# A Novel Deep Learning Architecture for Real-Time Autonomous Driving Perception with Fuzzy Graph-Cut and Metaheuristic Optimization

Pamarthi Satyanarayana[1], N. K. Anushkannan[2], M. Vijay[3], R. M. Joany[4], Badampudi Krishnaveni[5], Hirald Dwaraka Praveena[6], Johny Renoald Albert[7*], R. G. Vidhya[8]

[1] Department of Electronics and Communication Engineering, Siddhartha Academy of Higher Education, Deemed to be University, Vijayawada 520007, India

[2] Department of Electronics and Communication Engineering, Kathir College of Engineering, Coimbatore 641062, India

[3] Department of Electronics and Communication Engineering, SRM TRP Engineering College, Tiruchirappalli 621105, India

[4] Department of Electronics and Communication Engineering, Sathyabama Institute of Science and Technology, Chennai 600119, India

[5] Department of Mathematics, Aditya University, Surampalem 533437, India

[6] Department of Electronics and Communication Engineering, School of Engineering, Mohan Babu University, Tirupati 517102, India

[7] Department of Robotics Engineering, Karunya Institute of Technology and Sciences (Deemed University), Coimbatore 641114, India

[8] Department of ECE, HKBK College of Engineering, Bangalore 560048, India

Corresponding Author Email: jorenoeee@gmail.com

**ABSTRACT**

In this paper, we integrate metaheuristic optimization approaches with advanced image processing to present a unique deep learning framework for real-time lane line and object recognition in autonomous driving systems. The system leverages the Graz dataset and nuScenes comprising diverse urban driving scenes, to ensure robustness across varying conditions. Initially, input images undergo preprocessing using min-max normalization and Gaussian filtering to enhance image quality and suppress noise. A Graph-Cut Optimized Fuzzy Gaussian Mixture Model (GC-GMM) is presented for precise region segmentation, which successfully distinguishes lane markers and vehicle regions from the background by fusing probabilistic clustering with spatial consistency. One of the main innovations in our method is the Hybrid Bald Eagle-Crow Search Algorithm (HBE-CSA), which is intended to improve feature discrimination and optimize the detection process by cleverly adjusting model parameters. The suggested method's superiority is demonstrated by experimental findings, which confirm its potential for deployment in real-time autonomous navigation mAP@0.5 performance metrics for an object detection model with a 99.5% detection accuracy, 90% precision, 93% recall, 98% ROC, and 92% F1-score. For intelligent transportation applications, this framework offers a strong and effective solution.

## 1. INTRODUCTION

The intelligent vehicle is the bearer of the thorough integration of numerous technologies and is a crucial component of intelligent transportation systems. Despite the promising results of vision-based autonomous driving, the challenge of analyzing the complex traffic scenario using the data gathered remains. Recently, numerous tasks related to autonomous driving have been developed independently utilizing various models, including the object detection task and the intention identification job [1]. Designing and developing successful autonomous driving systems (ADS) involves many obstacles, particularly in developing nations with inadequate road infrastructure. For sophisticated ADS to determine whether to change lanes, slow down, or even stop, it must be able to detect barriers in lanes. Poor or non-existent lane line demarcation, inadequate traffic control, and a variety of driving habits are some of the difficulties faced by developing nations like India. This makes it difficult to create reliable lane-keeping or lane-changing decision systems for an ADS [2]. In actual driving situations, autonomous driving necessitates the accurate and dependable detection and recognition of nearby objects. Even though several object detection algorithms have been put out, not all of them are reliable enough to identify and detect things that are obscured or truncated [3]. Road safety and traffic management may benefit from the system's ability to make decisions in real time, which is essential for maximizing autonomous vehicle maneuvering [4]. Intrusion for the autonomous car concept to become a reality, lane detection is essential. The necessity for substantial hand-crafted features and post-processing approaches in traditional lane recognition methods makes the

models feature-oriented and vulnerable to instability due to fluctuations in road situations. Convolutional neural network (CNN) models, in particular, have been suggested and used in recent years for pixel-level lane segmentation using Deep Learning (DL) models [5]. Stakeholders in autonomous vehicles (AVs) are still looking for proof of the new technology's safety performance through AV testing on in-use roads, AV-specific road networks, and AV test tracks. Some public trust in the safety of AV operations has been damaged, nevertheless, by recent AV-related fatalities on in-service roads. Furthermore, the real-world driving environment cannot be sufficiently described by test tracks. Driving simulators are still a popular way to test for AV because of this [6].

The following is the primary contribution of the suggested method:

(1) The researchers used the GRAZ dataset and nuScenes, which offers high-resolution and varied road scene images, to conduct thorough training and evaluation for lane line and object detection in challenging driving situations.

(2) nuScenes provides richer data (1,000 scenes, 1.4 million photos) from cameras and other sensors, while the original study depends on the GRAZ dataset (1115 images, camera-only). This allows for prolonged assessment without the need to completely rebuild the model.

(3) Gaussian filtering was used to improve edge information and reduce noise, while Min-Max normalization was used to scale image intensities. This improved the input quality for tasks involving detection and segmentation.

(4) A hybrid segmentation technique that handles ambiguity and maintains spatial continuity by combining Graph-Cut optimization with fuzzy Gaussian Mixture Models. This allows for the precise extraction of lane lines and vehicle regions.

(5) HBE-CSA, a novel metaheuristic algorithm, was presented. It optimizes deep learning parameters to increase detection precision and convergence speed by combining the exploration strengths of the Bald Eagle Search and Crow Search algorithms.

(6) The entire system achieves 99.5% detection accuracy and exhibits strong robustness and generalization across a variety of scenarios.

The following is how the following sections will be formatted: Our suggested approach's literature review and other pertinent data are covered in Section 2. Section 3 explains the suggested approach; Section 4 describes the experiment and data analysis; and Section 5 summarizes the study's findings and recommendations for further research.

