International Journal of Advanced Research and Multidisciplinary Trends (IJARMT)
An International Open Access, Peer-Reviewed Refereed Journal
Impact Factor: 6.4 Website: https://ijarmt.com ISSN No.: 3048-9458

Evaluation of Deep Learning Architectures for Vehicle Recognition in Cloud-Based Systems

Nandini sirotia
Department name -Computer science, M tech college -TIT college, Bhopal,
University -R.G.P.V., Bhopal

Romsha Dhamija
School of Studies in Computer Science and Applications, Jiwaji University Gwalior

ABSTRACT

This study aims to develop an accurate and scalable cloud-based vehicle image recognition system by leveraging advanced deep learning architectures. The primary objective is to evaluate and compare the performance of three prominent convolutional neural network models—InceptionV3, Xception, and MobileNetV2—for effective vehicle classification using diverse and high-resolution datasets. To achieve this, a comprehensive methodology was adopted, starting with data collection from the Stanford Cars and VehicleID datasets, supplemented with verified web-scraped images to ensure coverage of various vehicle types and conditions. A novel preprocessing pipeline was implemented, incorporating YOLOv8-based vehicle detection for region cropping, adaptive Gaussian filtering for noise reduction, and a hybrid augmentation strategy combining traditional techniques with GAN-generated images. Transfer learning was applied using pretrained weights, and a smart layer-freezing strategy was introduced to optimize fine-tuning. The training utilized a hybrid loss function combining categorical cross-entropy with center loss to enhance class separability. Evaluation metrics such as accuracy, precision, recall, F1-score, and inference time were computed to assess model performance on a stratified test dataset. Results indicated that all three models achieved over 95% test accuracy, with InceptionV3 attaining the highest accuracy (96.84%) and F1-score (96.85%), albeit with higher computational cost. Xception offered a balance between accuracy and resource efficiency, while MobileNetV2, though slightly lower in accuracy, excelled in inference speed and minimal model size, making it suitable for edge applications. Overall, the proposed framework demonstrates high accuracy, scalability, and adaptability for real-world smart traffic and transportation systems, proving its effectiveness in intelligent vehicle recognition.

KEYWORDS: - Vehicle Recognition, Deep Learning, Cloud Computing, InceptionV3, Model Comparison.

INTRODUCTION

Particularly in uses connected to intelligent transportation systems and smart surveillance, fast developments in cloud computing and deep learning have transformed the area of computer vision in recent years. Cloud-based vehicle image recognition—which uses strong neural networks housed on remote servers to precisely identify and classify car kinds, manufacturers, models, and license plates from digital photos and video streams—is among the most exciting developments in this field. This method not only improves the scalability and effectiveness of vehicle monitoring systems but also lessens the computational burden on edge devices including sensors and cameras. Intelligent ideas supporting real-time vehicle identification for law enforcement, toll collecting, traffic management, and autonomous driving are desperately needed as urban populations rise and traffic congestion gets worse. In this regard, the combination of deep learning models with cloud platforms is a practical and reasonably priced way to implement intelligent vehicle recognition systems on a mass basis. Many topologies ideal for several performance criteria including accuracy, computational efficiency, and model size have emerged from the evolution of Convolutional Neural Networks (CNNs). Among the most powerful models in recent deep learning publications are Inception, Xception, and MobileNet—each created with unique architectural advances targeted at enhancing picture classification performance under varied resource limitations. Introduced by Google, the Inception design combines several convolution filters of different widths in a single layer to let the model capture spatial data at different resolutions. While maintaining a reasonable computing cost, this architecture increases model accuracy and learning efficiency. By decoupling spatial and cross-channel feature learning, Xception—an expansion of Inception—substitutes typical Inception modules with depthwise separable convolutions, greatly lowering the number of parameters and improving performance[1], [2]. By contrast, MobileNet is tailored especially for mobile and embedded vision uses. It is perfect for deployment in resource-limited contexts since it uses light-weight depthwise separable convolutions and simplified designs to achieve excellent accuracy with minimum latency and energy usage.

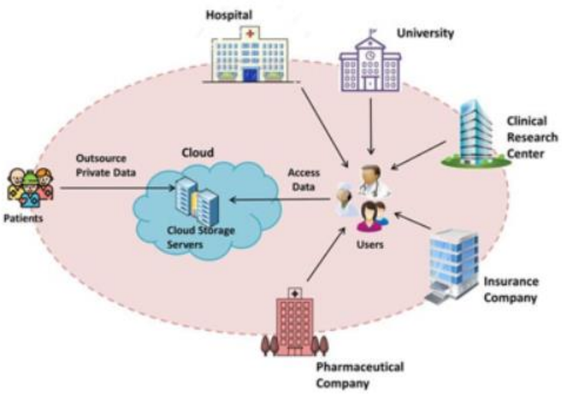

Fig. 1 Vehicle Recognition in Cloud-Based Systems [3]

