International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Real Time Classification of Vehicle Images with MLP & CNN

Syed Farhana Sultana, Chidurala Mukesh, Puppala Hemanth, Challa Ajay, K.Veeranjanrya Varaprasad

Students, Department of AIML. Malla Reddy Engineering College, Maisammaguda, Hyderabad-500100
[e]Asst. Professor, Department of IT. Malla Reddy Engineering College, Maisammaguda, Hyderabad-500100

ABSTRACT

The goal of this research is to develop a sophisticated machine learning model that can categorize vehicle photos into seven groups, including buses, trucks, motorbikes, and cars. Multi-Layer Perceptrons (MLP) and Convolutional Neural Networks (CNN) are two neural network types that are intended to enhance traffic monitoring and surveillance systems. To begin, the dataset must be examined and visualized in order to find important characteristics that aid in differentiating between vehicle kinds, such as size, shape, and color. In contrast to the CNN, which is intended for image processing and uses convolutional layers to extract significant spatial data, the MLP is a straightforward neural networkThe study will include data augmentation techniques, such flipping, resizing, and rotating photos, to increase the model's adaptability to new input and boost speed. Regularization strategies like dropout and L2 regularization will help avoid overfitting, while hyperparameter tuning will optimize variables like learning rates and batch sizes for improved training. The objective is to demonstrate how CNNs' ability to extract intricate spatial information allows them to perform significantly better than MLPs in the categorization of vehicle images. Additionally, the study will show how deep learning models may be enhanced by methods like regularization and data augmentation. The results will be helpful in the development of precise and effective vehicle categorization systems, which will find use in security monitoring, autonomous cars, and traffic management.

Keywords: Vehicle image classification, Multi-Layer Perceptron (MLP), Convolutional Neural Network (CNN), traffic monitoring, dataset exploration, data augmentation.

I. INTRODUCTION

Vehicle Image Classification is an important application of computer vision, wherein the aim is to detect and classify vehicles in images into various classes like cars, trucks, motorcycles, and buses. The process is commonly used in several real-world systems like autonomous vehicles, traffic monitoring, and smart city systems. The classification is typically done through two main steps: vehicle detection followed by classifying it into particular categories. This task proves to be rather difficult owing to the presence of factors like different vehicle types, different environmental situations, occlusions, and real-time processing requirements. To overcome these issues, Machine Learning (ML) algorithms like Multi-Layer Perceptron (MLP) and Convolutional Neural Networks (CNN) are adopted extensively. An MLP is a sort of neural network that propagates the data via multiple layers of neurons and is mostly utilized for classification. MLP models are typically employed for less complex, simpler tasks or where feature extraction is easy. Nonetheless, for the task of classifying vehicle images, CNNs tend to perform better because they can automatically learn hierarchical features from images. CNNs are made up of convolutional layers that run numerous filters to identify various patterns such as edges, textures, and shapes, making them particularly ideal for image-based applications. CNNs perform well in representing spatial relationships of the images, which are significant in differentiating between various vehicle types and adapting to size changes, orientation, and occlusions. In classification of vehicle images, MLP may be a component of the architecture of the model, usually utilized after a CNN has processed the features of the image. The CNN processes the complex data of the image, and the MLP is utilized in making the ultimate classification. By the integration of CNN for feature learning and MLP for classification, better accuracy and treatment of the difficulty imposed by multiform vehicle classes and environmental scenarios can be obtained. With powerful techniques such as transfer learning (using pre-trained models), data augmentation, and real-time computing, these models are capable of obtaining remarkable outcomes, making applications in autonomous vehicles, traffic flow, and car tracking systems more secure and intelligent. This synergy facilitates better and more efficient vehicle classification. Methods such as transfer learning, data augmentation, and real-time processing are also employed to enhance the model's accuracy and stability, particularly in dynamic and variable environments. Vehicle image classification has practical applications in autonomous cars, intelligent traffic management, surveillance security, and automated toll payment, where accuracy and real-time processing are vital for safety and efficiency.

