks simultaneously in a single forward pass, markedly boosting processing efficiency and slashing computational expenses. This synergy not only accelerates inference times but also heightens overall accuracy and performance by leveraging shared contextual data across tasks, exemplifying a significant leap in handling complex multi-task operations within autonomous systems. This dual-page expansion would delve into each component's contributions to the overarching system's

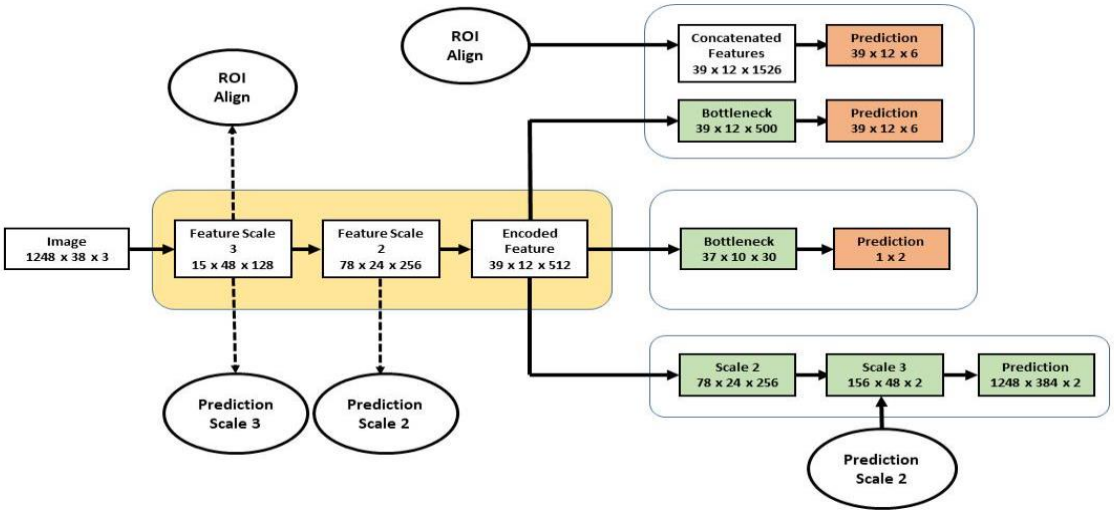

Fig. 2. Flowchart for multi-scale feature extraction for the MultiNet Methodology

Manuscript received April 7, 2025; Revised July 20, 2025; Accepted August 15, 2025; date of publication September 10, 2025
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeeemi.v7i4.813

efficacy and detail the interactions between the shared encoder and decoders.

The architecture depicted in Fig. 2 employs a multi-scale feature extraction approach where an input image transforms three distinct scales. This methodology ensures comprehensive feature extraction, capturing both macro and micro details essential for accurate object detection. The integration of ROI Align enhances the precision of feature extraction across these scales, ensuring that the features correspond accurately to regions of interest within the image. Subsequent concatenation of these features forms a rich, high-dimensional feature map, which is then streamlined through bottleneck layers. These layers are instrumental in distilling the most relevant features, reducing computational complexity, and preparing the model for final output predictions. The architecture supports multiple prediction outputs at different stages, facilitating real-time object detection and classification essential for dynamic environments such as autonomous driving.

1. Input Image

The model starts with an input image of size 1248×38×3, representing width, height, and three-color channels (RGB).

2. Feature Extraction

Feature extraction is performed at multiple scales. This diagram shows three scales: Scale 3 (15×48×128), Scale 2 (78×24×256), and Scale 1 (Encoded Feature 39×12×512). Each scale captures different levels of detail; smaller scales capture more global, structural information, while larger scales capture fine, detailed features.

3. Region of Interest (ROI) Align

The ROI align method is applied after the initial feature maps are generated. This technique helps in precisely extracting feature maps from the regions of interest, aligning them properly despite varying scales and positions within the image.

4. Concatenated Feature

Features from different scales or previous layers are concatenated. Here, a large feature map of size 39×12×1526 is created, which aggregates the information extracted from various parts of the image.

5. Bottlenecks and Predictions

Two bottleneck layers refine the features further. The first bottleneck compresses the concatenated features into a 500-dimensional space, and the second bottleneck processes these into a 30-dimensional space. These bottlenecks are crucial for reducing dimensionality and preparing features for final predictions.

Multiple predictions are made at different stages. The final prediction layers output detections for different attributes like object class, bounding box coordinates, etc., shown here with varying output sizes (e.g., 39×12×6 for delta predictions and 1×2 for another prediction).

