Deep Learning-Based Classification of Indian Road Vehicles Using a Custom Dataset and Pretrained Models

Bhakti Paranjape
Dr. Vishwanath Karad MIT World Peace University, Pune, India
bhakti.paranjape@mitwpu.edu.in (corresponding author)

Apurva Naik
Dr. Vishwanath Karad MIT World Peace University, Pune, India
apurva.naik@mitwpu.edu.in

Received: 27 June 2025 | Revised: 14 August 2025 | Accepted: 26 August 2025

 DOI: https://doi.org/10.48084/etasr.12979

ABSTRACT

Vehicle classification is a vital component of intelligent transportation systems, enabling applications such as traffic monitoring, automated parking, vehicle restrictions, autonomous driving, toll collection, and expressway traffic analysis. On Indian roads, a diverse range of uniquely designed vehicles—such as trucks, rickshaws, and certain bike types—poses distinct classification challenges. Traditional image processing and pattern recognition approaches, relying on handcrafted features and limited datasets, often struggled under real-world conditions affected by lighting, weather, and environmental variability. Deep learning methods have overcome many of these limitations, yet a region-specific benchmark dataset for Indian vehicles has been lacking. To address this gap, a custom dataset, the Bharatiya Vehicle Dataset (BhVD), was developed, containing four prominent vehicle categories: cars, bikes, trucks, and rickshaws. In this work, thirteen pretrained Keras models, originally trained on ImageNet, were fine-tuned and evaluated on the BhVD and two other Indian traffic datasets. The models were compared across accuracy, inference speed, and real-time applicability. The InceptionResNetV2 model achieved the highest accuracy of 94.34% on the BhVD, whereas MobileNet proved the fastest, with an inference time of 60 ms. The results demonstrate the effectiveness of transfer learning for region-specific vehicle classification and provide insights into selecting models that balance speed and accuracy for real-world deployment in Indian traffic environments.

Keywords-classification; object detection; Indian vehicles; Convolutional Neural Networks (CNNs)

I. INTRODUCTION

Object detection is a powerful technique for identifying and locating objects within an image, with key applications in autonomous vehicles and Advanced Driver Assistance Systems (ADAS) [1]. A camera mounted on the vehicle captures the traffic scene, detects various objects within it, and provides the driver with timely updates, enabling them to respond appropriately. Image classification is the starting point for object detection. In classification, a single label is assigned to represent the entire image. Object detection, however, extends this concept by identifying the different object classes present and pinpointing their positions within the image, typically by enclosing them in bounding boxes [2]. In traffic scenes, a significant proportion of the detected objects are various types of vehicles.

Machine learning enhances vehicle classification but faces challenges with similar-looking models and occlusions, which can lead to errors in camera-based systems [3]. Vehicle classification—identifying and grouping vehicles based on specific features—is vital for intelligent traffic systems, supporting applications like traffic monitoring, parking management, and vehicle type detection [4]. On Indian roads, this task is more complex due to unique vehicle types such as rickshaws, region-specific trucks, tempos, and bikes. However, the lack of benchmark datasets makes implementing and evaluating models more difficult. There are a few datasets, but they have their own limitations, as follows:

- Most datasets [5] do not include vehicles that are uniquely designed and commonly found on Indian roads, such as rickshaws, tempos, and certain types of trucks and bikes.
- The unique Indian vehicle classes are either missing or not collectively available within a single dataset [6-9].
- Even when such vehicle types are included, the images are often sourced from regions outside India, making them less representative of Indian road and traffic conditions [8, 9].

- There is currently no standard benchmark dataset dedicated to Indian vehicle classification, limiting consistent evaluation and comparison of model performance.

To address these shortcomings, a dataset, the Bharatiya Vehicle Dataset (BhVD) [10], was developed and proposed in this work. The BhVD includes four classes of Indian vehicles: cars, bikes, trucks, and rickshaws. The images were captured on roads in different parts of the city of Pune and some parts of Goa and Gujrat, using a smartphone camera. The images were then cropped to generate individual classes. It required significant man-hours for image capturing, copying, and cropping. The dataset includes 805 images, comprising 168, 171, 200, and 266 images of cars, bikes, trucks, and rickshaws, respectively. To date, there is no standard benchmark dataset available for vehicle classification specific to Indian roads [11]. Recognizing the need for a dataset covering these unique vehicle categories, the BhVD was created, and various pretrained models were implemented and compared on the proposed dataset.