II. RELATED WORK

In recent years, vehicle image classification has become a crucial research area due to its applications in intelligent transportation systems, autonomous driving, and traffic surveillance. Similar to how machine learning algorithms predict student performance in online education, vehicle classification benefits from deep learning to enhance recognition accuracy. Challenges such as varying lighting conditions, occlusion, and complex backgrounds have led researchers to explore advanced techniques. Convolutional Neural Networks (CNNs) have emerged as a dominant tool for vehicle classification due to their ability to learn hierarchical features. The Field Analysis Method (FAM), inspired by Item Response Theory (IRT) in education, has been applied to assess classification difficulty, much like predicting student performance in complex tasks. The relationship between vehicle characteristics such as color, shape, and make/model plays a vital role in classification performance. Machine learning algorithms extract features from images to predict vehicle types and characteristics. Similar to how learning analytics (LA) track student progress in online courses, vehicle classification models use data from high-resolution images, sensors, and environmental context (e.g., traffic conditions, time of day) to improve accuracy. These data sources train machine learning models to enhance predictions, especially in complex urban environments. Techniques like data augmentation, involving image transformations, help models generalize better, akin to refining learning analytics models in MOOCs. Transfer learning has become essential in vehicle image classification, especially when labeled data is scarce. Pre-trained models on large datasets can be fine-tuned to recognize specific vehicle types, similar to predictive models in MOOCs that estimate student outcomes with limited input. Generative Adversarial Networks (GANs) have been employed to generate synthetic training data, addressing the challenge of dataset limitations, much like simulating student behavior for improved educational predictions. Hybrid models combining CNNs with algorithms like Support Vector Machines (SVM) or k-Nearest Neighbors (k-NN) enhance classification accuracy by leveraging the strengths of multiple techniques, mirroring ensemble methods in educational analytics. The integration of multiple sensors, such as cameras, LiDAR, and radar, has improved vehicle classification, particularly in autonomous systems where image clarity may be compromised by weather or lighting. Data fusion techniques combine information from different sensors to enhance classification accuracy, similar to multi-source data integration in learning analytics. Despite advancements, challenges persist in classifying obstructed vehicles or those viewed from difficult angles. Real-time processing for autonomous vehicles and smart city applications remains a challenge. However, with ongoing research and innovation, machine learning techniques are expected to improve vehicle classification, much like the continuous enhancement of predictive models in MOOCs to adapt to diverse learning environments.

III. PROPOSED METHODOLOGY

The system of vehicle image classification proposed here uses deep learning methods, that is, CNN and MLP models, to attain high accuracy in the classification of 24 categories of vehicles. Deployed on Python with TensorFlow, Keras, and Scikit-learn, the system is based on a supervised learning methodology. The system consists of data preprocessing, augmentation, training of the model, and assessment. Images are resized, normalized, and augmented using methods such as rotation and flipping. CNN's design, including convolutional, pooling, and fully connected layers, supports feature extraction and classification. Patterns are identified by filters, dimensionality is reduced by pooling, and fully connected layers translate features to classes. Backpropagation adjusts weights, improving efficiency and generalization in large-scale classification.