6. Output Predictions

Final outputs include detailed predictions for each detected object in the image across multiple scales. These predictions include bounding box coordinates as well as class labels, which are essential for tasks like object detection and scene comprehension in applications involving autonomous driving. MultiNet uses its convolutional layers to process the input image and provide a comprehensive set of feature maps for object detection. In order to forecast the existence of

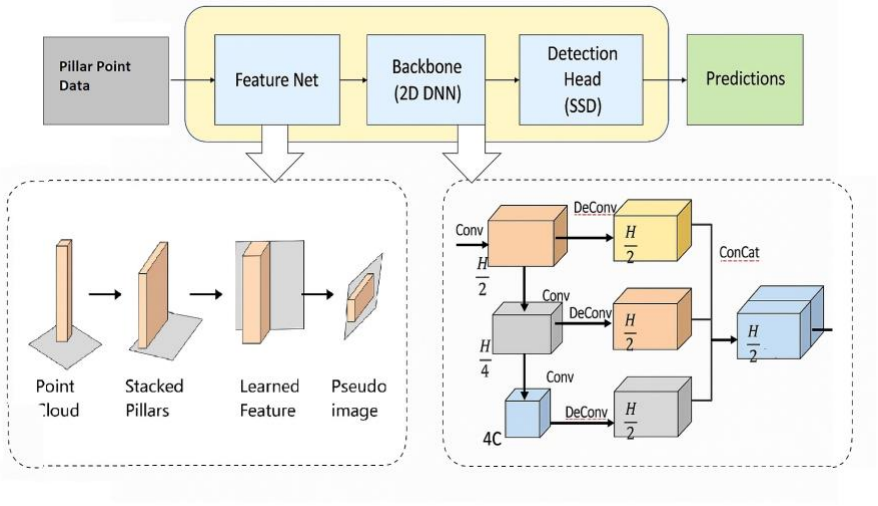

Fig.3. PointPillar Feature Integration for real-time detection and classification of objects

Manuscript received April 7, 2025; Revised July 20, 2025; Accepted August 15, 2025; date of publication September 10, 2025
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeeemi.v7i4.813

objects and their bounds, these feature maps are subsequently fed into direct bounding box regression layers or area proposal networks, if applicable. The categories of the detected items are determined in parallel by categorization layers. MultiNet++ is particularly well-suited for autonomous++ driving because it excels at handling the complex and dynamic nature of road environments. Autonomous vehicles need to constantly monitor their surroundings, detect objects, predict their trajectories, and make decisions in real time. MultiNet++'s unified architecture allows it to perform these tasks concurrently and efficiently, making it an extremely efficient remedy for both highway and urban driving scenarios. By detecting objects, segmenting the road, and classifying elements in the scene, MultiNet++ enables autonomous vehicles to understand their environment better and react accordingly. The multi-modal fusion of image-based features and other sensor data, such as LIDAR or radio detection and ranging (RADAR), further enhances the vehicle's perception abilities.

C. PointPillar Feature Integration

The PointPillars methodology innovatively enhances the processing of 3D point clouds, which is pivotal in autonomous vehicle technology. This approach effectively transforms the raw, unstructured point clouds into a neatly organized pillar format. Each "pillar" represents key features extracted from the data, structured in a way that optimizes spatial understanding. Subsequently, these pillars are rapidly encoded through a specialized neural network that guarantees rapid and effective operations. PointPillar feature integration used in architecture is shown in Fig. 3. The method excels in accurately capturing the environmental layout, crucial for real-time detection and classification of objects. By focusing on the spatial distribution and inherent structure of the surroundings, Point Pillars enables the system to interpret and respond to dynamic road situations quickly. This feature is vital in autonomous driving, where timely and accurate responses can significantly influence safety and operational effectiveness.

This organized representation improves the system's ability to recognize and respond to complicated surroundings while also speeding up data processing. Consequently, automobiles that have this equipment installed benefit from a more refined perception capability, which is crucial for navigating through diverse driving conditions and executing safe maneuvers without human intervention. Point Pillars stands out as a robust solution in the landscape of autonomous driving technologies, providing substantial improvements in how vehicles perceive and interact with their environment. By efficiently processing 3D point clouds, it supports enhanced situational awareness and decision-making capabilities, which are critical for the advancement of autonomous vehicle systems. The PointPillars methodology represents a notable progress in the 3D object detection field, particularly within the domain of autonomous vehicle technologies. It efficiently processes 3D point cloud information, that are typically captured by sensors such as LIDARs mounted on autonomous vehicles.

IV. Experiment

The proposed model uses an NVIDIA RTX 4060 GPU and 32 GB RAM, 2 TB storage for its implementation.

A. Dataset Used

A popular dataset for autonomous driving, containing images, point clouds, and annotations for object detection, tracking, and scene understanding, is used in our research [29]. The Kitti Dataset is used for training, validation, and testing of the proposed model. The dataset is publicly available at https://www.cvlibs.net/datasets/kitti/. This dataset contains 12919 images. The split of the dataset is as follows: training: 10335 images (80%), validation: 1292 images (10%), and test: 1292 images (10%).

B. Performance Metrics

To evaluate the performance of the proposed model, we use the metrics: accuracy, precision, recall, and F1 score [36]. Eq (7) to (10) present the mathematical expressions of these metrics.

1. Accuracy, in Eq. (7), is the ratio of correctly predicted data samples to the total number of input samples. In these equations, TP corresponds to true positives, FP to false positives, TN to true negatives, and FN to false negatives [45].