In [4], the authors have identified the need for a vehicle dataset representing unique vehicle categories and developed a database of 10,000 images, where various pretrained methods are implemented and compared. The authors also considered six classes of vehicles. Authors in [12] explored ways to boost the capabilities of existing pretrained networks by integrating them with advanced image enhancement techniques, aiming to improve vehicle recognition rates. Various enhancement methods—such as Discrete Wavelet Transform (DWT), Histogram Equalization (HE), Adaptive Gamma Correction with a Weighting Distribution Function (AGCWD), Homomorphic Filtering (HF), and Joint Histogram Equalization (JHE)—were applied to strengthen feature extraction. The study utilized several Convolutional Neural Network (CNN) architectures, including residual networks (ResNet-18, ResNet-50, ResNet-101), AlexNet, GoogLeNet, DenseNet-201, and VGG-19. For classification, both the SoftMax layer of CNNs and Support Vector Machines (SVM) were employed, enabling a comparative analysis of performance.

Authors in [13] proposed a vehicle type classification method using a semi-supervised CNN from vehicle frontal-view images. A study conducted by authors in [14] compared the performance of three machine learning algorithms for vehicle classification using acoustic signals. Since different types of vehicles generate distinct sound patterns, these audio characteristics can serve as a basis for classification. After extracting relevant features from the sound data, three algorithms—K-Means, K-Means++, and Artificial Neural Networks (ANNs)—were applied to categorize the vehicles into three primary groups: bikes, cars, and trucks. In [15], an improved Faster Region-based Convolutional Neural Network (Faster RCNN) model was applied to a self-built dataset for vehicle type detection and achieved improvements in both average target detection accuracy and detection rate. Authors in [16] introduced the Vehicular Ad Hoc Networks (VANETs) for vehicle classification and reviewed their capabilities based on existing literature. The study includes a comparative analysis, revealing that VANET-based approaches surpass traditional methods in performance. Authors in [17] concluded that Deep Neural Network (DNN) methods can be effectively relied upon for vehicle category detection and classification, both in urban traffic surveillance and autonomous vehicle applications. Authors in [18] introduced the Mayfly Optimization with Deep Learning-based Robust Object Detection and Classification (MFODL-RODC) approach for processing surveillance videos. A few other researchers, such as authors in [19-21] have studied vehicle classification using different methods, including You Only Look Once version 7 (YOLOv7), and version 8 (YOLOv8).

The key contributions of this work are:

- Developed a custom dataset reflecting Indian traffic conditions, including four distinct vehicle categories: cars, bikes, trucks, and rickshaws.
- Adapted and fine-tuned multiple pretrained deep learning models from Keras Applications—originally trained on ImageNet—for vehicle classification in Indian road environments.
- Assessed the models based on accuracy, processing speed, and their feasibility for real-time implementation.
- Showcased the effectiveness of transfer learning with pretrained Keras models on region-specific datasets, enabling faster experimentation and informed model selection for traffic analytics in Indian scenarios.

II. IMPLEMENTATION

Pretrained deep learning models available in Keras Applications [22] were utilized to perform vehicle classification on the proposed dataset. The models were evaluated using 150 images per class, totaling 600 images, and their performance was compared based on accuracy, precision, recall, loss, and processing time. Figure 1 shows the block schematic of the proposed vehicle classification framework, whereas Figure 2 shows sample images from the BhVD with all four classes.

The following models were compared across various parameters; InceptionResNetV2, ResNet101V2, Xception, InceptionV3, MobileNetV2, ResNet152V2, ResNet50V2, NASNetMobile, DenseNet201, DenseNet169, MobileNet, DenseNet121, ResNet152.

The model weights were automatically fetched upon initialization. Each model had been pretrained on the standard ImageNet dataset [23], and their performance was assessed across multiple metrics, including accuracy, loss, precision, recall, and the total number of parameters. The implementation was carried out in Python using the Google Colab Pro environment. A review of the architecture of each network is tabulated in Table I.