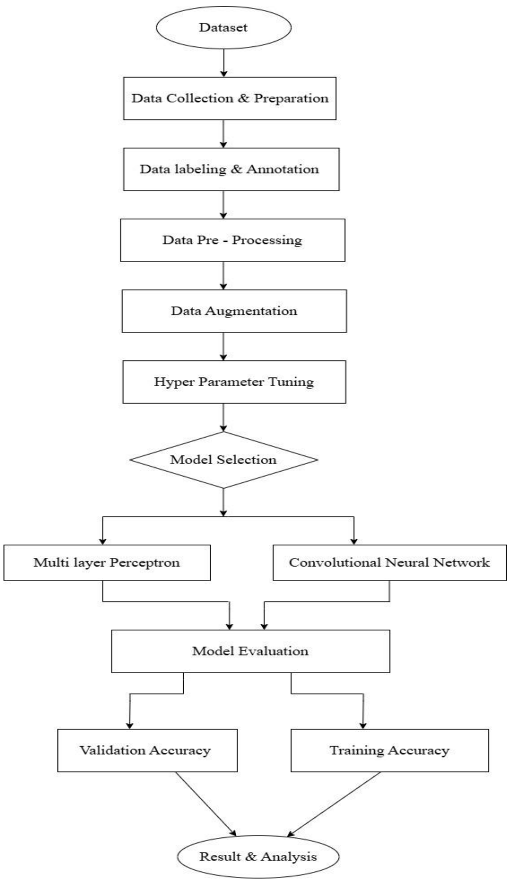


Figure 1: Demonstration of Proposed System

3.1. FEATURE EXTRACTION

Feature extraction is a critical step in vehicle image classification, where meaningful patterns and attributes are extracted from raw images to improve model performance. Convolutional Neural Networks (CNN) and Multi-Layer Perceptron (MLP) algorithms play a significant role in extracting and learning features from vehicle images. CNN excels in capturing spatial hierarchies, edges, and textures, while MLP processes the extracted features for classification. MLP is a feedforward neural network that processes the extracted features from CNN for classification. The extracted feature vector is **flattened** and passed through multiple dense layers with activation functions like ReLU (Rectified Linear Unit) and softmax. The **ReLU activation function**, which introduces non-linearity, is defined as

3.2. CONVOLUTION NEURAL NETWORK

CNN uses multiple layers, such as convolutional layers, pooling layers, and fully connected layers, to extract relevant features from images. The convolutional layer applies multiple filters (kernels) to the input image, detecting low-level features like edges and textures in initial layers and more complex patterns like shapes and vehicle parts in deeper layers. The convolution operation is mathematically defined as:

$$F(i,j) = \sum_{m}\sum_{n} I(i+m, j+n) \cdot K(m,n)$$

Where:

I (i , j) is the input image matrix,

K(m , n) is the filter (kernel) matrix,

F (i , j) is the feature map output at pixel position (i , j).

The pooling layer reduces spatial dimensions while retaining significant information, improving computational efficiency and reducing overfitting. Max pooling, commonly used in CNN, selects the highest value from a defined window, given by:

$$P(i,j) = \max_{m,n}(F(i+m, j+n))$$

3.3. MULTI PLAYER PERCEPTRON (MLP)

MLP is a feedforward neural network that processes the extracted features from CNN for classification. The extracted feature vector is **flattened** and passed through multiple dense layers with activation functions like ReLU (Rectified Linear Unit) and softmax. The **ReLU activation function**, which introduces non-linearity, is defined as:

$$f(x) = \max(0, x)$$

For multi-class classification, the **softmax activation function** is used to convert the output into probability scores for different vehicle categories. It is defined as:

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{N} e^{z_j}}$$

Where :

zi is the input to the softmax function,

N is the total number of output classes (vehicle categories).

By integrating CNN for **feature extraction** and MLP for **classification**, the system achieves robust vehicle image classification with high accuracy. CNN extracts essential spatial features, while MLP maps them to specific vehicle categories, ensuring effective recognition.

3.4. EVALUATION METRICS

Evaluating the performance of a vehicle image classification model using **Convolutional Neural Networks (CNN)** and **Multi-Layer Perceptron (MLP)** requires various metrics to assess its accuracy, robustness, and efficiency. Below are the key evaluation metrics used for vehicle image classification:

Table. 1. The performance metrics used for classification and regression

Metric	Formula
Validation Accuracy (VA)	$\frac{TP + TN}{TP + TN + FP + FN}$
Validation Loss (VL)	$\text{Loss} = -(y_{\text{true}} \cdot \log(y_{\text{pred}}) + (1 - y_{\text{true}}) \cdot \log(1 - y_{\text{pred}}))$
Training Accuracy(TA)	$\frac{TP + TN}{TP + TN + FP + FN}$
MSE Loss	$\frac{1}{n}\sum(y_{\text{true}} - y_{\text{pred}})^2$
CES	$(y_{\text{true}} \cdot \log(y_{\text{pred}}) + (1 - y_{\text{true}}) \cdot \log(1 - y_{\text{pred}}))$