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP + TN + FP + FN}} \quad (7)$$

2. Precision, described in Eq. (8) refers to the ratio between correctly predicted positive samples and the total predicted positive samples, high precision relates to the low false positive rate [45].

Table 1. Results obtained during Training phase for various parameters

Model	Input Network Resolution	Recall (%)	Precision (%)	F1 Score (%)	Accuracy (%)	Training Time required in hrs.
Proposed Model- Yolov8 MultiNet+-	800x800	82.1	92.7	87.1	93.5	3.8

Manuscript received April 7, 2025; Revised July 20, 2025; Accepted August 15, 2025; date of publication September 10, 2025
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeeemi.v7i4.813

$$\text{Precision}=\frac{TP}{TP+FP} \quad (8)$$

2. Recall, as seen in Eq. (9), is the ratio of correctly predicted positive samples to all samples in the actual class [45].

$$\text{Recall}=\frac{TP}{TP+FN} \quad (9)$$

3. F1 Score, in Eq. (10), is defined as the Harmonic Mean between precision and recall [46].

$$\text{F1 Score}=2*\frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (10)$$

4. mAP50 and mAP95: These metrics represent the mean Average Precision at IoU thresholds of 0.50 and 0.95, respectively.

V. Results

A. Training of the Proposed Model

Table 1 displays the results of training datasets trained on a GPU for various parameters. The speed and effectiveness of the training process for deep learning models can be significantly impacted by multiple hardware combinations. Our approach, which is tailored for autonomous driving situations and object identification, exhibits notable enhancements. Fig. 4 displays the model's training loss as it decreases over each epoch. A typical trend shows a sharp decrease initially, followed by a gradual decline, indicating the model is learning from the training data. Table 2 compares the performance of various object detection models, including YOLOv3, SSD MobileNet v2, Faster R-CNN, EfficientDet-D0, and Yolov8 MultiNet++, using a GPU. The proposed object detection model performs exceptionally well, surpassing other models in terms of F1 score (82.70%), accuracy (78.50%), recall (76.80%), and precision (89.40%). This implies that the suggested model is very successful in precisely and quickly detecting objects. While they perform competitively, Faster R-CNN and EfficientDet-D0 have slower run times each frame (280 ms and 200 ms, respectively). Although SSD MobileNet v1 and YOLOv3 are faster, accuracy and precision are sacrificed. The proposed model is a great option for object identification jobs because of its excellent accuracy, precision, and speed balance (180 ms run time per frame), especially in situations where efficiency and accuracy are critical. Its sophisticated architecture and optimization methods, which allow it to recognize things quickly and accurately, are responsible for its exceptional performance. The strong recall and precision rates of the suggested model show that it can accurately identify the majority of objects while reducing false positives. This is especially crucial for autonomous cars, where precise object detection is essential. The proposed model beats Faster R-CNN by 5.6%, EfficientDet-D0 by 2.8%, YOLOv3 by 7.9%, and SSD MobileNet v1 by 12.3% in terms of recall. This implies that the suggested model detects objects more accurately, especially in complicated settings. Additionally, the precision rate of the suggested model

Table 2. Testing trained model in terms of various parameters only on GPU

Model	Input Network Resolution	Recall (%)	Precision (%)	Accuracy (%)	F1 (%)	Runtime per frame (ms)
Faster R-CNN	800×800	71.20	82.70	73.80	76.40	280
SSD MobileNet v1	300×300	64.50	70.30	68.20	67.20	110
YOLOv3	608×608	68.90	78.40	72.10	73.30	150
EfficientDet-D0	512×512	74.00	85.60	75.30	79.50	200
Proposed Model	800×600	76.80	89.40	78.50	82.70	180

Table 3. Testing trained model in terms of various parameters only on TPU

Model	Input Network Resolution	Recall (%)	Precision (%)	Accuracy (%)	F1 (%)	Runtime per frame (ms)
Faster R-CNN	800x800	73.40%	84.10%	74.70%	78.20%	150
SSD MobileNet v1	300x300	66.10%	72.20%	69.80%	69.00%	90
YOLOv3	608x608	70.20%	80.50%	73.50%	75.00%	120
EfficientDet-D0	512x512	75.50%	86.80%	76.90%	80.90%	170
Proposed Model	800x600	78.60%	90.80%	79.70%	84.30%	140

Manuscript received April 7, 2025; Revised July 20, 2025; Accepted August 15, 2025; date of publication September 10, 2025
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeeemi.v7i4.813

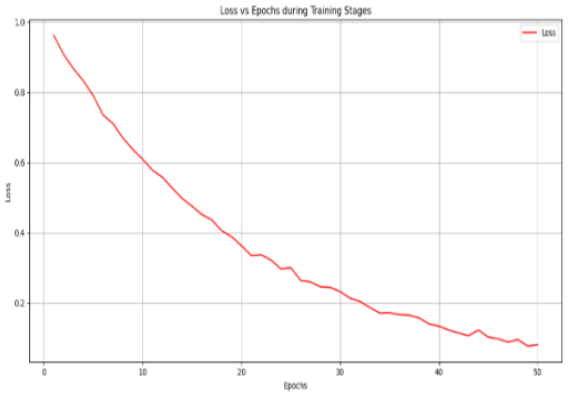

Fig. 4 Plot of Loss vs. Epoch during model training.