For comparison, another dataset—the Indian Vehicle Dataset (IVD) [7]—was evaluated using the same models. This dataset consisted of only two vehicle categories: "auto" (156 images) and "truck" (196 images). The models achieved relatively low accuracy on this dataset, with results ranging between 50% and 70%. The primary reason for this poor performance was that the images were more suitable for object detection tasks rather than classification. Figure 3 illustrates sample images from the "auto" class in the IVD. To improve accuracy, the images were cropped to isolate only the relevant object class, resulting in the creation of a modified dataset named IVD_Cropped. Figure 4 presents sample cropped images from the "auto" and "truck" classes.

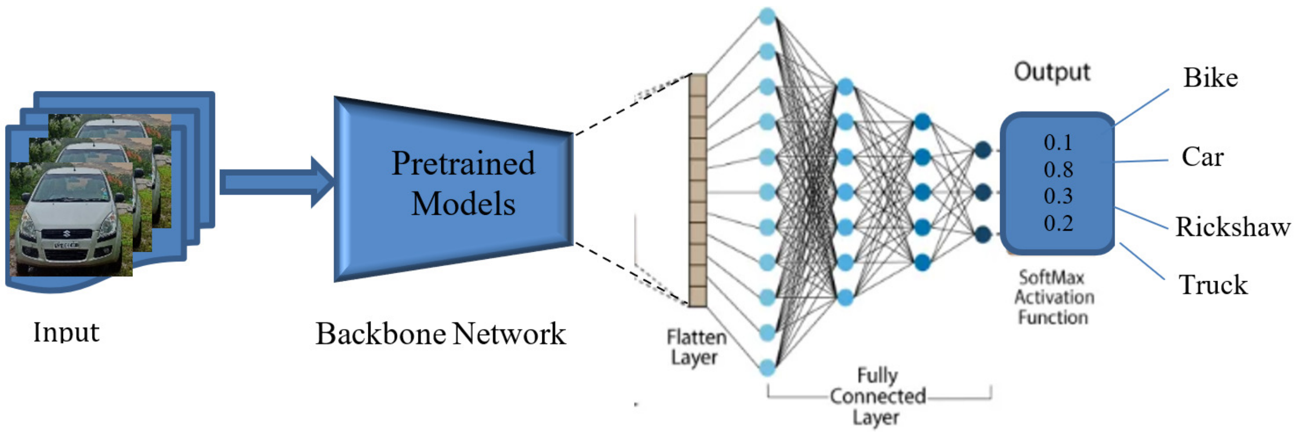

Fig. 1. Block schematic of the proposed vehicle classification process.

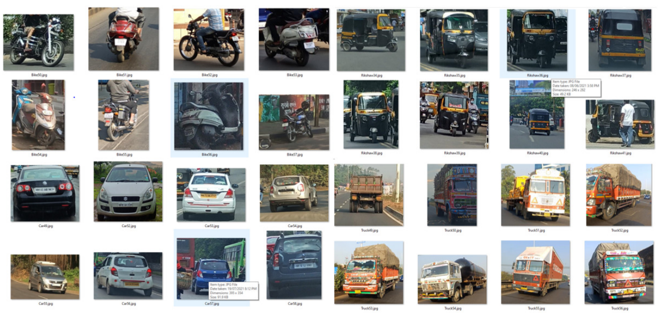

Fig. 2. Sample images from the proposed BhVD from all four classes.

Fig. 3. Sample images from the IVD for the auto class.

Fig. 4. Sample cropped images from the IVD_Cropped dataset for the auto and truck classes.

TABLE I. REVIEW OF THE ARCHITECTURE OF EACH NETWORK

Model	Key architectural features	Efficiency / complexity	Performance notes	No of layers
ResNet	Spatial resolution reduced after first 2 layers; uses residual connections	Requires 2–3 weeks training on 8 GPUs	Strong baseline; widely used in competitions	152 layers
InceptionV3	Inception modules with 12× fewer parameters than many competitors	Parameter-efficient	Won ILSVRC2014 (6.67% top-5 error)	48 layers
MobileNet	Depthwise separable convolutions	Very small model size and low computational cost	Outperforms GoogleNet and VGGNet in efficiency	53 layers
DenseNet	Concatenates feature maps from all previous layers; 1×1 convolution before 3×3 convolution for efficiency	Extra memory usage due to feature concatenation	Reduces size/complexity while maintaining accuracy	121 layers
Xception	Depthwise separable convolutions; fewer connections make it lighter	Lightweight architecture	Better results than InceptionV3	71 layers
InceptionResNetV2	Combines Inception modules with residual connections	High computational cost	Achieves the best accuracy among compared models	164 layers