IV.RESULT AND DISCUSSION

The dataset of vehicle images includes a total of 15,645 color images, organized into 7 distinct classes. These classes are as follows: **City Car**, which comprises micro cars, hatchbacks, and city cars; **Sedan**, covering both sedans and coupes; **Multi-Purpose Vehicle (MPV)**, which includes only MPVs; **Sport Utility Vehicle (SUV)**, consisting solely of SUVs; **Van**, which encompasses vans and campervans; **Truck**, covering medium-sized trucks; and **Big Truck**, which includes large-sized trucks.This dataset was utilized to develop predictive models for real-time vehicle image classification. Out of the total dataset, 10,000 images were selected and further split into 7,000 images for training, 1,500 for validation, and 1,500 for testing. Initially, the dataset contained various features and image categories representing different types of vehicles. To enhance computational efficiency and optimize performance, the dataset was refined through preprocessing and feature engineering. Key preprocessing techniques involved image resizing, normalization, and data augmentation, including operations such as rotation, flipping, and zooming, to enhance the model's generalization and robustness.

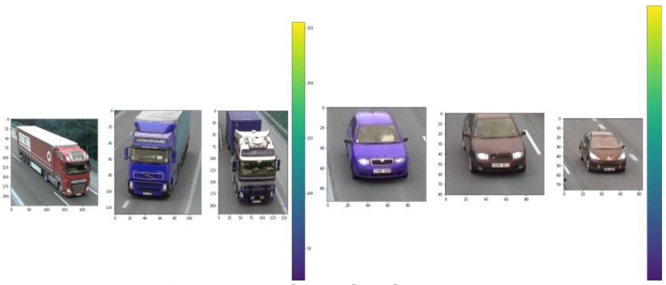


Figure 2: Sample Truck and Car Dataset

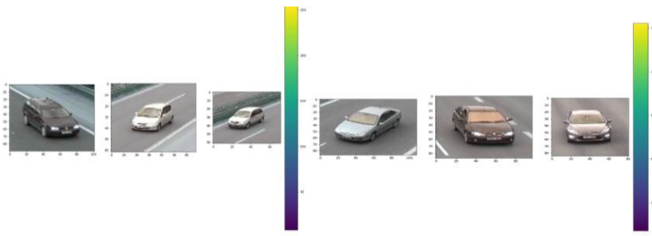
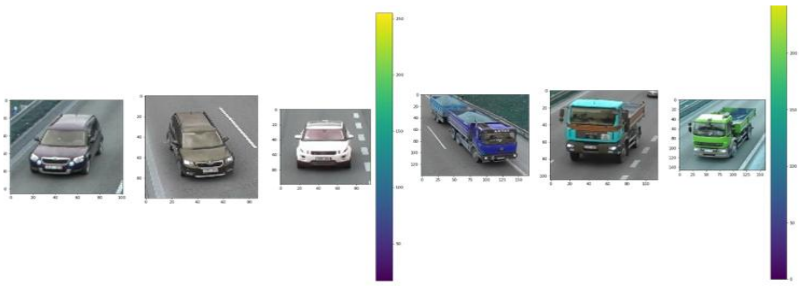