is higher than that of the other models, suggesting that it can reduce false positives. The proposed model's overall efficacy in object detection is demonstrated by its greater accuracy and F1 score when compared to the other models. The suggested model outperforms Faster R-CNN by 4.7%, EfficientDet-D0 by 3.2%, YOLOv3 by 6.4%, and SSD MobileNet v1 by 10.3% in terms of accuracy rate. The suggested model's F1 score is 15.5% higher than SSD MobileNet v1, 3.2% higher than EfficientDet-D0, 9.4% higher than YOLOv3, and 6.3% higher than Faster R-CNN. Table 3 compares the performance of various object identification models, including Faster R-CNN, SSD MobileNet v1, YOLOv3, EfficientDet-D0, and a proposed model, based on resolution, recall, precision, accuracy, F1 score, and runtime per frame. The proposed model has the greatest accuracy of 79.70% and an F1 score of 84.30%, showing an excellent balance of precision and recall. Notably, it has the maximum precision of 90.80% and a competitive recall of 78.60%. EfficientDet-D0 also performs well in terms of accuracy and F1 score, although it takes longer to run than other models. The trade-off between accuracy and runtime is a common feature of the models. Models that attain more accuracy, such as the proposed model and EfficientDet-D0, have longer runtimes, whereas speedier models, such as SSD MobileNet v1, trade off accuracy for speed. However, EfficientDet-D0's runtime is significantly longer than that of other models despite its high-performance metrics, implying potential inefficiencies in its implementation or architecture. Overall, the proposed model exhibits a promising balance of performance metrics, making it ideal for object detection in autonomous vehicles. Fig. 5. displays both the original image and the images that the suggested model identified. We offer a model that accurately detects and predicts objects in the image. In the bounding box, the object detection confidence score is displayed. A higher confidence score indicates a higher likelihood of object detection. Greater scores signify a higher level of certainty in detecting a certain object.

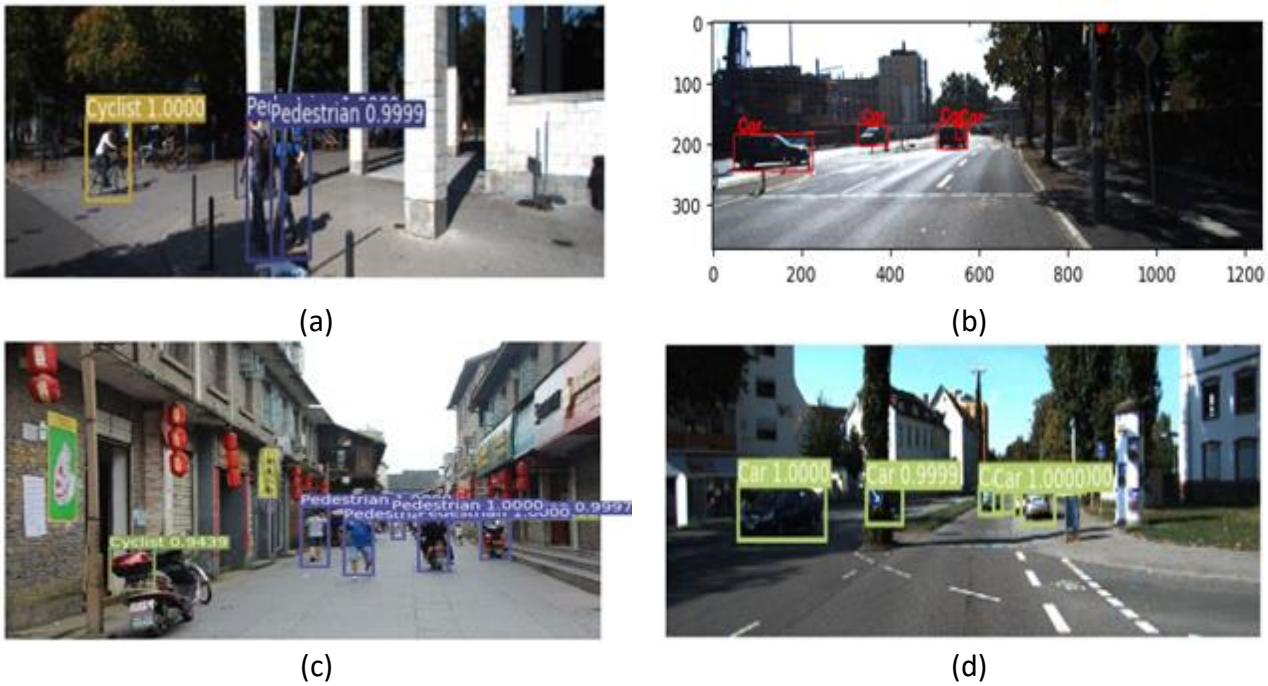

Fig. 5. Detected Objects in Images, (a) with confidence score, (b) without confidence score, (c) second image with confidence score, (d) third image with confidence score