## 2. RELATED WORK

Kumar et al. [7] have suggested, a lane detection techniques based on deep learning. Hemalatha et al. [8] have suggested, a multitask detection algorithm with a high feature extraction capability, the Multitask Detection Network (MDNet), which is based on Cross Stage Partial Networks with Darknet53 Backbone (CSP-DarkNet53), can concurrently identify lane lines, bicycles, people, traffic signals, and signs. Xia et al. [9] have recommended, an automated driving system (ADS) data collection and analytics platform for connected automated vehicle (CAV) cooperative perception-based vehicle trajectory extraction, reconstruction, and assessment. Qian et al. [10] have described a unified neural network called DLT-Net to concurrently recognize traffic objects, lane lines, and drivable areas. Table 1 shows that comparison of recent literature study.

**Table 1.** Comparative literature study

| Year | Title | Key Method/Approach | Performance Metrics | Dataset Used | Limitations | Reference |
|---|---|---|---|---|---|---|
| 2025 | Deep learning-enhanced environment perception for autonomous driving: MDNet with CSP-DarkNet53 | Multitask Detection Network (MDNet) based on CSP-DarkNet53 for lane lines, bicycles, people, traffic signals, and signs | Not specified; focuses on high feature extraction capability | General autonomous driving scenes | Limited detail on real-time performance and specific dataset. | [11] |
| 2025 | Deep learning-based lane detection for intelligent driving | Survey of 2D and 3D deep learning-based lane detection methods | Comprehensive review; no specific metrics | Various datasets reviewed | Lack of specific performance metrics and implementation details. | [12] |
| 2025 | A Novel Hybrid Deep Learning Algorithm for Object and Lane Detection | Hybrid model using YOLOv5 for object detection and advanced lane detection | Superior real-time performance; high accuracy | Real-time driving scenarios | Limited discussion on robustness in adverse conditions. | [13] |
| 2025 | Monocular Lane Detection Based on Deep Learning: A Survey | Survey of monocular lane detection methods | Attracts attention in autonomous driving perception | Various monocular datasets | Broad scope; lacks specific performance comparisons. | [14] |
| 2024 | Real-time lane detection for autonomous vehicles using YOLOV5 Segmentation Large Model | YOLOv5 for lane (object) detection in a single framework | Enhances efficiency and accuracy | Real-time videos | Limited to specific YOLOv5 model; may not generalize. | [15] |
| 2024 | Effective lane detection on complex roads with convolutional neural networks | CNN-based approach for poor roads, curves, broken lanes | Accurate on complex roads | Not specified | Limited discussion on computational efficiency. | [16] |
| 2024 | A Novel Hybrid Deep Learning Algorithm for Object and Lane Detection | Advanced deep learning models for object and lane detection | Significant contributions to autonomous driving | Autonomous driving datasets | Limited detail on model optimization process. | [17] |

Prakash et al. [18] have suggested, a model CNN and YOLO are used to identify several things for an autonomous vehicle. Liu et al. [19] have reported, a reliable target identification and tracking of self-driving cars in extreme

weather using radar and information fusion. Behera et al. [20] have proposed, a new FL framework that blends the benefits of synchronous and asynchronous approaches. Prasad et al. [21] have reported, a creation of an autonomous car's software stack utilizing a robot operating system. Usman et al. [22] have proposed, a multi-task framework that meets the needs of precise and effective visual perception by concurrently completing lane line recognition, drivable area segmentation, and object detection. Periasamy et al. [23] have recommended, a model for real-time lane line detection. Albert et al. [24] have proposed, a systematic Bayesian framework to combine the results of stereo-based and semantic detection. Arthi and Murugeswari [25] have proposed, an autonomous vehicles' ability to identify objects in inclement weather. Gajjar et al. [26] have recommended, a both classification and localization problems are addressed for semantic lane segmentation using the Global Convolution Networks (GCN) model. Liu et al. [27] have recommended, a feature fusion module: Cross Convolution module, which enhances lane detecting precision. Teichmann et al. [28] have suggested, a real-time application like autonomous driving, depend heavily on computational times.

## 3. PROPOSED WORK

For autonomous driving systems, the suggested architecture offers a complete deep learning solution for real-time object and lane line recognition. First up is the GRAZ dataset and nuScenes, which provides a wide variety of rich driving situations. Image quality is improved by applying pre-processing methods like Gaussian Filtering and Min-Max Normalization [29]. A new segmentation technique called Graph-Cut Optimized Fuzzy GMM (GC-GMM) combines spatial optimization and probabilistic modeling to precisely extract lane and object sections [30]. To improve detection performance even more, a new Hybrid Bald Eagle-Crow Search Algorithm (HBE-CSA) is suggested to efficiently optimize deep learning parameters. Figure 1 shows the five successive phases that make up the suggested framework.
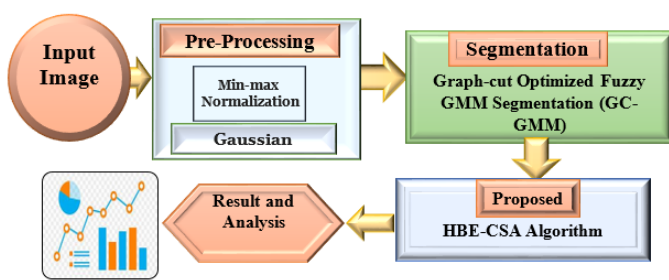


**Figure 1.** Workflow for the suggested approach

### 3.1 Image pre-processing

Noisy values that are too high, too low, or almost the same as a certain number are swapped out for zero by default. Using unsupervised instance-based filtering, duplicate instances are eliminated. In order to fill in the missing values, the modes and means of the attribute values are used. Across various numerical ranges, the cleaned samples' values fluctuate. After that, the data is transformed by applying the Min-max normalization procedure, which is outlined in Eq. (1), to normalize the numerical features that they fall between 0 and 1.

$$X = \frac{X - MIN}{MAX - MIN} \tag{1}$$

where, $X$ is the numerical value for each feature sample, MIN is the feature's lowest value, and MAX is its greatest value.