Figure 3: Sample Multi purpose vehicle & Car Dataset



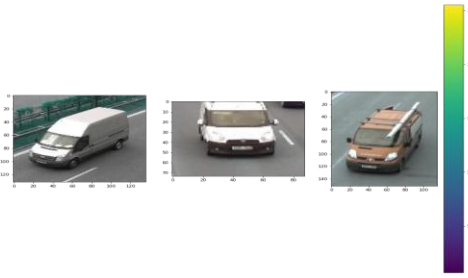


Figure 5: Sample van Dataset

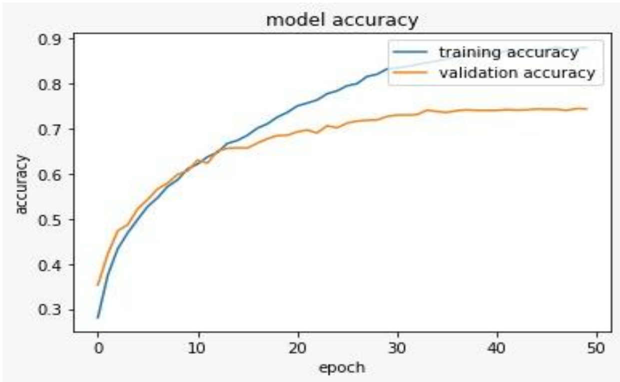


Fig .6. Model Accuracy of MLP

The Fig 6 depicts the accuracy progression of an MLP (Multi-Layer Perceptron) model over 50 epochs. The **x-axis** indicates the number of epochs, representing full passes through the training data, while the **y-axis** reflects the accuracy, which measures the proportion of correctly classified samples. The **blue line** shows the training accuracy, which rises consistently as the model improves its performance on the training data. The **orange line** represents the validation accuracy, which also increases in the initial epochs but plateaus around the 20th epoch, suggesting diminishing improvement on unseen data. The relatively small gap between the training and validation accuracy indicates that the model is learning effectively with only mild signs of overfitting, thus maintaining a reasonable balance between training performance and generalization.

Table-2. Analysis of Accuracy Graph of MLP

Phase	Training Accuracy	Validation Accuracy	Interpretation
Start (First Few Epochs)	Low & Increasing	Low & Increasing	The model is learning effectively.
Midway (Epochs 10–30)	Continues rising	Matches training temporarily	Ideal learning stage.
End (Epochs 30–50)	87.71%	74.29%	Overfitting detected.

Table 2 during the initial epochs, both training and validation accuracy increase significantly, suggesting effective learning by the model. As training progresses, the training accuracy continues to improve, while the validation accuracy stabilizes around epochs 30-40—a typical pattern. The gap that forms between the two accuracies may indicate overfitting, where the model excels at learning the training data but struggles to generalize to new data.

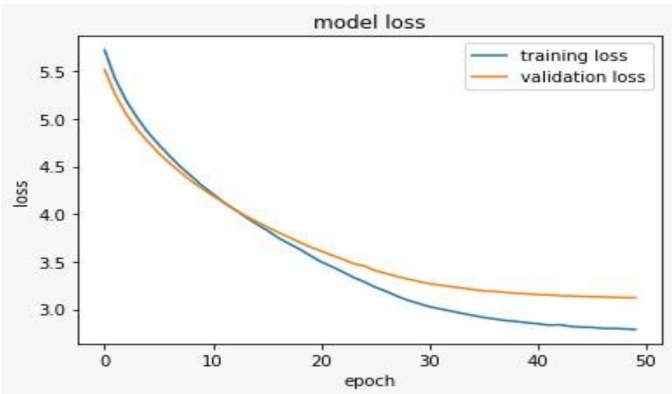


Figure .7. Model Loss of MLP

The Fig 7 illustrates the training and validation loss of an MLP model across 50 epochs, where the x-axis represents the number of epochs and the y-axis measures the loss, reflecting how inaccurate the model's predictions are. The blue line (training loss) decreases steadily, indicating that the model is improving its accuracy on the training dataset. Similarly, the orange line (validation loss) drops significantly at first, demonstrating effective learning, but it flattens around the 30th epoch. This trend indicates that while the model is learning effectively, the gap between the two lines and the plateau in validation loss may suggest early signs of overfitting as training continues.

Table .3. Analysis of the Loss Graph of MLP

Phase	Training Loss	Validation Loss	Interpretation
Start (First Few Epochs)	High & Decreasing	High & Decreasing	Model is learning effectively.
Midway (Epochs 10–30)	Continues decreasing	Close to training loss	Good generalization.
End (Epochs 30–50)	Lower than validation loss	Stops decreasing	Overfitting detected.