Manuscript received April 7, 2025; Revised July 20, 2025; Accepted August 15, 2025; date of publication September 10, 2025
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeeemi.v7i4.813

VI. Discussion

Table 4 displays the results of the comparison of the proposed method with various detection techniques for various parameters. On the KITTI dataset, YOLOv8 MultiNet++ obtains the highest mAP50 score (55.70%) and mAP95 score (34.90%), demonstrating its remarkable capacity for object detection with a reasonable level of accuracy. With a mAP50 score of 52.20% and an mAP95 score of 31.40%, EfficientDet comes in second. SSD and YOLOv4 exhibit competitive performance, as seen by their respective mAP50 ratings of 48.50% and 43.50% and mAP95 scores of 28.80% and 22.30%, respectively. With respective mAP50 ratings of 42.70% and mAP95 scores of 21.90%, respectively, faster R-CNN lags behind.

With the quickest inference performance (41 FPS) on the KITTI dataset, YOLOv8 MultiNet++ is appropriate for real-time applications. With an inference speed of 31 FPS, YOLOv4 comes in second. EfficientDet and SSD both exhibit competitive performance, with inference speeds of 26 and 22 frames per second, respectively. With inference speeds of 7 frames per second, Faster R-CNN lags behind. On the KITTI dataset, YOLOv8 MultiNet++ performs exceptionally well on all three metrics (mAP50, mAP95, and FPS). It is a great option for object detection jobs because of its quick inference speed and high and moderate accuracy in object detection. Comparable performance is also shown by EfficientDet and SSD; YOLOv4 performs worse. Faster R-CNN is less appropriate for real-time applications due to its sluggish inference speed.

While our unified deep learning architecture achieves strong results in real-time object detection and semantic reasoning, it is important to recognize the limitations of the datasets used, which may affect model performance and generalization. Sensor noise from devices like LiDAR and cameras, caused by hardware constraints or environmental factors such as rain or fog, can degrade detection accuracy by introducing missing or distorted data [47]. Environmental variations, including changes in lighting conditions and weather, pose additional challenges since many datasets do not fully capture this diversity, potentially limiting robustness in extreme or rare scenarios [48]. Furthermore, dataset bias is common as most publicly available datasets are collected in specific geographic and traffic conditions, which may restrict model generalization to different environments [49]. Addressing these factors through more diverse data collection, noise-aware training, sensor fusion, and domain adaptation techniques is crucial for improving real-world applicability and robustness of autonomous vehicle perception systems.

VII. Conclusion

The recommended approach performs remarkably well, and is robust in autonomous vehicle perception by utilizing the advantages of YOLOv8 and MultiNet++. The integration of YOLOv8's efficiency and MultiNet++'s feature extraction capabilities results in a model that achieves state-of-the-art accuracy on various object detection benchmarks while maintaining real-time processing speeds. Our approach makes use of the advantages of current frameworks, such as the reliable one-stage monocular 3D object identification framework, the quick encoding techniques of PointPillars in order to identify objects from point clouds, and MultiNet's real-time semantic reasoning capabilities. By employing a shared encoder and task-specific decoders that carry out classification, detection, and segmentation simultaneously, the suggested approach seeks to improve computational efficiency and accuracy. When tested on difficult datasets, the architecture performs exceptionally well in terms of speed and precision, making it appropriate for applications that require real-time like autonomous driving. Overall, our system represents a significant step forward in the process of creating autonomous vehicle perception systems, paving the way for safer and more efficient autonomous driving solutions. In our model, we used Yolov8 MultiNet and during testing got an accuracy of 78.5%, precision of 89.4%, recall of 76.8% and FPS 41. In order to better assess the efficacy of our methodology across several autonomous vehicle platforms, future research might concentrate on using larger and more varied datasets.

References

[1] Y. Han, H. Zhang, H. Li, Y. Jin, C. Lang, Y. Li, Collaborative perception in autonomous driving: Methods, datasets, and challenges, IEEE Intelligent Transportation Systems Magazine (2023).

[2] D. Parekh, N. Poddar, A. Rajpurkar, M. Chahal, N. Kumar, G. P. Joshi,
W. Cho, A review on autonomous vehicles: Progress, methods and challenges, Electronics 11 (14) (2022) 2162.

[3] N. Sanil, V. Rakesh, R. Mallapur, M. R. Ahmed, et al., Deep learning techniques for obstacle detection and avoidance in driverless cars, in: 2020 International Conference on Artificial Intelligence and Signal Processing (AISP), IEEE, 2020, pp. 1–4.

[4] W. He, Z. Liu, S. Wang, Research on the application of recognition and detection technology in automatic driving, Highlights in Science, Engineering and Technology 94 (2024) 504–509.

[5] M. Teichmann, M. Weber, M. Zoellner, R. Cipolla, R. Urtasun, Multinet: Real-time joint semantic reasoning for autonomous driving, in: 2018 IEEE

Manuscript received April 7, 2025; Revised July 20, 2025; Accepted August 15, 2025; date of publication September 10, 2025
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeeemi.v7i4.813

intelligent vehicles symposium (IV), IEEE, 2018, pp. 1013–1020.