III. RESULTS

Thirteen pretrained Keras models were tested across three datasets, and their accuracies were compared. The proposed BhVD consistently outperformed the other two datasets, with InceptionResNetV2 achieving the highest accuracy. As shown in Table II, the IVD produced lower classification accuracy, whereas its cropped version improved results but still fell short of the proposed dataset's performance. Figure 5 visualizes the accuracy of all models across the three datasets for easy comparison.

TABLE II. COMPARISON OF ACCURACY ON THREE DATASETS

Sr. No	Model	Proposed BhVD	IVD	IVD_Cropped
1	InceptionResNetV2	94.34	60.22	87.04
2	ResNet101V2	93.43	52.49	81.48
3	Xception	93.25	59.12	90.74
4	InceptionV3	92.88	60.22	83.33
5	MobileNetV2	91.79	56.35	75.93
6	ResNet152V2	91.61	56.91	90.74
7	ResNet50V2	91.06	59.67	83.33
8	NASNetMobile	90.69	58.01	70.37
9	DenseNet201	86.86	60.22	66.67
10	DenseNet169	86.13	60.22	74.07
11	MobileNet	85.77	54.14	79.63
12	DenseNet121	79.01	59.12	83.33
13	ResNet152	75.00	68.51	48.15

The models were also evaluated under different training parameters, with inference times (in milliseconds) summarized in Table III. MobileNet variants proved the fastest, whereas ResNet models were the slowest; the remaining models fell between these two extremes in terms of speed.

InceptionResNetV2 emerged as the best overall performer, delivering the highest accuracy and recall along with the lowest loss. MobileNetV2 offered a good balance between accuracy and speed, making it ideal for real-time use. In contrast, models such as ResNet152 and DenseNet121 performed poorly, with low recall and weaker overall results. Some models achieved high precision but low recall, meaning they were more cautious in predictions—resulting in fewer misclassifications but potentially missing many true cases. Whether this trade-off is acceptable depends on the application's requirements. Table IV presents a comparison of all models on the BhVD in terms of accuracy, precision, recall, and loss.

TABLE III. COMPARISON OF THE MODELS WITH RESPECT TO INFERENCE TIME ON BHVD, IVD, AND IVD_CROPPED DATASETS

Model	BhVD (ms)	IVD (ms)	IVD_Cropped (ms)
InceptionV3	91	98	78
ResNet50V2	98	107	85
Xception	131	134	109
ResNet152V2	213	214	194
MobileNet	60	62	47
ResNet101V2	154	170	134
ResNet152	213	214	181
DenseNet201	154	169	136
DenseNet169	128	147	110
DenseNet121	106	108	91
InceptionResNetV2	180	182	173
MobileNetV2	61	60	52

TABLE IV. ACCURACY, PRECISION, RECALL, AND LOSS OF ALL MODELS ON BHVD

Model	Accuracy (%)	Precision (%)	Recall (%)	Loss
InceptionResNetV2	94.34	92.74	83.94	0.3878
ResNet50V2	91.06	90.00	72.00	0.5158
Xception	93.25	97.17	75.18	0.5037
ResNet152V2	91.61	88.89	75.91	0.5131
MobileNet	85.77	83.91	53.28	0.7498
ResNet101V2	93.43	91.74	81.02	0.9174
ResNet152	75.00	0	0	1.3867
DenseNet201	86.86	97.1	48.91	0.7419
DenseNet169	86.13	91.78	48.91	0.7481
DenseNet121	79.01	72.00	26.28	0.9665
MobileNetV2	91.79	91.82	73.72	0.5626
NASNetMobile	90.69	91.35	69.34	0.598

Since the BhVD dataset consistently produced higher accuracy, all further experiments were conducted exclusively on this dataset. The train-test split ratio was adjusted to observe its impact on performance. Figure 6 shows the results for two split ratios: 33% and 20%. While a smaller test set is acceptable for large datasets, in the case of a smaller dataset, a larger training set is essential to ensure the model learns sufficient features from the images. Consequently, the accuracy was higher when the test set was limited to 20%.