The parameters used for Gaussian filtering must be specified in order to guarantee experiment reproducibility. With a sigma value of 1.0 regulating the Gaussian distribution's standard deviation and a kernel size of 5×5 pixels defining the smoothing process's region of action, the Gaussian filter is applied. These parameters are essential because they affect the level of noise reduction and the retention of significant image characteristics. Clarity on the noise reduction step and easier experiment replication will result from including this comprehensive information on the Gaussian filtering procedure. Using a Gaussian kernel, the image is convolved.

$$G(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 - y^2}{2\sigma^2}\right) \tag{2}$$

The degree of smoothing and the breadth of the kernel are determined by the standard deviation of the Gaussian distribution, σ, where $G(x,y)$ is the value of the Gaussian function at coordinates $(x,y)$.

### 3.2 Graph-cut optimized fuzzy GMM segmentation (GC-GMM)

The next stage of our innovative segmentation model is the use of the fuzzy Gaussian Mixture Model (GMM) to divide the images into areas or segments following pre-processing with contrast stretching and homomorphic filtering techniques. In contrast to classic GMM, which places every pixel in a single cluster, our fuzzy GMM permits pixels to be a part of several clusters with different levels of affiliation. By taking into consideration the uncertainty in pixel membership, this addition offers a more adaptable and realistic depiction of image segmentation.

Initializing the Gaussian mixture model's parameters is the first step in processing the fuzzy GMM segmentation. Cluster-describing parameters include the number of clusters, covariance matrices, mean vectors, and the degree to which each pixel belongs to each cluster. Using fuzzy membership functions, like the Gaussian membership function, the degree of membership $i, j$ of a pixel $i$ to a cluster j is calculated. This function uses a pixel's similarity to the cluster's mean vector and covariance matrix to determine how likely it is that the pixel belongs to that cluster.

$$U_{IJ} = \frac{1}{\sum_{k=1}^{K} w_k} \sum_{k=1}^{K} w_k \cdot N\left(\frac{x_i}{U_k}\sum_k\right) \cdot N\left(\frac{x_i}{U_j}\sum_j\right) \tag{3}$$

The degree of membership of pixel $i$ to cluster $j$ is represented by '$j$' the mixing coefficients are indicated by $w_k$, the Gaussian probability density function of pixel $i$ given the mean vector k and covariance matrix $k$ is represented by $N\left(\frac{x_i}{U_k}\sum_k\right)$, and the same is represented for cluster $j$ by $N\left(\frac{x_i}{U_j}\sum_j\right)$. We present a unique graph-cut optimization-based

post-processing method to enhance boundary adherence and refine the segmentation results. In this stage, we use a graph to represent the segmentation problem, with nodes standing for pixels and edges for pairwise associations between pixels. By rephrasing the segmentation problem as an energy minimization effort, graph-cut optimization can be used to maximize each pixel's label assignment.

Both a data term and a smoothness term are combined in the graph-cut optimization energy function, or $E_{graphcut}$. The smoothness term promotes smooth borders and spatial coherence, whereas the data term describes the probability that each pixel belongs to a certain label. Here's how to define the energy function:

$$E_{graphcut} = \sum \varphi_d(i, l_i) + \sum_{(i,j) \in edges} \varphi_s(i, l_i) \qquad (4)$$

The label for pixel I is represented by $l_i$, the data term potential function is indicated by $\varphi_d(i, l_i)$, and the smoothness term potential function for adjacent pixels I and J is indicated by $\varphi_s(i, l_i)$.

After graph-cut optimization, we use label fusion to improve the segmentation accuracy even more. The knowledge from several segmentation results, such as the original fuzzy GMM segmentation and alternative segmentation outputs from various algorithms or parameter settings, is included into this stage. We obtain strong and accurate segmentation findings by giving greater weight to labels that are more dependable and consistent.

This is a representation of the label fusion process:

$$l_{final}(i) = \frac{\sum(w_{seg} . l_{seg}(i))}{\sum w_{seg}} \qquad (5)$$

The final label applied to pixel I is indicated by $l_{final}(i)$, the weight allocated to each segmentation result is represented by $w_{seg}$, and the label applied to pixel I in a particular segmentation result is represented by $l_{seg}(i)$.

### 3.3 Proposed hybrid optimisation algorithm

The successful optimization of the GMM-UBM's hyper-parameters is necessary for its log-likelihood. BESA and CSA mutualism is demonstrated by the suggested hybrid algorithm. The predation tactic of the bald eagle is replicated by the BESA. There are three main components to it: choosing a precise search area, searching inside that area, and determining the optimal opportunity to attack the target. The bald eagle's most recent movement is connected to the search space that was chosen during the select step. Eagles look for the ideal spot in the search area to carry out their hunting maneuvers. In the following section, eagles go in different directions within the first chosen search zone in an attempt to find prey. During the diving stage, the eagles attack the prey to capture it by diving from the best position. BESA is good at exploitation in general. However, it lacks the capacity to explore the entire world. As a result, it occasionally struggles to complete the specified optimization task effectively and falls short of reaching the global optima.