From Table-3 in the early epochs, both training and validation loss drop significantly, corresponding to the observed increase in accuracy. As training progresses, the training loss continues to decline, while the validation loss decreases initially but then levels off around epochs 30-40 and may slightly rise afterward, further indicating potential overfitting. Based on Tables 1 and 2, the following key observations can be made that, in Initial Phase (First Few Epochs) training and validation losses are high but decreasing and both accuracies are low and gradually increasing, showing the model is effectively learning basic patterns. In Midway Phase (Epochs 10–30), **t**raining loss continues to decrease while validation loss remains close, indicating good generalization. Training accuracy rises steadily, and validation accuracy briefly matches it, reflecting an ideal learning stage and in end phase (Epochs 30–50) , Training loss drops below validation loss, signaling overfitting. Training accuracy reaches 87.71%, while validation accuracy lags at 74.29%, confirming the model overfits the training data at later stages.

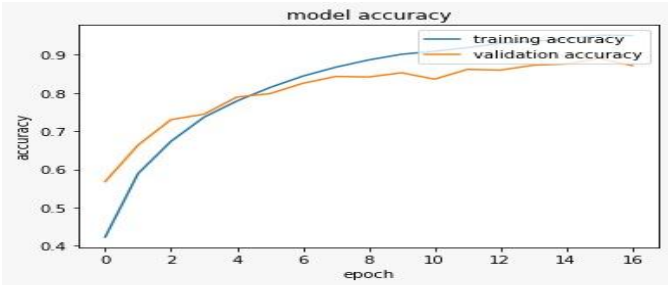


Fig .8. Model Accuracy of CNN

The Fig 8 shows the training and validation accuracy of a CNN model across 16 epochs. The **x-axis** represents the number of epochs, while the **y-axis** indicates the accuracy, measured as the percentage of correctly classified images. The **blue line** (training accuracy) increases consistently, reflecting the model's learning progress, eventually surpassing 90% accuracy. Meanwhile, the **orange line** (validation accuracy) rises quickly during the initial epochs and then plateaus around 80%, with slight fluctuations. The narrow gap between the lines suggests effective generalization and minimal overfitting, although the minor divergence in later epochs indicates a potential risk of overfitting if training continues.

Phase	Training Accuracy	Validation Accuracy	Interpretation
Start (First Few Epochs)	Low & Increasing	Low & Increasing	Model is learning patterns effectively.
Midway (Epochs 10–30)	Keeps rising	Matches training accuracy	Ideal learning phase.
End (Epochs 30–50)	95.00%	87.12%	Good generalization, minimal overfitting.

Table .4. Analysis of Accuracy Graph of CNN

Table 4 Illustrates the CNN performs significantly better than MLP with higher validation accuracy. Validation accuracy remains close to training accuracy, indicating good generalization

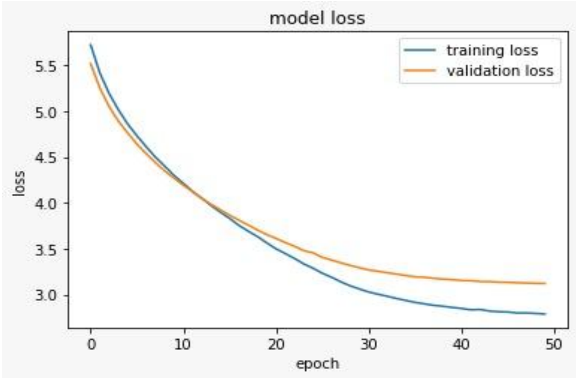


Fig .9. Model loss of CNN

The Fig 9 illustrates the training and validation loss trends of a CNN over 50 epochs, with the **x-axis** representing epochs (complete passes through the dataset) and the **y-axis** showing the loss value, which indicates prediction errors. The **training loss** (blue line) decreases consistently, reflecting that the model is improving its accuracy on the training data. The **validation loss** (orange line) also drops initially and then stabilizes, suggesting that the model is learning effectively and maintaining generalization to unseen data. The minimal gap between the two lines indicates reduced overfitting and a good balance between training and validation performance.