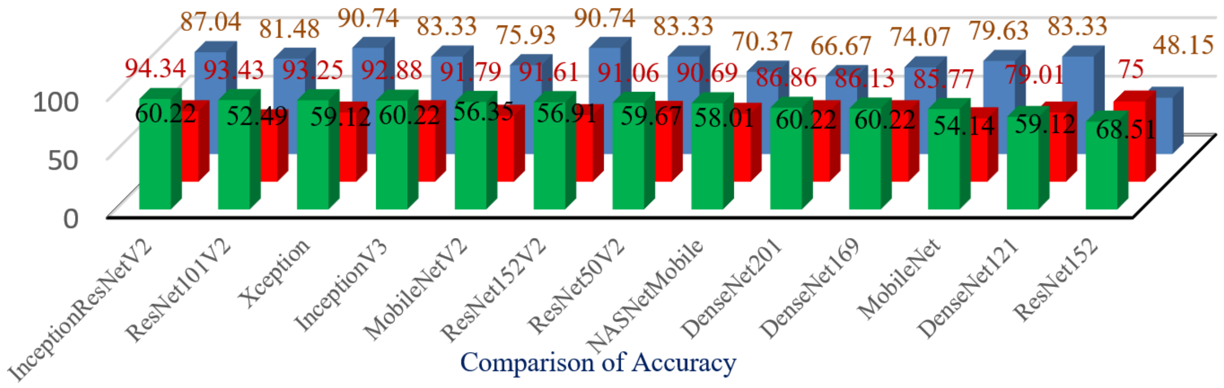

Fig. 5. Comparison of the models with respect to accuracy obtained on BhVD, IVD, and IVD_Cropped datasets.

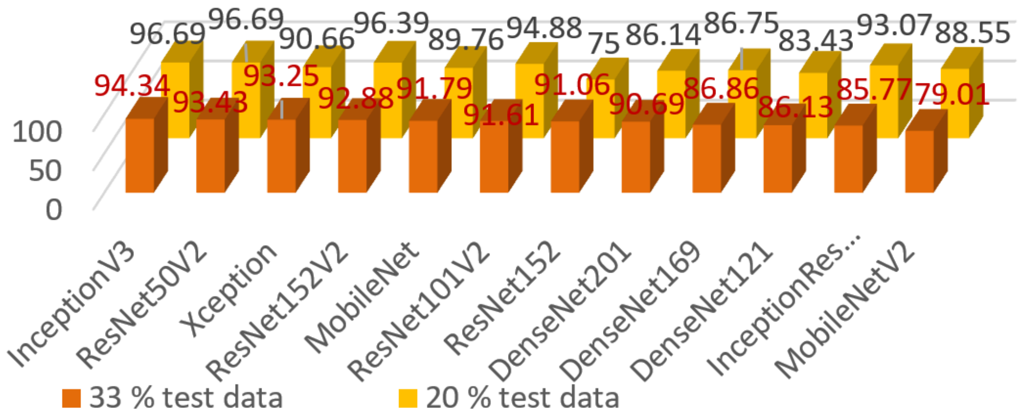

Fig. 6. Comparison of the models with respect to the training–testing split ratio on BhVD.

The total number of parameters for each model was recorded and compared with those from the ImageNet dataset to evaluate model efficiency and generalization capability. The results indicate that the pretrained models maintained comparable performance when fine-tuned on the proposed BhVD dataset, achieving similar accuracy levels to those obtained on ImageNet, with only minimal differences observed. Interestingly, the BhVD dataset required fewer learnable parameters than ImageNet, demonstrating that high classification accuracy can be achieved even with reduced computational complexity. Table V presents a detailed comparison of the top-5 accuracy and parameter counts for both datasets, whereas Figure 7 illustrates the classification model's output for the four vehicle classes considered in this study.