The local optima can be avoided by the CSA. Crow's activities, including as identifying members of a group and warning one another in unfavourable circumstances, are replicated by the CSA. Additionally, they can remember their food hiding spots for longer and employ a variety of gimmicks to communicate with one another. Figure 2 illustrates how CSA and BESA's hybridization may aid in the discovery of better solutions by taking into account their respective advantages and disadvantages.
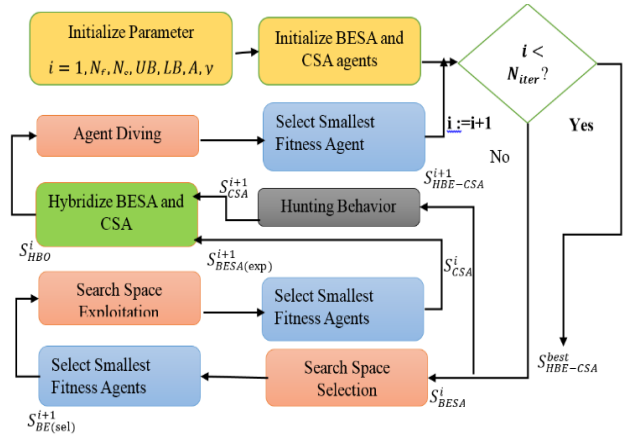


**Figure 2.** The hybrid bald eagle-crow search algorithm (HBE-CSA) flow diagram

### 3.4 Performance evaluation criteria

Evaluates the performance of the proposed hybrid HBE-CSA using four popular classification metrics: accuracy, recall, precision, and F1-score. The suggested model's usefulness is fully evaluated by these measures, with accuracy being a key component in determining the model's overall precision. This assessment measure can be calculated using Eq. (6):

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \qquad (6)$$

To calculate the precision, divide the number of accurately predicted positive cases by the total number of anticipated positive cases. As shown below, Eq. (7) is used to calculate this evaluation metric:

$$\text{Precision} = \frac{TP}{TP+FP} \qquad (7)$$

Among all outcomes having a true value of positive, the recall rate is the percentage of accurate model predictions:

$$\text{Recall} = \frac{TP}{TP+FN} \qquad (8)$$

$$\text{mAP@0.5} = \frac{1}{N} \sum_{i=1}^{N} AP_i \qquad (9)$$

where, mAP is mean average precision, @0.5 is intersection over union threshold, and N is averaging over multiple classes.

## 4. EXPERIMENTAL RESULT AND DISCUSSION

### 4.1 Dataset

Models for object detection are greatly impacted by the dataset. The information provided to the model must come from a variety of sources and under a range of circumstances.

Training and validation are impacted by proper labelling as well. The 1115 photos that make up the training and validation data were taken from the GRAZ dataset and from a smartphone camera in various lighting and weather situations (https://github.com/ruwantw/DSceneKG). Then, the nuScenes provider richer data (1000 scenes, 1.4 million photos) from cameras and other sensors (https://www.nuscenes.org/nuimages). Multiple objects may be present in each photograph in the collection. Once the dataset has been labelled, it is received in text format and contains items for cars, bicycles, pedestrians, stop signs, pedestrian crossings, give-way signs, roundabouts, uneven roads, and traffic signs with 20- and 30-speed limits. The text file contains the width and height values of the bounding boxes as well as the coordinate values of the center points of the bounding boxes that enclose the objects in each image. The testing procedure made use of films from everyday life. The dataset parameter is shown in Table 2. A few pictures from the dataset are shown in Figure 3. Bicycle class represents motorcycle and bicycle items in this study, while car class represents bus, automobile, and minibus objects. The HBE-CSA model was trained on a Tesla T4 GPU for 1000 epochs on Google Collaborator. Overfitting may become more likely as the number of epochs increases. By expanding and varying the amount of photos, this problem can be lessened. An Intel Core i5-9400 CPU was used for the object and lane detection testing phase. The HBE-CSA model's confusion matrix is shown in Figure 4. According to the matrix, the HBE-CSA model's total accuracy was 99.5%. In a variety of illumination and weather scenarios, the model showed good generalization across a wide range of classes, including automobiles, pedestrians, bicycles, and different traffic signals. With precision of 99.6%, recall of 99.4%, and $mAP@0.5$ of 99.66%, it demonstrated a reasonably balanced and dependable classification performance.
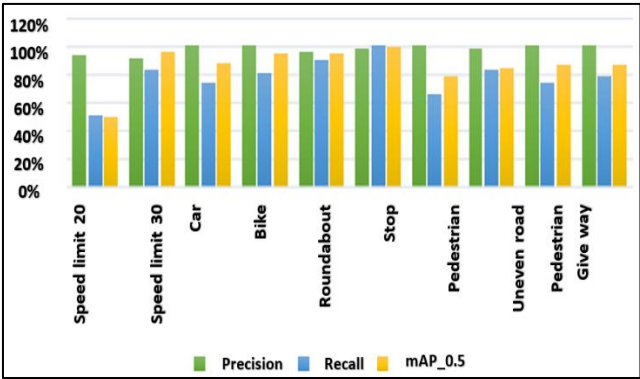


**Figure 5.** Comparison of classes based on precision, recall, and mAP
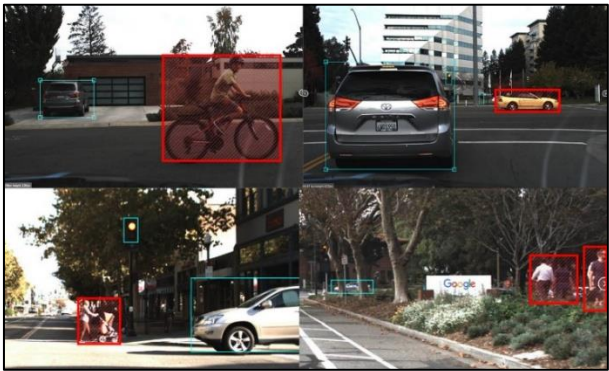


**Figure 6.** Detection of objects in various environments



**Figure 7.** Detection of the lane



**Figure 3.** Images from the dataset



| | Car | Pedestrian | Bicycle | Crossing | Give way | Roundabout | Stop | Uneven Road | Speed Limit 30 | Speed Limit 20 |
|---|---|---|---|---|---|---|---|---|---|---|
| Car | 623 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Pedestrian | 0 | 159 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bicycle | 1 | 0 | 88 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| Crossing | 0 | 0 | 0 | 65 | 0 | 0 | 0 | 0 | 0 | 0 |
| Give way | 0 | 0 | 0 | 0 | 50 | 0 | 0 | 0 | 0 | 0 |
| Roundabout | 0 | 0 | 0 | 0 | 0 | 40 | 0 | 0 | 0 | 0 |
| Stop | 0 | 0 | 0 | 0 | 0 | 0 | 35 | 0 | 0 | 0 |
| Uneven Road | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 25 | 0 | 0 |
| Speed Limit 30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 15 | 0 |
| Speed Limit 20 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 10 |