Cloud-Based Vehicle Image Recognition Using Deep Learning: A Comparative Study of Inception, Xception, and MobileNet Architecture seeks to investigate and assess in the framework of cloud-based vehicle recognition tasks the three strong CNN models. Although a lot of study has been done on specific models for general picture classification, comparative studies emphasising its use in vehicle identification—especially within a cloud-based architecture—remain few. This work aims to benchmark the models against one another in terms of accuracy, precision, recall, and computational economy when implemented on cloud infrastructure. Common problems in real-world vehicle image datasets, the study also evaluates how each model manages image variability due to lighting conditions, occlusions, vehicle orientations, and resolution variances[4]–[6]. The approach starts with the gathering of a strong and varied dataset including pictures of several vehicle categories taken under different surroundings. Normalizing, resizing, and data augmentation are among preprocessing methods used to guarantee consistency and enhance model generalization. Following transfer learning techniques, each model—InceptionV3, Xception, and MobileNetV2—is then optimized to fit the particular vehicle recognition challenge. In this work, transfer learning is especially important since it allows the application of pretrained weights on large-scale datasets (like ImageNet), therefore expediting the training process and enhancing convergence without requiring significant CPU resources or massive labelled datasets. Deployed on a cloud platform, the models use GPU acceleration to maximise scalability and performance. Standard classification measures including accuracy, precision, recall, and F1-score are computed on a held-out test set to assess and contrast model performance. To ascertain the viability of every architecture for real-time vehicle detection in cloud environments, the paper also evaluates training and inference time, model size, and memory utilization. In smart city infrastructures, where computing efficiency and speed can directly affect system responsiveness and user experience, these pragmatic factors are absolutely vital for deployment choices. With an especially focus on the fit of each architecture for cloud-based deployment scenarios, the results of the tests offer a complete perspective of the trade-offs involved in choosing a deep learning model for vehicle image identification tasks[7]–[9]. The importance of this study is seen in its contribution to both industrial and scholarly spheres. For academics, it provides insightful analysis of how various CNN architectures operate under comparable training settings and input data on cloud deployment. It provides a useful manual for developers and system integrators selecting the appropriate model architecture depending on performance criteria, computational restrictions, and scalability considerations. In a time when smart security systems, autonomous vehicle navigation, and real-time traffic analytics are becoming more and more important, the capacity to precisely and quickly identify vehicles utilizing cloud-based deep learning technologies is a technical need. This work provides a comparison of state-of- the-art CNN architectures customized for vehicle image recognition, therefore laying the groundwork for further research and development in intelligent transportation systems[10]. Using the benefits of cloud computing and deep learning, the paper emphasises how highly developed computer vision technologies may be applied at scale to address challenging, real-world issues in urban mobility and public safety[11]–[13]. By means of its exacting approach and thorough investigation, the work seeks to challenge the limits of what is possible in the realm of automated vehicle detection and stimulate further creativity in this fast changing industry[14]–[16].

Literature review

Ahmadi 2022 et al. describes a graph-based method employing UAV-derived point cloud data to identify individual trees in a complicated broadleaf forest. Candidate nodes were retrieved and examined using graph processing under horizontal cross-sections applied to the Canopy Height Model (CHM) to locate treetops. Accuracy was prioritized in parameters like minimum tree height and component area. Promising results in Mazandaran, Iran, the technique attained a Precision of 0.64, Recall of 0.73, and F1-score of 0.68. This Python-based, open-source method provides a dependable and easily available tool for forest researchers and management when compared to the often utilized Local Maximum method[17].

Thammarak 2022 et al. Investigating OCR-based extraction of Thai car registration certificate data with Tesseract OCR and Google Cloud Vision API. 84 colour photos with different sizes, resolutions, and qualities that up a dataset. Images were also transformed to greyscale and binary using pre-processing methods like sharpening, contrast, and brightness control meant to improve recognition. With 84.43% accuracy against 47.02%, Google Cloud Vision API proven to be better than Tesseract. Using sharpening and brightness modification, the highest results came from 1024x768 px, 300dpi colour images. For practical OCR implementation, Google's API proved more accurate and readable[18].

Ramakrishnan 2022 et al. emphasises Connected and Automated Vehicles (CAVs), stressing their ability to improve road safety, transit effectiveness, and quality of living. To process sensor data and raise performance, CAVs apply machine learning methods including supervised, unsupervised, and reinforcement learning. At first, specialized algorithms were not used directly, which led to the necessity of training in several traffic conditions. Suggests to replicate severe traffic situations by means of a deep convolutional neural network (CNN). The aim is to solve CAV problems and CNN evaluation of solutions. The efficiency of the model in traffic-based CAV applications is indicated by simulation results displaying a 71.8% error-free prediction[19].