Table 5 Analysis of Loss Graph of CNN

Phase	Training Loss	Validation Loss	Interpretation
Start (First Few Epochs)	High & Decreasing	High & Decreasing	Model is learning effectively.
Midway (Epochs 10–30)	Decreases consistently	Close to training loss	Good generalization.
End (Epochs 30–50)	Continues decreasing	Slight fluctuations	Overfitting is minimal.

Table 5 The loss values remain stable and decrease simultaneously, indicating effective model learning. Toward the end, the validation loss shows slight fluctuations, suggesting mild overfitting, though it is not significant. Key Observations from Both Tables that **a**ccuracy increases steadily during the initial and midway epochs, showing effective learning. Training accuracy reaches 95% by the end, while validation accuracy stabilizes at 87.12%, indicating good generalization. Loss decreases consistently across all phases. Training loss drops significantly, and validation loss stays close to it, reflecting minimal over-fitting. Slight fluctuations in validation loss near the end suggest stable but not perfect generalization. The model demonstrates Overall Performance as strong learning in early epochs, maintains balance between training and validation metrics, and shows minimal overfitting with sustained accuracy and decreasing loss throughout.

Table 6 Comparison of MLP vs CNN Model

Metric	MLP Model	CNN Model	Best Model
Training Accuracy	87.71%	95.00%	CNN ✅
Validation Accuracy	74.29%	87.12%	CNN ✅
Training Loss	Keeps decreasing	Keeps decreasing	CNN ✅
Validation Loss	Plateaus	Stays close to training loss	CNN ✅

The table 6 provides a comparative analysis of the performance metrics for the MLP (Multi-Layer Perceptron) and CNN (Convolutional Neural Network) models used for vehicle image classification. It highlights four key metrics: training accuracy, validation accuracy, training loss, and validation loss. The CNN model outperforms the MLP model across all metrics. Specifically, the CNN model achieves a higher training accuracy of **95.00%** compared to **87.71%** for the MLP model, and it also exhibits superior validation accuracy at **87.12%**, outperforming the MLP's **74.29%**. This indicates that the CNN generalizes better to unseen data. Both models show a consistent decrease in training loss, suggesting effective learning; however, while the MLP model's validation loss plateaus, indicating potential overfitting, the CNN's validation loss remains close to its training loss, demonstrating better generalization. Based on these performance metrics, CNN is marked as the best model due to its superior accuracy and lower risk of overfitting.