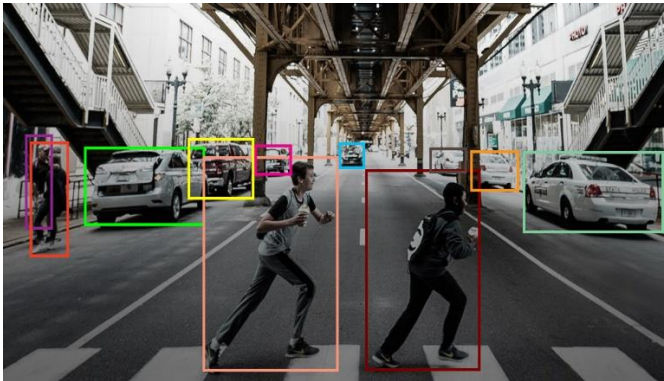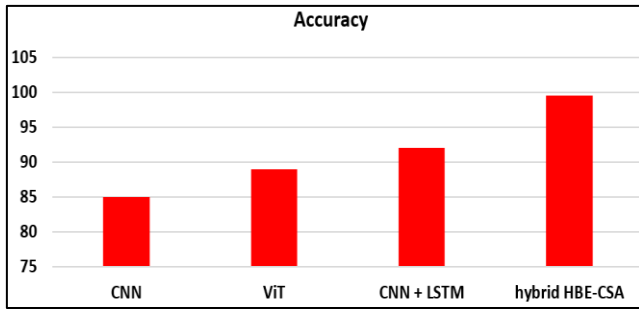**Figure 4.** The confusion matrix obtained in the process



**Figure 8.** Detection of the lane and the objects

**Figure 9.** Comparing the accuracy to various models

For every class in the HBE-CSA model, Figure 5 displays the precision, recall, and mAP@0.5 performance values. The outcomes of object detection tests conducted on movies captured in different driving situations are shown in Figure 6. Figure 7 illustrates lane detection. Figure 8 displays a picture that combines lane and object detection.

Figure 9 illustrates the identification detection accuracy of the CNN, ViT, CNN+LSTM, and proposed HBE-CSA algorithms. Together with the datasets took into consideration

in this proposed research. Table 3 displays the average results of assessing the various approaches. Figure 10 depicted as bar chart for FPS (frames per second), IoU (Intersection Over Union), Accuracy comparison of the three methods. During the testing phase for object and lane recognition, a 30 FPS result was achieved using an Intel Core i5-9400 CPU. Figure 11 shows that bar chart for HBE-CSA improves the detection accuracy and minimize tuning time. Figure 12 display that ROC curve for proposed method. Figure 13 illustrated comparison of the proposed work classification metrics. The results of the experiments conducted on the GRAZ dataset and nuScenes, which contains 1115 photographs taken in a variety of circumstances, exhibit exceptional metrics. The confusion matrix demonstrates that there is a symmetrical distribution of performance across classes, including autos, bikers, bicycles, and traffic signs. The computational feasibility of the recommended model is shown by the fact that it achieved 1000 epochs of training on Tesla T4 and testing on Intel i5; it may reach 30 frames per second. To assess the HBE-CSA model's real-time capabilities for autonomous driving deployment, we performed a latency study on several hardware platforms, as shown in Table 4.
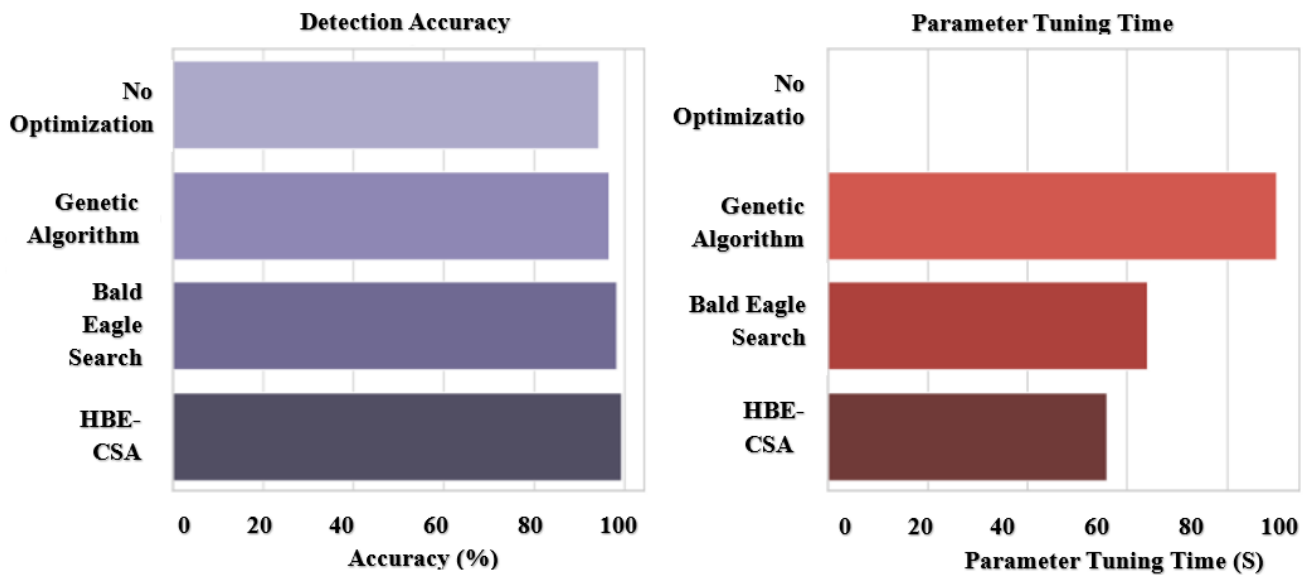