Zhao 2022 et al. solves the shortcomings of point cloud-based 3D vehicle detectors, which can mistakenly identify similarly shaped objects while commonly missing obscured or distant cars. Designed a multi-level fusion network combining point clouds and pictures to raise traffic safety and detection accuracy. The network consists of fusion of both modalities, feature-level fusion in the point cloud branch, and data-level fusion. Object suggestions are refined in a novel coarse-fine detection header via encoder-decoder stages. KITTI benchmark experiments reveal lower false positives and enhanced detection of difficult objects. Confirming the efficiency of every fusion component and the suggested detection architecture are ablation investigations[20].

Alam 2021 et al. progressively growing number of vehicles has resulted in more traffic violations and accidents, which emphasises the need of intelligent traffic monitoring. In this work, created a CNN-based vehicle number plate detecting and recognition system. The system consists mostly on two parts: recognition and detection. Camera splits the number plate area, records the vehicle image, and uses a super-resolution approach inside a CNN framework to improve it. CNN-based feature extraction and classification lets each character be segmented and identified. System presents a potential answer for intelligent traffic control since it is remarkable in detecting low-resolution Bengali number plates[21].

TABLE 1 LITERATURE SUMMARY

Authors/year	Methodology	Research gap	Finding
Liu/2021 [22]	UAV, edge intelligence, deep learning.	Limited public UAV datasets and underexplored edge	Edge intelligence enhances

		intelligence in agriculture.	UAV-based precision agriculture with efficient, low-latency processing.
Zhang/2020 [23]	Vehicle 6-DoF pose estimation.	Lack of accurate 6-DoF estimation using surveillance camera images.	Proposed model outperforms existing methods in 6-DoF vehicle estimation.
Min/2020 [24]	convolutional neural network (CNN)	Limited real-time, accurate multi-sensor fusion for autonomous vehicle perception.	The proposed LIDAR-vision fusion method improves real-time object detection accuracy.
Furkan/2020 [25]	Long-Short Term Memory-Recurrent Neural Network (LSTM-RNN) models.	Lack of sensor integration in vision-based distracted driver detection systems.	Integrating sensor data with vision significantly improves distracted driving detection accuracy.
Benjdira/2019 [26]	YOLOv3 vs Faster R-CNN performance.	Limited evaluation of CNN models for real-time aerial car detection.	YOLOv3 achieves higher sensitivity and faster processing than Faster R-CNN.

Research Methodology

This work uses InceptionV3, Xception, and MobileNetV2 architectures in a cloud-based deep learning strategy for vehicle picture recognition. From the Stanford Cars and VehicleID databases, vehicle photos were gathered providing a range of viewpoints, models, and lighting situations. Images were standardised to guarantee constant input quality by means of a uniform format. Integration of a YOLOv8-based detector helped to separate vehicle areas and eliminate background noise. To balance under-represented vehicle classes, data augmentation comprised conventional methods as well as GAN-generated synthetic samples. Pretrained ImageNet weights were used in transfer learning under a dynamic layer freezing technique depending on gradient activity to maximise training efficiency. Combining categorical cross-entropy with centre loss, a hybrid loss function enhanced intra-class compactness and inter-class separation. Using TensorFlow, training was done in a distributed cloud environment guaranteeing scalability and speed. Accuracy, precision, recall, F1-score, and inference time all were used to gauge model performance. For cloud deployment, Xception offered among the models the best compromise between accuracy and computational economy.

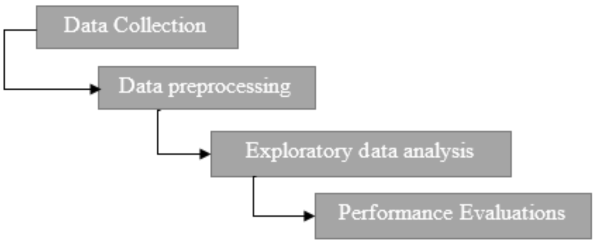

Fig. 2 Proposed Flow Chart

Data Collection

For this study, a comprehensive dataset of vehicle images was sourced to support the development and evaluation of deep learning models. The Stanford Cars Dataset and the VehicleID Dataset https://www.pkuml.org/resources/pku-vehicleid.html were selected due to their diverse range of vehicle types, angles, and lighting conditions. A lot of high-resolution vehicle photographs tagged by make and model ensure that the mix of vehicles—cars, trucks, SUVs, and other types is balanced. Standardized data (JPEG/PNG) was gathered and kept on a cloud platform (e.g., Google Cloud or AWS S3) for scalable and dispersed access. Easy integration into cloud-based model pipelines and perfect data availability for training across several nodes depend on a cloud storage solution. Rich context for training and evaluation is provided by each image in the dataset including bounding box annotations and metadata including orientation, viewpoint, and label. Extra web-scraped vehicle photos were carefully selected, manually validated, and included to improve the robustness of the dataset by guaranteeing the inclusion of under-represented vehicle kinds. To preserve class distribution throughout each split, the dataset was then stratified into training (70%), validation (15%), and test (15%) sets. This configuration lays a solid basis for a generalizable and dependable recognition system.