[6] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, O. Beijbom, Pointpillars: Fast encoders for object detection from point clouds, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 12697–12705.

[7] D. Feng *et al.*, "Deep Multi-Modal Object Detection and Semantic Segmentation for Autonomous Driving: Datasets, Methods, and Challenges," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 3, pp. 1341–1360, Mar. 2021, doi: 10.1109/TITS.2020.2972974.

[8] F. Liu, "Image Object Detection Algorithm for Autonomous Vehicles," 2024, pp. 225–233. doi: 10.2991/978-94-6463-512-6_26.

[9] F. S. Alsubaei, F. N. Al-Wesabi, and A. M. Hilal, "Deep Learning-Based Small Object Detection and Classification Model for Garbage Waste Management in Smart Cities and IoT Environment," *Applied Sciences*, vol. 12, no. 5, p. 2281, Feb. 2022, doi: 10.3390/app12052281.

[10] D. PS and U. V, "A Novel Hybrid Deep Learning Approach For 3D Object Detection And Tracking In Autonomous Driving," *Computer Science*, vol. 25, no. 3, Oct. 2024, doi: 10.7494/csci.2024.25.3.5597.

[11] Y. Parmar, S. Natarajan, and G. Sobha, "DeepRange: deep-learning-based object detection and ranging in autonomous driving," *IET Intelligent Transport Systems*, vol. 13, no. 8, pp. 1256–1264, Aug. 2019, doi: 10.1049/iet-its.2018.5144.

[12] S. Abdigapporov, S. Miraliev, V. Kakani, and H. Kim, "Joint Multiclass Object Detection and Semantic Segmentation for Autonomous Driving," *IEEE Access*, vol. 11, pp. 37637–37649, 2023, doi: 10.1109/ACCESS.2023.3266284.

[13] "YOLOv1 to YOLOv10: A Comprehensive Review of YOLO Variants and Their Application in Medical Image Detection," *Journal of Artificial Intelligence Practice*, vol. 7, no. 3, 2024, doi: 10.23977/jaip.2024.070314.

[14] T. Bui, G. Wang, G. Wei, and Q. Zeng, "Vehicle Multi-Object Detection and Tracking Algorithm Based on Improved You Only Look Once 5s Version and DeepSORT," *Applied Sciences*, vol. 14, no. 7, p. 2690, Mar. 2024, doi: 10.3390/app14072690.

[15] S. Guo *et al.*, "A Review of Deep Learning-Based Visual Multi-Object Tracking Algorithms for Autonomous Driving," *Applied Sciences*, vol. 12, no. 21, p. 10741, Oct. 2022, doi: 10.3390/app122110741.

[16] R. Walambe, A. Marathe, K. Kotecha, and G. Ghinea, "Lightweight Object Detection Ensemble Framework for Autonomous Vehicles in Challenging Weather Conditions," *Comput Intell Neurosci*, vol. 2021, no. 1, Jan. 2021, doi: 10.1155/2021/5278820.

[17] R. Wang *et al.*, "A Real-Time Object Detector for Autonomous Vehicles Based on YOLOv4," *Comput Intell Neurosci*, vol. 2021, no. 1, Jan. 2021, doi: 10.1155/2021/9218137.

[18] T. Sharma *et al.*, "Deep Learning-Based Object Detection and Classification for Autonomous Vehicles in Different Weather Scenarios of Quebec, Canada," *IEEE Access*, vol. 12, pp. 13648–13662, 2024, doi: 10.1109/ACCESS.2024.3354076.

[19] P. Padmane, T. Dasare, P. Deshkar, N. Dasgupta, A. Kale, and B. Hamdard, "A Review on Real Time Object Detection Using Deep Learning," *Int J Res Appl Sci Eng Technol*, vol. 11, no. 4, pp. 1215–1217, Apr. 2023, doi: 10.22214/ijraset.2023.50281.

[20] A. Kishore Kumar and V. Palanisamy, "Detection of lanes, obstacles and drivable areas for self-driving cars using multifusion perception metrics," *Journal of Autonomous Intelligence*, vol. 7, no. 3, Jan. 2024, doi: 10.32629/jai.v7i3.1059.

[21] Z. Guo, Y. Huang, X. Hu, H. Wei, and B. Zhao, "A Survey on Deep Learning Based Approaches for Scene Understanding in Autonomous Driving," *Electronics (Basel)*, vol. 10, no. 4, p. 471, Feb. 2021, doi: 10.3390/electronics10040471.

[22] Y. Dai, D. Kim, and K. Lee, "An Advanced Approach to Object Detection and Tracking in Robotics and Autonomous Vehicles Using YOLOv8 and LiDAR Data Fusion," *Electronics (Basel)*, vol. 13, no. 12, p. 2250, Jun. 2024, doi: 10.3390/electronics13122250.