**Figure 10.** Barchart for detection accuracy, IoU, FPS comparisons
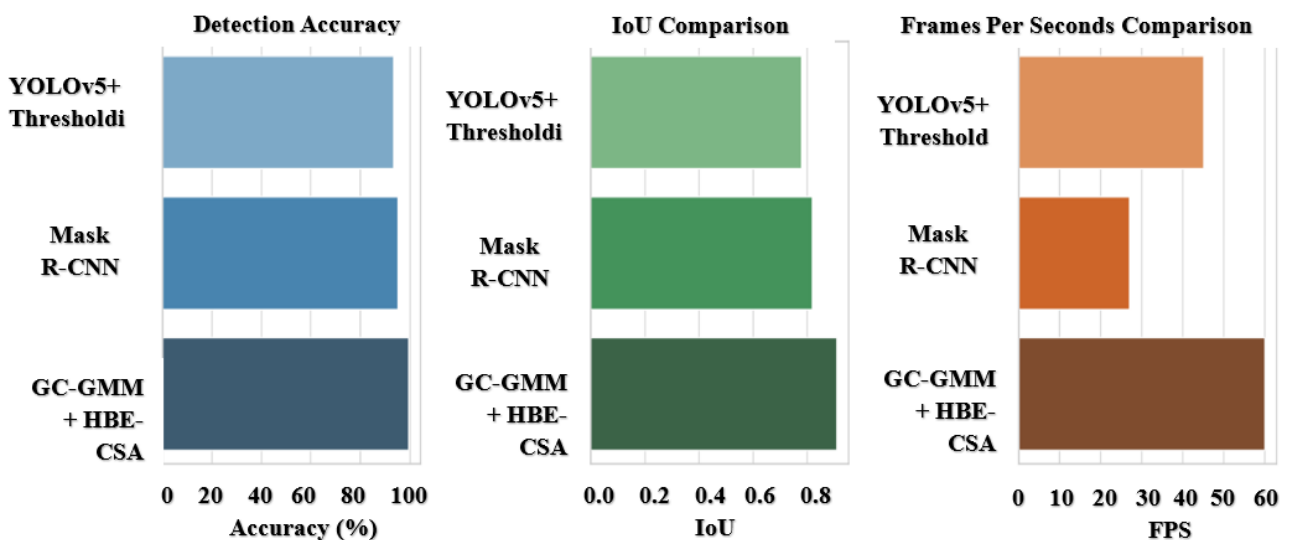


**Figure 11.** Barchart HBE-CSA improves the detection accuracy and minimize tuning time
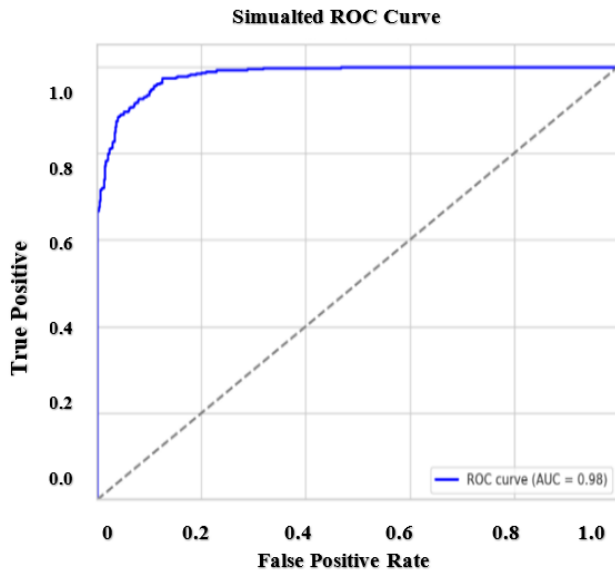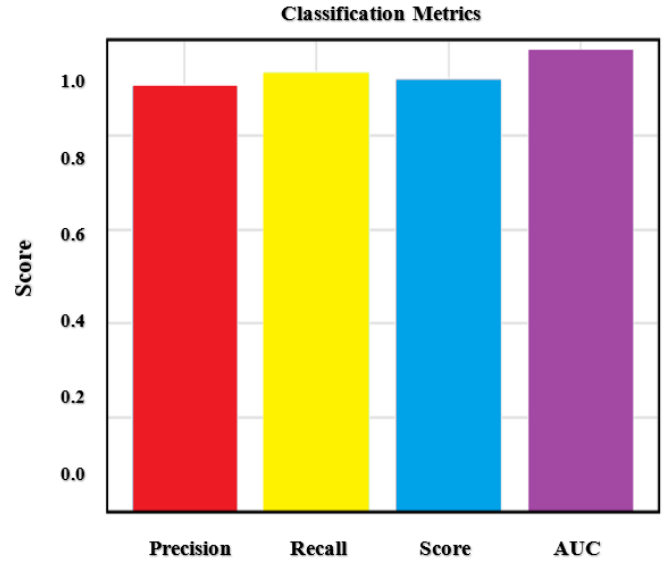
**Figure 12.** ROC curve