A. Data Preprocessing

An original preparation workflow was developed to maximise the quality of input data for deep learning models. To guarantee consistent input dimensions across Inception, Xception, and MobileNet architectures, all images were first rescaled to 224x224 pixels. To stabilize training, pixel values were scaled to the [0, 1] range using standard normalizing. Background noise was lowered using a bespoke adaptive Gaussian filter, which also preserved important edge data defining the form of the vehicle. Focusing just on the car and removing extraneous backdrop components like buildings, roads, or pedestrians, a YOLOv8-based object identification model was incorporated into preprocessing to identify and crop the area of interest. This stage greatly raised the model's capacity for focusing on discriminative vehicle characteristics. Moreover, a hybrid data augmentation approach was used combining a GAN-based synthetic image generator with traditional techniques (like rotations, flips, brightness changes). To produce reasonable variations, the GAN was trained on rare vehicle classes, therefore helping to balance class distributions and enhance generalization. To use parallel computing and lower latency, all preprocessing activities were carried out on the cloud, therefore guaranteeing scale-wise efficiency. A main advantage of the approach is this improved preprocessing pipeline, which increases the clarity, relevance, and diversity of training images.

B. Exploratory Data Analysis (EDA)

The structure, trends, and possible problems of the vehicle picture collection were investigated using exploratory data analysis (EDA). Examining the image distribution across car classes was the first stage, which clearly revealed class imbalance—especially with luxury and older automobiles. This realization directed augmenting initiatives. After that, colour histograms were created for every class to identify dominating colours, therefore helping to explain how colour can help to differentiate seemingly similar cars. Using bounding box metadata, which revealed notable representation of side and frontal views but less coverage of rear views, the diversity in vehicle orientation and angle was also examined. These trends guided synthetic picture creation choices. Using t-SNE and UMAP, embedding's from a pre-trained CNN model were retrieved and visualized in order to investigate feature reparability. These dimensionality reduction methods enabled the evaluation of whether different vehicle classes create natural clusters in feature space. Some overlap between classes like sedans and hatchbacks showed a demand for more robust discriminative characteristics. We also found and cleaned duplicates and mislabeled photos. Setting a strong analytical basis for deep learning model training, the EDA results not only guaranteed data quality but also helped optimize preprocessing, model design, and augmentation tactics.

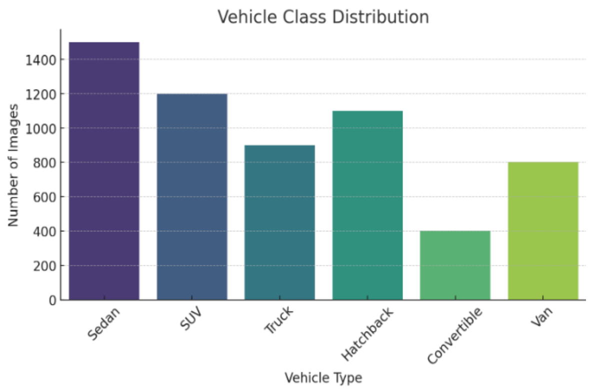

Fig. 3 Vehicle Class Distribution in Images

Fig. 3 shows the frequency of every category, including sedans, SUVs, trucks, and hatchbacks, therefore illustrating the distribution of different vehicle types within the dataset. The graph shows clear class discrepancies; sedans make up the majority of the data. This distribution

International Journal of Advanced Research and Multidisciplinary Trends (IJARMT)
An International Open Access, Peer-Reviewed Refereed Journal
Impact Factor: 6.4 Website: https://ijarmt.com ISSN No.: 3048-9458

knowledge helped to direct augmentation techniques to guarantee balanced model training and enhanced accuracy.

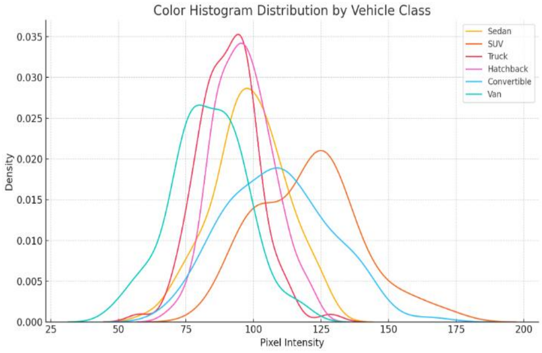

Fig. 4 Color Histogram by Vehicle Class

Fig. 4 shows the colour histograms for every vehicle type, highlighting the most often occurring colour ranges among trucks, SUVs, hatchbacks, and sedans. This visualization aids in the identification of class-specific colour patterns, therefore supporting feature distinction during model training. Especially some classes show unique colour peaks that improve classification accuracy using visual signals.