TABLE V. TOP-5 ACCURACY, LEARNABLE PARAMETERS, AND TRAINABLE PARAMETERS FOR BHVD AND IMAGENET DATASETS

Model	Top-5 accuracy		Learnable parameters		Trainable parameters on BhVD
	Image Net	BhVD	ImageNet	BhVD	
Xception	0.945	0.9325	22,910,480	20,869,676	8,196
ResNet152	0.931	0.7500	60,419,944	58,379,140	8,196
ResNet50V2	0.930	0.9106	25,613,800	23,572,996	8,196
ResNet101V2	0.938	0.9343	44,675,560	42,634,756	8,196
ResNet152V2	0.942	0.9161	60,380,648	58,339,844	4,100
Inception V3	0.937	0.9288	23,851,784	21,810,980	8,196
Inception ResNetV2	0.953	0.9434	55,873,736	54,342,884	8,196
MobileNet	0.895	0.8577	4,253,864	3,232,964	7,684
MobileNet V2	0.901	0.9179	3,538,984	2,263,108	6,660
DenseNet 121	0.923	0.7901	8,062,504	7,041,604	4,100
DenseNet 169	0.932	0.8613	14,307,880	12,649,540	6,148
DenseNet 201	0.936	0.8686	20,242,984	18,329,668	5,124
NASNetMobile	0.919	0.9069	5,326,716	4,273,944	4,228

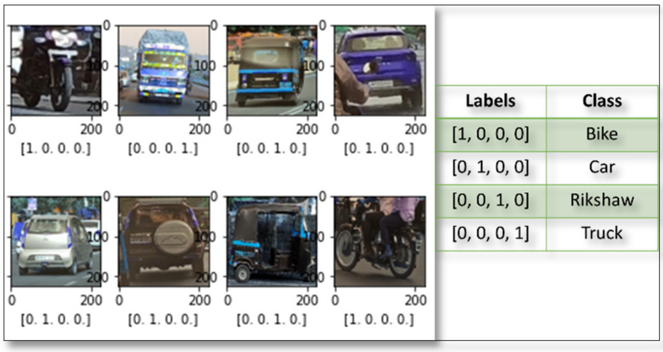

Fig. 7. Classification output for the four vehicle classes on BhVD.

IV. CONCLUSION

Keras Applications provide a convenient platform for researchers to experiment with custom datasets or explore a variety of models already trained on large-scale benchmarks such as ImageNet. These pretrained models can be quickly loaded with their existing weights and adapted to specific tasks. In this study, multiple Keras models were applied to a newly developed dataset, the Bharatiya Vehicle Dataset (BhVD), featuring four vehicle classes—cars, bikes, trucks, and rickshaws—tailored to the Indian traffic environment. The models were evaluated across several performance metrics and compared on three datasets representing Indian road conditions. Among the tested models, InceptionResNetV2 delivered the highest accuracy of 94.34% on BhVD, whereas MobileNet emerged as the fastest, achieving an inference time of just 60 ms. This work highlights the effectiveness of transfer learning for region-specific vehicle classification and provides insights into selecting models that balance accuracy and speed for real-world traffic applications.

REFERENCES

[1] 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, Mar. 2024, Art. no. 103, https://doi.org/10.3390/a17030103.

[2] S. Cao, "Review of Object Detection Challenges in Autonomous Driving," *Applied and Computational Engineering*, vol. 8, no. 1, pp. 707–713, Aug. 2023, https://doi.org/10.54254/2755-2721/8/20230306.

[3] M. Won, "Intelligent Traffic Monitoring Systems for Vehicle Classification: A Survey," *IEEE Access*, vol. 8, pp. 73340–73358, 2020, https://doi.org/10.1109/ACCESS.2020.2987634.

[4] M. A. Butt *et al.*, "Convolutional Neural Network Based Vehicle Classification in Adverse Illuminous Conditions for Intelligent Transportation Systems," *Complexity*, vol. 2021, no. 1, Feb. 2021, Art. no. 6644861, https://doi.org/10.1155/2021/6644861.

[5] "Vehicle Classification - Dataset." Kaggle. [Online]. Available: https://www.kaggle.com/datasets/maciejgronczynski/vehicle-classification-dataset.

[6] M. Maher, "Vehicle Image Classification." Kaggle. [Online]. Available: https://www.kaggle.com/datasets/mohamedmaher5/vehicle-classification.

[7] "Indian Vehicle Dataset." Kaggle. [Online]. Available: https://www.kaggle.com/datasets/dataclusterlabs/indian-vehicle-dataset.