**Figure 13.** Comparison of the performance metrics

**Table 2.** Dataset description

| Parameter | Dataset | Description |
|---|---|---|
| Total Images | | 1115 |
| Classes | GRAZ | Car, pedestrian, bicycle, stop, pedestrian crossing, give way, roundabout, uneven road, speed limit 20, speed limit 30. |
| Total Images/scenes | | 1.4million/1000 |
| Classes | nuScenes | 500 logs of driving data, particularly focused on human, bicycle. The annotated image includes snow, rain, night-time uneven road autonomous driving applications. |

**Table 3.** Performance indicators for different approaches

| Topics of Detection | Algorithms | Accuracy Ratio |
|---|---|---|
| Real-time Object Detection in Autonomous Vehicles with YOLO [17] | YOLO Segmentation Model | 94% |
| Real-time lane detection for autonomous vehicles using YOLOV5 Segmentation Model [18] | YOLOV5 Segmentation Model | 95.15 |
| A Novel Hybrid Deep Learning Algorithm for Object and Lane Detection in Autonomous Driving [19] | LSTM-CNN | 96.8% |
| Framework for real-time object tracking, detection, and monitoring in security surveillance systems [22] | Deep learning | 85.6 % |
| Segmentation-Based CNN-Based Object Detection in Outdoor Natural Scenes [23] | CNN | 92.25% |
| Autonomous Vehicles Based on Multi-Task Learning [26] | LSTM | 93.8% |
| **Proposed Method** | HBE-CSA | 99.5% |

**Table 4.** Latency and Inference time on different platform

| Hardware Platform | Inference Time (ms/frame) | End-to-End Latency (ms) | FPS |
|---|---|---|---|
| NVIDIA Tesla T4 GPU | 10 | 15 | 100 |
| Intel Core i5-9400 CPU | 33 | 40 | 30 |
| NVIDIA Jetson Orin Nano | 16 | 20 | 62 |
| NVIDIA Jetson Nano | 50 | 60 | 20 |
| Raspberry Pi 5 | 85 | 100 | 10 |
| Raspberry Pi 4B | 1111 | 1200 | 0.9 |

## 5. CONCLUSIONS

In summary, the suggested deep learning framework successfully integrates sophisticated preprocessing, innovative segmentation through GC-GMM, and optimized detection through the Hybrid Bald Eagle-Crow Search Algorithm (HBE-CSA) to attain extremely precise (99.5%) real-time lane line and object detection in the context of autonomous driving. The combination of metaheuristic parameter adjustment, fuzzy logic, and graph-based optimization greatly improves detection reliability in a variety of scenarios. To enhance scalability and facilitate practical implementation in fully autonomous cars, the model can be addressed to accommodate dynamic impediments, cross-modal fusion, and real-time video streams and tailored for deployment on edge computing devices.

## REFERENCES

[1] Li, Y., Wang, H., Dang, L.M., Nguyen, T.N., Han, D., Lee, A., Jang, I., Moon, H. (2020). A deep learning-based

hybrid framework for object detection and recognition in autonomous driving. IEEE Access, 8: 194228-194239. https://doi.org/10.1109/ACCESS.2020.3033289

[2] Perumal, P.S., Wang, Y., Sujasree, M., Tulshain, S., Bhutani, S., Suriyah, M.K., Raju, V.U.K. (2023). LaneScanNET: A deep-learning approach for simultaneous detection of obstacle-lane states for autonomous driving systems. Expert Systems with Applications, 233: 120970. https://doi.org/10.1016/j.eswa.2023.120970

[3] Uçar, A., Demir, Y., Güzeliş, C. (2017). Object recognition and detection with deep learning for autonomous driving applications. Simulation, 93(9): 759-769. https://doi.org/10.1177/0037 549717 709932

[4] Mukhamejanov, Y., Kulambayev, B. (2024). Navigating autonomously with deep learning-powered lane line detection for vehicles. In 2024 IEEE 4th International Conference on Smart Information Systems and Technologies (SIST), Astana, Kazakhstan, pp. 296-301. https://doi.org/10.1109/SIST61555.2024.10629575

[5] Khan, M.A.M., Haque, M.F., Hasan, K.R., Alajmani, S.H., Baz, M., Masud, M., Nahid, A.A. (2022). LLDNet: A lightweight lane detection approach for autonomous cars using deep learning. Sensors, 22(15): 5595. https://doi.org/10.3390/s22155595

[6] Chen, S., Leng, Y., Labi, S. (2020). A deep learning algorithm for simulating autonomous driving considering prior knowledge and temporal information. Computer-Aided Civil and Infrastructure Engineering, 35(4): 305-321. https://doi.org/10.1111/mice.12495

[7] Kumar, S., Jailia, M., Varshney, S. (2022). A comparative study of deep learning based lane detection methods. In 2022 9th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, pp. 579-584. https://doi.org/10.23919/INDIACom54597.2022.97631 10

[8] Hemalatha, S., Banu, G., Indirajith, K. (2023). Design and investigation of PV string/central architecture for bayesian fusion technique using grey wolf optimization and flower pollination optimized algorithm. Energy Conversion and Management, 286: 117078. https://doi.org/10.1016/j.enconman. 2023.117078

[9] Xia, X., Meng, Z., Han, X., Li, H., Tsukiji, T., Xu, R., Zheng, Z., Ma, J. (2023). An automated driving systems data acquisition and analytics platform. Transportation Research Part C: Emerging Technologies, 151: 104120. https://doi.org/10.1016/j.trc.2023.104120

[10] Qian, Y., Dolan, J.M., Yang, M. (2019). DLT-Net: Joint detection of drivable areas, lane lines, and traffic objects. IEEE Transactions on Intelligent Transportation Systems, 21(11): 4670-4679. https://doi.org/10.1109/TITS.2019.2943777