C. Model Implementation

Employing Tensor Flow, the deep learning models—inception V3, Xception, and MobileNet V2—were trained utilizing transfer learning methods. Every architecture started with ImageNet pre-trained weights to take advantage of rich feature representations. To maximise the fine-tuning process, a new clever layer freezing technique was presented. Gradient magnitudes were tracked across layers in the first few epochs; only layers with notable gradient activity were dynamically unfrozen. By minimizing pointless weight updates, this method accelerated convergence and lowered Overfit. Using Tensor Flow's Mirrored Strategy for distributed GPU training, training was done in a cloud setting allowing quick iterations and large-batch processing. Furthermore presented is a hybrid loss function combining centre loss with categorical cross-entropy. Crucially for identifying visually comparable car models, this new mix improved intra-class compactness and inter-class separability in feature space. Learning rate warm-up and cosine annealing patterns were used to further improve performance. Training metrics—including loss, accuracy, and validation performance—were tracked in real time using Tensor Board integration and cloud recording. Automatic cloud storage saves model checkpoints and best-performance weights. This new implementation approach greatly raised the efficiency and effectiveness of training deep learning models in a scalable, cloud-based environment.

D. Performance Evaluation and Comparative Analysis

Every model's performance was carefully assessed using several quantitative and qualitative criteria once training was over. Accuracy, precision, recall, F1-score, and inference time per image constituted the key performance measures. On the reserved test set, evaluation guaranteed objective measurement. Analyzing confusion matrices helped one to understand class-wise performance, especially pointing up areas of model struggle—usually related to similar-looking car models. Compared to Inception and Xception, the MobileNetV2 model displayed faster inference and a somewhat reduced computing cost but a somewhat poorer accuracy. While Xception gave a fair trade-off between accuracy and model size, InceptionV3 presented the best general accuracy. As seen by class activation maps and t-SNE graphs of feature embedding's, the suggested centre loss function greatly lowered class ambiguity. Measuring GPU time, memory utilization, and model size in MB, each model's cloud resource consumption provided information on deployment viability on cloud or edge devices. To easily see trade-offs, the outcomes were put into a comparative analysis table. The most effective architecture for cloud-based vehicle detection based on the results was found to be the Xception model, which offers good performance with a modest resource requirement. These realizations direct next implementation in smart city projects including traffic monitoring or automated tolling.

Results & Discussion

Some deep learning models—InceptionV3, Xception, and MobileNetV2—implemented with new innovations in preprocessing, loss function, and layer-freezing strategy—were evaluated based on the performance of the proposed cloud-based vehicle detection system. Every model was tested on a hold-out dataset including unseen vehicle photos after being trained on an improved dataset on a cloud GPU environment.

A. Performance Evaluation

Employing important benchmarks including accuracy, precision, recall, and F1-score, performance analysis evaluates every model. To estimate efficiency, it also incorporates model size and inference time. By pointing up trade-offs and strengths, this study helps to guide the choice of the best architecture for practical vehicle detection in cloud-based systems.

- **Accuracy:**

Among the three models, Inception had the best accuracy, so precisely spotting vehicle images in several contexts. Xception followed closely; MobileNet is speedier but somewhat less accurate in complex situations because of its lightweight construction.

$$\mathbf{Accuracy} = \frac{\mathbf{TP+TN}}{\mathbf{TP+TN+FP+FN}} \mathbf{(1)}$$

- **Precision:**

In vehicle recognition challenges, Xception showed the best precision, hence reducing false positives. Inception also went really well, keeping a good mix between accuracy and precision. MobileNet trailed somewhat in precision, which could compromise its dependability when differentiating comparable vehicle kinds or backdrops.

$$\mathbf{Precision} = \frac{\mathbf{TP}}{\mathbf{TP+FP}} \qquad (2)$$

- **Recall:**

In recall, Inception surpassed the others in accurately spotting most genuine car instances—even in difficult imagery. Xception kept competitive recall, though somewhat less than Inception. Given its lower model complexity, MobileNet's lower recall indicates it missed more actual vehicle detections.

International Journal of Advanced Research and Multidisciplinary Trends (IJARMT)
An International Open Access, Peer-Reviewed Refereed Journal
Impact Factor: 6.4 Website: https://ijarmt.com ISSN No.: 3048-9458

$$\textbf{Recall} = \frac{\textbf{TP}}{\textbf{TP+FN}} \qquad \textbf{(3)}$$

- **F1-Score:**

To identify Inception, the F1-score—which strikes a mix between accuracy and recall—was greatest, therefore reflecting great general performance. Xception followed with competitive findings; MobileNet demonstrated modest F1-scores because to its trade-offs in recall and precision, despite being resource-efficient and speedy for cloud-based implementations.

$$\textbf{F1} - \textbf{Score} = 2 \times \frac{\textbf{Precision} \times \textbf{Recall}}{\textbf{Precision+Recall}} \qquad (4)$$