[8] "Vehicle Type Recognition." Kaggle. [Online]. Available: https://www.kaggle.com/datasets/kaggleashwin/vehicle-type-recognition.

[9] Z. Luo *et al.*, "MIO-TCD: A New Benchmark Dataset for Vehicle Classification and Localization," *IEEE Transactions on Image Processing*, vol. 27, no. 10, pp. 5129–5141, Oct. 2018, https://doi.org/10.1109/TIP.2018.2848705.

[10] B. Paranjape, "Bharatiya Vehicles Dataset (BhVD)." Kaggle, 2025, https://doi.org/10.34740/kaggle/dsv/12771309.

[11] G. Varma, A. Subramanian, A. Namboodiri, M. Chandraker, and C. V. Jawahar, "IDD: A Dataset for Exploring Problems of Autonomous Navigation in Unconstrained Environments," in *2019 IEEE Winter Conference on Applications of Computer Vision*, Waikoloa, HI, USA, 2019, pp. 1743–1751, https://doi.org/10.1109/WACV.2019.00190.

[12] V. K. Kiran, S. Dash, and P. Parida, "Improvement on Deep Features through Various Enhancement Techniques for Vehicles Classification," *Sensing and Imaging*, vol. 22, no. 1, Sep. 2021, Art. no. 41, https://doi.org/10.1007/s11220-021-00363-1.

[13] B. Hicham, A. Ahmed, and M. Mohammed, "Vehicle Type Classification Using Convolutional Neural Network," in *2018 IEEE 5th International Congress on Information Science and Technology*, Marrakech, Morocco, 2018, pp. 313–316, https://doi.org/10.1109/CIST.2018.8596500.

[14] S. D. Badiger and M. UttaraKumari, "Vehicle Classification Using Machine Learning Algorithms Based on the Vehicular Acoustic Signature," *Science, Technology and Development*, vol. 8, no. 11, pp. 369–374, Nov. 2019.

[15] L. Suhao, L. Jinzhao, L. Guoquan, B. Tong, W. Huiqian, and P. Yu, "Vehicle type detection based on deep learning in traffic scene," *Procedia Computer Science*, vol. 131, pp. 564–572, Jan. 2018, https://doi.org/10.1016/j.procs.2018.04.281.

[16] H. Shokravi, H. Shokravi, N. Bakhary, M. Heidarrezaei, S. S. Rahimian Koloor, and M. Petrů, "A Review on Vehicle Classification and Potential Use of Smart Vehicle-Assisted Techniques," *Sensors*, vol. 20, no. 11, Jun. 2020, Art. no. 3274, https://doi.org/10.3390/s20113274.

[17] D. Sharma, Z. A. Jaffery, and N. Ahmad, "Categorical vehicle classification using Deep Neural Networks," in *2019 International Conference on Power Electronics, Control and Automation*, New Delhi, India, 2019, pp. 1–6, https://doi.org/10.1109/ICPECA47973.2019.8975613.

[18] V. Saikrishnan and M. Karthikeyan, "Mayfly Optimization with Deep Learning-based Robust Object Detection and Classification on Surveillance Videos," *Engineering, Technology & Applied Science Research*, vol. 13, no. 5, pp. 11747–11752, Oct. 2023, https://doi.org/10.48084/etasr.6231.

[19] 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, https://doi.org/10.1109/ACCESS.2024.3354076.

[20] A. Gholamhosseinian and J. Seitz, "Vehicle Classification in Intelligent Transport Systems: An Overview, Methods and Software Perspective," *IEEE Open Journal of Intelligent Transportation Systems*, vol. 2, pp. 173–194, 2021, https://doi.org/10.1109/OJITS.2021.3096756.

[21] Md. A. Hossain, Md. A. Rahman, R. Khanom, S. Sultana, and A. Hossain, "Real-time Vehicle Detection and Classification on the Padma Multipurpose Bridge in Bangladesh Using a Deep Learning Model," in *2023 International Conference on Next-Generation Computing, IoT and Machine Learning*, Gazipur, Bangladesh, 2023, pp. 1–6, https://doi.org/10.1109/NCIM59001.2023.10212549.

[22] "Keras documentation: Keras Applications." Keras. https://keras.io/api/applications/.

[23] O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, Dec. 2015, https://doi.org/10.1007/s11263-015-0816-y.