[23] G. Sistu *et al.*, "NeurAll: Towards a Unified Visual Perception Model for Automated Driving," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, IEEE, Oct. 2019, pp. 796–803. doi: 10.1109/ITSC.2019.8917043.

[24] M. Liu, S. Luo, K. Han, B. Yuan, R. F. DeMara, and Y. Bai, "An Efficient Real-Time Object Detection Framework on Resource-Constricted Hardware Devices via Software and Hardware Co-design," in *2021 IEEE 32nd International Conference on Application-specific Systems, Architectures and Processors (ASAP)*, IEEE, Jul. 2021, pp. 77–84. doi: 10.1109/ASAP52443.2021.00020.

[25] Z. Dai, Z. Guan, Q. Chen, Y. Xu, and F. Sun, "Enhanced Object Detection in Autonomous Vehicles through LiDAR—Camera Sensor Fusion," *World Electric Vehicle Journal*, vol. 15,

Manuscript received April 7, 2025; Revised July 20, 2025; Accepted August 15, 2025; date of publication September 10, 2025
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeeemi.v7i4.813

no. 7, p. 297, Jul. 2024, doi: 10.3390/wevj15070297.

[26] R. Murendeni, A. Mwanza, and I. C. Obagbuwa, "Using a YOLO Deep Learning Algorithm to Improve the Accuracy of 3D Object Detection by Autonomous Vehicles," *World Electric Vehicle Journal*, vol. 16, no. 1, p. 9, Dec. 2024, doi: 10.3390/wevj16010009.

[27] C. K. -, "Autonomous Vehicles: Applications of Deep Reinforcement Learning," *International Journal For Multidisciplinary Research*, vol. 6, no. 1, Feb. 2024, doi: 10.36948/ijfmr.2024.v06i01.13792.

[28] O. A. Fawole and D. B. Rawat, "Recent Advances in 3D Object Detection for Self-Driving Vehicles: A Survey," *AI*, vol. 5, no. 3, pp. 1255–1285, Jul. 2024, doi: 10.3390/ai5030061.

[29] N. U. A. Tahir, Z. Zhang, M. Asim, J. Chen, and M. ELAffendi, "Object Detection in Autonomous Vehicles under Adverse Weather: A Review of Traditional and Deep Learning Approaches," *Algorithms*, vol. 17, no. 3, p. 103, Feb. 2024, doi: 10.3390/a17030103.

[30] P. Azevedo and V. Santos, "YOLO-Based Object Detection and Tracking for Autonomous Vehicles Using Edge Devices," 2023, pp. 297–308. doi: 10.1007/978-3-031-21065-5_25.

[31] J. Feng, F. Wang, S. Feng, and Y. Peng, "A Multibranch Object Detection Method for Traffic Scenes," *Comput Intell Neurosci*, vol. 2019, pp. 1–16, Nov. 2019, doi: 10.1155/2019/3679203.

[32] S. Sun, Y. Yin, X. Wang, D. Xu, W. Wu, and Q. Gu, "Fast object detection based on binary deep convolution neural networks," *CAAI Trans Intell Technol*, vol. 3, no. 4, pp. 191–197, Dec. 2018, doi: 10.1049/trit.2018.1026.

[33] S. ARTHAM, S. Borde, and S. Shekhar, "Deep Learning for Autonomous Vehicle Object Detection," Oct. 30, 2023. doi: 10.21203/rs.3.rs-3506149/v1.

[34] M. Sukkar, M. Shukla, D. Kumar, V. C. Gerogiannis, A. Kanavos, and B. Acharya, "Enhancing Pedestrian Tracking in Autonomous Vehicles by Using Advanced Deep Learning Techniques," *Information*, vol. 15, no. 2, p. 104, Feb. 2024, doi: 10.3390/info15020104.

[35] S. Shaikh, J. Chopade, and G. Kharate, "Object Classification and Tracking Using Scaled P8 YOLOv4 Lite Model," *Periodica Polytechnica Electrical Engineering and Computer Science*, vol. 67, no. 1, pp. 102–111, Jan. 2023, doi: 10.3311/PPee.20685.

[36] R. Kadu and S. Pawar, "Advanced Bi-CNN for Detection of Knee Osteoarthritis using Joint Space Narrowing Analysis," *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 7, no. 1, pp. 80–90, Nov. 2024, doi: 10.35882/jeeemi.v7i1.574.

[37] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans Pattern Anal Mach Intell*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.

[38] W. Liu *et al.*, "SSD: Single Shot MultiBox Detector," Dec. 2015, doi: 10.1007/978-3-319-46448-0_2.

[39] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," Apr. 2020, [Online]. Available: http://arxiv.org/abs/2004.10934

[40] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and Efficient Object Detection," Nov. 2019, [Online]. Available: http://arxiv.org/abs/1911.09070

[41] Viraktamath, S. V., Yavagal, M., & Byahatti, R. (2021). Object detection and classification using YOLOv3. International Journal of Engineering Research & Technology (IJERT), 10(02), 197-202.

[42] Warule, P., Chandratre, S., Mishra, S. P., & Deb, S. (2024). Detection of the common cold from speech signals using transformer model and spectral features. Biomedical Signal Processing and Control, 93, 106158.