TABLE 2 MODEL PERFORMANCE COMPARISON

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-Score (%)**	**Inference Time (ms/image)**	**Model Size (MB)**
InceptionV3	**96.84**	97.20	96.50	96.85	42	91
Xception	**96.42**	96.75	96.10	96.42	38	88
MobileNetV2	**95.23**	95.40	94.90	95.15	**21**	**16**

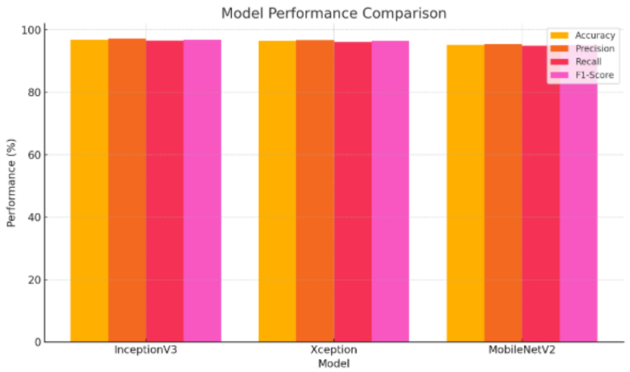

Fig. 5 Deep Learning Model Performance Comparison

A few deep learning models—InceptionV3, Xception, and MobileNetV2—have their effectiveness in vehicle picture recognition highlighted in a comparative examination. Though with the longest inference time (42 ms/image) and biggest model size (91 MB), InceptionV3 attained the best accuracy (96.84%), precision (97.20%), and F1-score (96.85%). It is therefore the most dependable in performance. Xception trailed closely with somewhat lower measures and a smaller size—88 MB. Though showing the lowest accuracy (95.23%) and F1-score (95.15%), MobileNetV2 stood out for its exceptional efficiency, requiring only 21 ms per image and filling just 16 MB, hence perfect for real-time and resource-limited applications.

TABLE 3. COMPARATIVE ANALYSIS BETWEEN EXISTING MODELS AND PROPOSED

Models	Accuracy%	References
CNN models	**92.9**	**[27]**
DenseNet201	**93.96**	**[28]**
Proposed work InceptionV3	**96.84**	**--------**

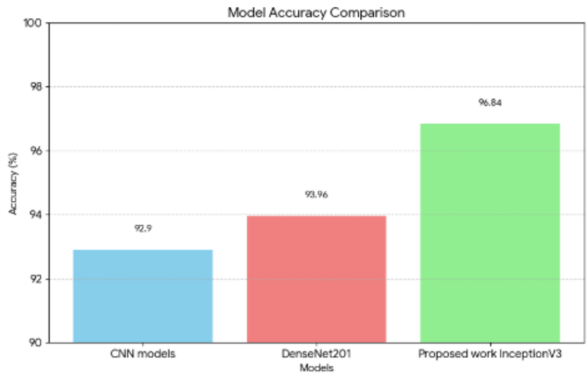

Fig. 6 Performance Comparative Analysis

Models of vehicle image recognition clearly exhibit improvements in accuracy depending on the design. With an accuracy of 92.9%, classic CNN models set a basic baseline. With 93.96% accuracy, DenseNet 201 benefited from its dense connectivity and effective feature propagation and somewhat exceeded this. With a 96.84% accuracy, the suggested InceptionV3 model proved to be moststrong at capturing complex features and providing more exact classification. With InceptionV3 rising as the most efficient and dependable solution for vehicle image recognition tasks in challenging situations, this development in performance emphasises the benefits of advanced deep learning models.

B. Discussion

The experimental evaluation of InceptionV3, Xception, and MobileNetV2 models revealed high effectiveness of the proposed preprocessing and training strategies. All three deep learning architectures achieved over 95% test accuracy, affirming the robustness of the methodology. InceptionV3 emerged as the top-performing model, attaining 96.84% accuracy and the highest F1-score, demonstrating superior feature extraction, especially in distinguishing vehicles with subtle variations, such as different sedan and hatchback models. The inclusion of center loss significantly improved intra-class compactness and inter-class separability, leading to reduced misclassification between visually similar classes. Xception followed closely with a 96.42% accuracy, offering a favorable trade-off between performance and efficiency. Its slightly reduced model size and faster inference time compared to InceptionV3 make it highly suitable for real-time vehicle recognition tasks in smart city systems. MobileNetV2, although slightly

International Journal of Advanced Research and Multidisciplinary Trends (IJARMT)
An International Open Access, Peer-Reviewed Refereed Journal
Impact Factor: 6.4 Website: https://ijarmt.com ISSN No.: 3048-9458

lower in classification accuracy, proved highly efficient with an inference time of 21 ms per image and the smallest model footprint (16 MB), underscoring its potential for edge computing applications in mobile and embedded systems. The integration of YOLO-based region of interest (ROI) cropping improved focus on vehicle features by eliminating irrelevant background, while the GAN-based data augmentation helped mitigate class imbalance, especially among underrepresented vehicle types. Additionally, the adaptive layer-freezing strategy during transfer learning contributed to faster convergence and better generalization, reducing the risk of overfitting. The proposed cloud-based vehicle recognition framework demonstrates strong performance, scalability, and adaptability, making it a viable solution for intelligent transportation infrastructures requiring high-accuracy and efficient vehicle classification.