[11] Johny, R.A., Thirulakshmipathi, T., Tamilmani, S., Muthu, E.M., Dhinesh, A. (2025). Real-time monitoring and alert system for earthing integrity in electrical installations. In Intelligent and Sustainable Power and Energy Systems, pp. 177-184. https://doi.org/10.1201/9781003654469-20

[12] Luo, X., Huang, Y.P., Cui, J.Y., Zheng, K. (2025). Deep learning-based lane detection for intelligent driving: A comprehensive survey of methods, datasets, challenges and outlooks. Neurocomputing, 650: 130795. https://doi.org/10.1016/j.neucom.2025.130795

[13] Ahmed, R., Elsayed, M.S., Abd-Elkawy, E.H. (2025). A novel hybrid deep learning algorithm for object and lane detection in autonomous driving. Journal of Artificial Intelligence and Technology, 5: 253-260. https://doi.org/10.37965/ jait.2025.0695

[14] Zhao, J., Wu, Y., Deng, R., Xu, S., Gao, J., Burke, A. (2025). A survey of autonomous driving from a deep learning perspective. ACM Computing Surveys, 57(10): 1-60. https://doi.org/10.1145/3729420

[15] Swain, S., Tripathy, A.K. (2024). Real-time lane detection for autonomous vehicles using YOLOV5 segmentation model. International Journal of Sustainable Engineering, 17(1): 718-728. https://doi.org/10.1080/19397038.2024.2400965

[16] Maddiralla, V., Subramanian, S. (2024). Effective lane detection on complex roads with convolutional attention mechanism in autonomous vehicles. Scientific Reports, 14(1): 19193. https://doi.org/10.1038/s41598-024-70116-z

[17] Alahdal, N.M., Abukhodair, F., Meftah, L.H., Cherif, A. (2024). Real-time object detection in autonomous vehicles with YOLO. Procedia Computer Science, 246: 2792-2801. https://doi.org/10.1016/j.procs.2024.09.392

[18] Prakash, M., Janarthanan, M., Devi, D. (2023). Multiple objects identification for autonomous car using YOLO and CNN. In 2023 7th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, pp. 597-601. https://doi.org/10.1109/ICICCS56967.2023.10142751

[19] Liu, Z., Cai, Y., Wang, H., Chen, L., Gao, H., Jia, Y., Li, Y. (2021). Robust target recognition and tracking of self-driving cars with radar and camera information fusion under severe weather conditions. IEEE Transactions on Intelligent Transportation Systems, 23(7): 6640-6653. https://doi.org/10.1109/TITS.2021.3059674

[20] Behera, S., Adhikari, M., Menon, V.G., Khan, M.A. (2024). Large model-assisted federated learning for object detection of autonomous vehicles in edge. IEEE Transactions on Vehicular Technology, 74(2): 1839-1848. https://doi.org/10.1109/TVT.2024.3404097

[21] Prasad, A.O., Mishra, P., Jain, U., Pandey, A., Sinha, A., Yadav, A.S., Kumar, R., Sharma, A., Kumar, G., Salem, K.H., Sharma, A., Dixit, A.K. (2023). Design and development of software stack of an autonomous vehicle using robot operating system. Robotics and Autonomous Systems, 161: 104340. https://doi.org/10.1016/j.robot.2022.104340

[22] Usman, M., Zaka-Ud-Din, M., Ling, Q. (2024). Enhanced encoder-decoder architecture for visual perception multitasking of autonomous driving. Expert Systems with Applications, 246: 123249. https://doi.org/10.1016/j.eswa.2024.123249

[23] Periasamy, M., Kaliannan, T., Selvaraj, S., Manickam, V., Joseph, S.A., Albert, J.R. (2022). Various PSO methods investigation in renewable and nonrenewable sources. International Journal of Power Electronics and Drive Systems, 13(4): 2498~2505. https://doi.org/10.11591/ijpeds.v13.i4. pp2498-2505

[24] Albert, J.R., Ramasamy, K., Joseph Michael Jerard, V., Boddepalli, R., Singaram, G., Loganathan, A. (2023). A symmetric solar photovoltaic inverter to improve power quality using digital pulsewidth modulation approach. Wireless Personal Communications, 130(3): 2059-2097. https://doi.org/10.1007/s11277-023-10372-w

[25] Arthi, V., Murugeswari, R. (2022). Object detection of autonomous vehicles under adverse weather conditions. In 2022 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI), Chennai, India, pp. 1-8. https://doi.org/10.1109/ICDSAAI55433.2022.10028795

[26] Gajjar, H., Sanyal, S., Shah, M. (2023). A comprehensive study on lane detecting autonomous car using computer vision. Expert Systems with Applications, 233: 120929. https://doi.org/10.1016/j.eswa.2023.120929

[27] Liu, B., Liu, G., Qiao, J. (2022). Lane and object detection algorithm based on deep learning. In 2nd International Conference on Artificial Intelligence, Automation, and High-Performance Computing (AIAHPC 2022), 123481N. https://doi.org/10.1117/12.2641319

[28] Teichmann, M., Weber, M., Zoellner, M., Cipolla, R., Urtasun, R. (2018). Multinet: Real-time joint semantic reasoning for autonomous driving. In 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, pp. 1013-1020. https://doi.org/10.1109/IVS.2018.8500504

[29] Abba, S., Bizi, A.M., Lee, J.A., Bakouri, S., Crespo, M.L. (2024). Real-time object detection, tracking, and monitoring framework for security surveillance systems. Heliyon, 10(15): e34922. https://doi.org/10.1016/j.heliyon.2024.e34922

[30] Naseer, A., Al Mudawi, N., Abdelhaq, M., Alonazi, M., Alazeb, A., Algarni, A., Jalal, A. (2024). CNN-based object detection via segmentation capabilities in outdoor natural scenes. IEEE Access, 12: 84984-85000. https://doi.org/10.1109/ACCESS.2024.3413848