[43] Wang, T., Yang, F., & Tsui, K. L. (2020). Real-time detection of railway track component via one-stage deep learning networks. Sensors, 20(15), 4325.

[44] Mazumdar, A., & Rawat, A. S. (2019, September). Learning and recovery in the ReLU model. In *2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton)* (pp. 108-115). IEEE

[45] Warule, P., Mishra, S. P., & Deb, S. (2023). Time-frequency analysis of speech signal using Chirplet transform for automatic diagnosis of Parkinson's disease. *Biomedical Engineering Letters*, *13*(4), 613-623

[46] Warule, P., Mishra, S. P., & Deb, S. (2023). Time-frequency analysis of speech signal using wavelet synchrosqueezing transform for automatic detection of Parkinson's disease. IEEE Sensors Letters, 7(10), 1-4.

[47] X. Chen, H. Ma, J. Wan, B. Li, T. Xia, Multi-view 3d object detection network for autonomous driving, in: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2017, pp. 1907–1915.

[48] C. Sakaridis, D. Dai, L. Van Gool, Semantic foggy scene understanding with synthetic data, International Journal of Computer Vision 126 (2018) 973–992.

Manuscript received April 7, 2025; Revised July 20, 2025; Accepted August 15, 2025; date of publication September 10, 2025
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeeemi.v7i4.813

[49] A. Geiger, P. Lenz, R. Urtasun, Are we ready for autonomous driving? the kitti vision benchmark suite, in: 2012 IEEE conference on computer vision and pattern recognition, IEEE, 2012, pp. 3354–3361.

Authors Biography

Vishal Anil Aher has completed bachelor's degree in Electronics & Telecommunication Engineering from Savitribai Phule Pune University, Pune in the year 2006. He earned his Master's degree in Electronics & Telecommunication Engineering (VLSI & Embedded System) from Savitribai Phule Pune University, Pune in 2012. He has 18 years of teaching experience as an Assistant Professor at Pravara Rural Engineering College, Loni. He is currently a Research Scholar in Electronics and Telecommunication Engineering at Sanjavani College of Engineering, Kopargaon. He has published 25 research papers in international journals and presented 26 papers at conferences. https://orcid.org/0009-0000-7042-3920

Dr. Satish R. Jondhale received his B.E.in Electronics and Tele-communication in 2006, his M.E. in Electronics and Telecommunication in 2012, and his Ph.D. in Electronics and Telecommunication in 2019 from Savitribai Phule Pune University, Pune, India. He has been working as an Assistant Professor in the Electronics and Telecommunication Department at Amrutvahini College of Engineering, Sangamner, Maharashtra, India, for more than a decade. His research interests are Signal processing, Target Localization and Tracking, Wireless Sensor Networks, Artificial Neural Networks and Applications, Image Processing, and Embedded System Design. He has several publications in reputed international journals. He has published two authored books entitled "Received Signal Strength Based Target Localization and Tracking Using Wireless Sensor Network", and "Internet of Things: From Theory to Practice" with Springer and CRC Press, respectively. He has been a reviewer for peer-reviewed journals such as IEEE Transactions on Industrial Informatics, IEEE Sensors, Signal Processing (Elsevier), IEEE Access, IEEE Signal Processing Letters, Ad Hoc, Sensor Wireless Networks, and so on. https://orcid.org/0000-0003-2908-5610

Dr. Balasaheb Shrirangrao Agarkar holds a Ph.D. in Electronics Engineering from Swami Ramanand Teerth Marathwada University, Nanded (India), received in 2016. He received a bachelor's degree (BE) in Electronics Engineering and a master's (M. Tech.) in Electronics Design Technology from Dr. Babasaheb Ambedkar Matathwada University, Chh. Sambhajinagar (India) in 1990 and 1998, respectively. Currently, he is working as a Professor in the Department of Electronics and Computer Engineering, Sanjivani College of Engineering, Kopargaon, India. He is a member of the board of studies (BoS) and a research guide in Electronics and Telecommunication Engineering at Savitribai Phule Pune University, Pune, India. His area of research interests is Computer Networks, Packet Classification Algorithms, Artificial Neural Networks, Neuro-Fuzzy Systems, and Image Processing. He has published 12 research papers in international journals. https://orcid.org/0000-0002-2775-80

Dr. Sachin Vasant Chaudhari holds a Ph.D. in electronics engineering. He is an Associate Professor in the Electronics and Telecommunication Engineering Department at Sanjivani College of Engineering, Kopargaon, which is affiliated with Savitribai Phule Pune University, Pune, India. He has worked on a broad range of topics, including automated deep learning, UAV route planning, hybrid energy systems, and medical applications. He has published over 35 papers in reputed International Journals. He is a professional membership of The Institution of Engineers (IEI) & International Association of Engineers (IANG) https://orcid.org/0009-0005-8856-8905 .

Manuscript received April 7, 2025; Revised July 20, 2025; Accepted August 15, 2025; date of publication September 10, 2025
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeeemi.v7i4.813