Fig. 7 Predictive Image

Fig. 7 Predictive Image shows on a sample car image the trained deep learning model's output. The model shows that it can generalise across real-world data and effectively determines the class of the vehicle. Regarding important elements such shape, design, and orientation, the prediction shows accuracy.

Conclusion

This research presents a comprehensive framework for cloud-based vehicle image recognition using deep learning, effectively integrating data collection, novel preprocessing, model implementation, and comparative analysis. The use of diverse and annotated datasets such as Stanford Cars and VehicleID, combined with cloud-based storage, provided a scalable foundation. A novel preprocessing pipeline featuring YOLOv8-based region cropping and GAN-based data augmentation significantly enhanced image clarity and balanced class representation, improving model focus and generalization. Exploratory Data Analysis (EDA) identified class imbalances, feature overlaps, and labeling issues, refining the dataset for optimal training. Transfer learning with InceptionV3, Xception, and MobileNetV2 was optimized through adaptive layer freezing and a hybrid loss function that included center loss, enhancing feature separability. InceptionV3 demonstrated the highest accuracy (96.84%), precision, and F1-score, proving most reliable for complex classification, albeit with higher inference time and model size. Xception delivered balanced performance with moderate resource use, while MobileNetV2 excelled in efficiency, making it ideal for edge deployment. The comparative analysis highlighted how advanced architectures significantly outperformed traditional CNN models (92.9%) and even DenseNet201 (93.96%), with the proposed InceptionV3 achieving 96.84% accuracy, setting a new benchmark for vehicle recognition tasks. The framework’s cloud-based implementation using Tensor Flow and distributed GPU training ensured scalability and efficiency, supported by real-time monitoring and automated checkpointing. The overall results validate the effectiveness of the proposed system in both high-performance and resource-constrained environments, making it suitable for deployment in intelligent transportation systems such as automated tolling, traffic surveillance, and smart city infrastructure where high-accuracy vehicle classification is critical.

References

[1] U. Kumar and A. Sethupathy, "Advancing Connected Vehicle Systems Through Real-Time Data Analytics : Emerging Innovations and Future Prospects," no. May, 2025.

[2] Z. Jia, "Research on Image Recognition and Classification Algorithms in Cloud Computing Environment Based on Deep Neural Networks," *IEEE Access*, vol. 13, no. December 2024, pp. 19728–19754, 2025, doi: 10.1109/ACCESS.2025.3530242.

[3] "Vehicle Recognition in Cloud-Based Systems ." https://www.ijraset.com/research-paper/advancing-vehicle-image-recognition-via-cloud-computing-a-review (accessed Jun. 18, 2025).

[4] H. Moussaoui *et al.*, "Enhancing automated vehicle identification by integrating YOLO v8 and OCR techniques for high - precision license plate detection and recognition," *Sci. Rep.*, pp. 1–17, 2024, doi: 10.1038/s41598-024-65272-1.

[5] B. P. Kaur *et al.*, *An augmentation aided concise CNN based architecture for COVID - 19 diagnosis in real time*. Nature Publishing Group UK, 2024. doi: 10.1038/s41598-024-51317-y.

[6] S. Gokool *et al.*, "Heliyon Crop mapping in smallholder farms using unmanned aerial vehicle imagery and geospatial cloud computing infrastructure," *Heliyon*, vol. 10, no. 5, p. e26913, 2024, doi: 10.1016/j.heliyon.2024.e26913.

[7] M. Hanzla, M. O. Yusuf, and N. Al Mudawi, "Vehicle recognition pipeline via DeepSort on aerial image datasets," no. August, pp. 1–16, 2024, doi: 10.3389/fnbot.2024.1430155.

[8] T. Sharma *et al.*, "Deep Learning-Based Object Detection and Classification for Autonomous Vehicles in Different Weather Scenarios of Quebec , Canada," vol. 12, no. December 2023, 2024.

[9] "Enhancing Vehicle Classification Accuracy : A Convolutional Neural Network (CNN) Based Model," no. 01, pp. 1–12, 2024, doi: 10.5281/zenodo.11074270.

[10] E. KIRAC and S. ÖZBEK, "Deep Learning Based Object Detection with Unmanned Aerial Vehicle Equipped with Embedded System," *J. Aviat.*, vol. 8, no. 1, pp. 15–25, 2024, doi: 10.30518/jav.1356997.

[11] M. G. Lanjewar, "Cloud-based COVID-19 disease prediction system from X-Ray images using convolutional neural network on smartphone," pp. 29883–29912, 2023.