V. CONCLUSION

In this paper, we established a data-driven approach to vehicle image classification using deep learning models, specifically Multi-Layer Perceptron (MLP) and Convolutional Neural Networks (CNN). In recent years, vehicle image classification has become increasingly relevant due to its role in intelligent transportation systems, automated traffic monitoring, and urban surveillance. Within this data-driven framework, deep learning emerges as a powerful tool for accurately classifying vehicles into predefined categories such as cars, trucks, buses, and motorcycles, ultimately contributing to efficient traffic management and surveillance. By leveraging deep learning architectures, we aimed to identify key factors that influence image classification accuracy and develop models capable of handling complex visual data. The research findings demonstrated the superior performance of CNNs compared to MLPs, with CNNs significantly outperforming other approaches due to their ability to automatically learn spatial features and hierarchical patterns from image data. This makes CNNs the preferred choice for vehicle image classification, particularly in complex multiclass tasks where accurate spatial representation is essential. A critical finding of this research is CNNs' ability to generalize effectively, capturing detailed spatial patterns that are critical for distinguishing between visually similar vehicle categories. The comparative analysis confirmed that while MLPs can handle simpler tasks, they lack the spatial feature extraction capabilities necessary for image-based classification, leading to lower accuracy. The analysis further showed that incorporating advanced techniques like data augmentation, hyperparameter tuning, and regularization methods significantly improved model performance. Techniques such as image rotation, flipping, and scaling enhanced the diversity of the training dataset, enabling the models to generalize better to unseen data. Additionally, regularization methods like dropout and L2 regularization reduced overfitting, while hyperparameter tuning optimized key parameters for faster and more efficient convergence. This research also addressed challenges like potential overfitting and the trade-off between training and validation accuracy by fine-tuning the models to strike an optimal balance. The paper emphasizes the growing importance of deep learning in computer vision and intelligent transportation, illustrating how image classification models can enhance vehicle detection, monitoring, and traffic flow analysis. The ability to classify vehicles accurately provides opportunities for improving autonomous vehicle systems, automated toll collection, and urban surveillance, which can lead to better resource allocation, optimized traffic management, and safer transportation networks. Moreover, this research highlights the need for scalable, real-world deployment of deep learning models to enhance real-time vehicle classification and improve the overall efficiency of transportation systems. In summary, this research demonstrates the potential of deep learning in vehicle image classification, offering a robust tool for improving intelligent traffic monitoring, surveillance, and vehicle detection. By combining predictive modeling with adaptive image processing techniques, transportation systems can become more efficient, responsive, and data-driven. As deep learning technologies continue to advance, these models are expected to evolve further, enhancing their accuracy and contributing to the ongoing transformation of intelligent transportation and automated vehicle monitoring systems.

REFERENCES

[1]. Gamer, M., et al. (1990). Infrared Detectors for Counting, Classifying, and Weighing Vehicles. Journal of Applied Optics, 29(11), 1799-1807.

[2]. Huang, F., et al. (2012). Vehicle Classification using Clustering and Neural Networks. IEEE International Conference on Granular Computing, Hangzhou, China, 2012.

[3]. Meshram, S. A., & Malviya, A. V. (2013). Traffic Surveillance by Counting and Classification of Vehicles from Video using Image Processing. International Journal of Computer Applications, 75(8), 27-34.

[4]. Singh, A. K., & Bajiya, N. C. (2015). Image Vehicle Classification Based on Adaptive Edge Detection & PCA Method. *International Journal of Computer Applications*, 113(11), 1-7.

[5]. Tripathi, J., et al. (2015). Automatic Vehicle Counting and Classification. *International Conference on Intelligent Systems and Computing, 1(4), 312-318.

[6]. Shankar, M., et al. (2015). Intelligent Vehicle Recognition System using Reinforcement Learning. Second International Conference on Advances in Computing and Communication Engineering, Dehradun, India, 2015.

[7]. Patel, P., & Tiwari, S. (2016). Flame Detection using Image Processing Techniques. International Journal of Computer Applications, 143(6), 31-35.

[8]. Chen, M., et al. (2017). CNN-based Vehicle Classification with Big Data Analytics. IEEE Transactions on Intelligent Transportation Systems, 18(8), 2223-2231.

[9]. Gavhane, A., et al. (2018). Deep Neural Networks for Automated Vehicle Recognition. Second International Conference on Electronics, Communication, and Aerospace Technology (ICECA), Coimbatore, India, 2018.

[10]. Dahiwade, D., et al. (2019). CNN-based Vehicle Classification Model. 3rd International Conference on Computing Methodologies and Communication (ICCMC), 1-6.

[11]. Yu, H. Q. (2019). Hybrid Deep Learning Model with Feature Extraction for Vehicle Classification. IEEE Transactions on Vehicular Technology, 68(10), 9701-9709.

[12]. Shetty, S. V., et al. (2019). Vehicle Classification Using ResNet-50 Deep Learning Model. International Journal of Engineering and Technology, 8(6), 2473-2478.

[13]. Grampurohit, S., et al. (2020). Role of Vehicle Classification in Autonomous Driving and Intelligent Surveillance Systems. International Conference for Emerging Technology (INCET), Belgaum, India, 2020.