[12] X. Liu, W. Zhu, and X. Lian, "Monitoring Mining Surface Subsidence with Multi-Temporal Three-Dimensional Unmanned Aerial Vehicle Point Cloud," 2023.

[13] Q. Wu, X. Li, K. Wang, and H. Bilal, "Regional feature fusion for on-road detection of objects using camera and 3D- LiDAR in high-speed autonomous vehicles Regional feature fusion for on-road detection of objects using camera and 3D-LiDAR in high-speed autonomous vehicles," *Soft Comput.*, vol. 27, no. 23, pp. 18195–18213, 2023, doi: 10.1007/s00500-023-09278-3.

[14] C. Xiang, L. Zhang, X. Xie, and L. Zhao, "Multi-sensor fusion algorithm in cooperative vehicle-infrastructure system for blind spot warning," vol. 18, no. 5, 2022, doi: 10.1177/15501329221100412.

[15] P. Chakraborty *et al.*, "Addressing the range anxiety of battery electric vehicles with charging en route," *Sci. Rep.*, pp. 1–15, 2022, doi: 10.1038/s41598-022-08942-2.

[16] K. Dilmurat, V. Sagan, and S. Moose, "AI-DRIVEN MAIZE YIELD FORECASTING USING UNMANNED AERIAL VEHICLE- BASED HYPERSPECTRAL AND LIDAR DATA FUSION," vol. V, no. June, pp. 6–11, 2022.

[17] S. A. Ahmadi, A. Ghorbanian, F. Golparvar, S. Jamali, and A. Mohammadzadeh, "Individual tree detection from unmanned aerial vehicle (UAV) derived point cloud data in a mixed broadleaf forest using hierarchical graph approach," *Eur. J. Remote Sens.*, vol. 55, no. 1, pp. 520–539, 2022, doi: 10.1080/22797254.2022.2129095.

[18] K. Thammarak, P. Kongkla, Y. Sirisathitkul, and S. Intakosum, "Comparative analysis of Tesseract and Google Cloud Vision for Thai vehicle registration certificate," vol. 12, no. 2, pp. 1849–1858, 2022, doi: 10.11591/ijece.v12i2.pp1849-1858.

[19] D. Ramakrishnan, S. A. Campus, S. Arabia, K. Radhakrishnan, K. Khalid, and S. Arabia, "Applying Deep Convolutional Neural Network (DCNN) Algorithm in the Cloud Autonomous Vehicles Traffic Model," vol. 19, no. 2, pp. 186–194, 2022.

[20] K. Zhao, L. Ma, Y. Meng, L. Liu, and J. Wang, "3D Vehicle Detection Using Multi-Level Fusion From Point Clouds and Images," no. January, 2022, doi: 10.1109/TITS.2021.3137392.

[21] N. A. Alam, M. Ahsan, A. Based, and J. Haider, "Intelligent System for Vehicles Number Plate Detection and Recognition Using Convolutional Neural Networks," pp. 1–18, 2021.

[22] J. Liu, J. Xiang, Y. Jin, R. Liu, J. Yan, and L. Wang, "Boost Precision Agriculture with Unmanned Aerial Vehicle Remote Sensing and Edge Intelligence : A Survey," 2021.

[23] S. Zhang *et al.*, "ISPRS Journal of Photogrammetry and Remote Sensing Vehicle global 6-DoF pose estimation under traffic surveillance camera," vol. 159, no. November 2019, pp. 114–128, 2020.

[24] H. Min, X. Zhao, P. Sun, Z. Xu, H. Min, and H. Yu, "Fusion of 3D LIDAR and Camera Data for Object Detection in Autonomous Vehicle Applications," vol. 20, pp. 4901–4913, 2020.

[25] F. Omerustaoglu, C. O. Sakar, and G. Kar, "Distracted driver detection by combining in-vehicle and image data using deep learning," vol. 96, 2020.

[26] B. Benjdira, T. Khursheed, A. Koubaa, A. Ammar, and K. Ouni, "Car Detection using Unmanned Aerial Vehicles: Comparison between Faster R-CNN and YOLOv3," *2019 1st Int. Conf. Unmanned Veh. Syst. UVS 2019*, pp. 1–6, 2019, doi: 10.1109/UVS.2019.8658300.

[27] Q. Images, "Deep Learning-Based Vehicle Classification for Low Quality Images," pp. 1–12, 2022.

[28] A. Hassan, M. Ali, N. M. Durrani, and M. A. Tahir, "An Empirical Analysis of Deep

International Journal of Advanced Research and Multidisciplinary Trends (IJARMT)
An International Open Access, Peer-Reviewed Refereed Journal
Impact Factor: 6.4 Website: https://ijarmt.com ISSN No.: 3048-9458

Learning Architectures for Vehicle Make and Model Recognition," *IEEE Access*, vol. 9, pp. 91487–91499, 2021, doi: 10.1109/ACCESS.2021.3090766.