Object Detection and Classification for Autonomous Vehicles Using Deep Learning Techniques

Kiran Neupane

Department of Electrical Engineering and Computer Science
University of Wyoming
Date: May 10, 2024

1 Abstract

The existence of emergency vehicles, installed traffic signs, and automated driving systems requires a sophisticated, precise, and powerful object detection system to keep these vehicles aware of the objects around them. Vehicles adjust their speed and behavior accordingly to avoid colliding with objects on the road and to adhere to traffic laws regarding the presence of emergency vehicles and posted traffic signs. Obtaining high-quality data and ensuring the safety and reliability of object detection systems in real environments are the main challenges associated with this project. To solve these problems, we will use data augmentation techniques, rigorous testing, validation, and verification procedures, including simulation-based testing and continuous system performance monitoring. The project's main objective is to create an intelligent object classification system for autonomous vehicles to forecast the presence of objects under investigation over the road network by utilizing the machine learning technique, Convolution Neural Network (CNN).

2 Introduction

Autonomous systems function with high autonomy and self-determination under challenging, ambiguous conditions. They possess the ability to perceive, learn, and act in a self-aware manner. These systems can also respond skillfully to sudden changes in their surroundings. These kinds of systems have the power to change society fundamentally. Autonomous vehicles (AVs) are one of the many varieties of autonomous systems technology. At level four, driving is completely automated, and the driver does not need to take the wheel. Fully autonomous level 5 is available [1]. Automated driving systems need a sophisticated, precise, and powerful object detection system to keep these cars aware of the objects around them. As a result, autonomous cars modify their speed and operations to avoid crashing into existing objects and follow the driving rules around the existence of emergency vehicles and installed traffic signs. There are four main sub-systems of autonomous vehicles: hardware, software, communication, and human-machine interaction. It also includes cameras, lidar, and GPS. These components facilitate obtaining features that are necessary for object detection. Moving targets can be identified, located, and tracked using lidar and radar technologies, which can produce a map of the surrounding area for the vehicle [2]. In automotive driving applications, objects of interest include automobiles, bicycles, motorbikes, pedestrians, and other obstacles on or near roads. Radars extract characteristics related to depth perception using highly dependable processing techniques, thereby providing direct perception-related inputs. RGB cameras record red, green, and blue light wavelengths to produce images that closely resemble human vision. Processing these photos makes it easier to recognize objects. Additionally, the GPS helps these vehicles navigate by identifying each vehicle's longitude, latitude, speed, and direction. Regular vehicles, big trucks, emergency vehicles, pedestrians, bicycles, traffic lights, and traffic signs are considered in this project.

3 Literature Review

Many researchers have focused on object detection and classification issues related to the road network (such as passenger cars, emergency vehicles, bicycles, trucks, and pedestrians). These studies aim to successfully lower the rising number of traffic accidents on road networks. The author of [1] introduced a deep convolutional neural network designed to extract image representations automatically. In [3], the authors implemented 3D object detection and classification, focusing on vehicles as the identified objects. In [4], the authors explored the impact of fog on object classification in driving scenarios, aiming to enhance performance in foggy weather. The author of [5] also demonstrated how weather conditions such as rain, fog, and snow influence an autonomous vehicle's ability to detect objects.

In this project, we have used Convolutional Neural Networks (CNN) to classify the environmental object of Autonomous vehicles. In addition to the existing literature, our project has introduced a novel contribution by focusing on the roadside environment. We have utilized four key features—object size, speed, motion status, and color—to enhance the object detection process for autonomous vehicles. This approach extends the scope of current research and provides a comprehensive understanding of the factors influencing accurate classification in real-world road scenarios. For this project, we have reduced the number of categories by dividing the dataset into three categories based on their mobility status: vehicles, pedestrians, and road signs. Furthermore, considering the speed of particles in the road will aid in classification and position prediction, hence developing a reliable model. Figure 1 illustrates the main objective of this work. However, there are many challenges that we have to tackle in this study such as obtaining high-quality data, ensuring the safety and reliability of object detection systems in real environments, and computational burden (not having access to GPU, TPU). To solve these problems, we will be using data augmentation techniques, rigorous testing, validation, and verification procedures, including simulation-based testing and continuous monitoring of system performance.

Figure 1: Autonomous vehicle pattern detection.

4 Methodology

The Resnet-50 architecture of convolutional neural network (CNN) is used in this project. A convolutional neural network is an artificial neural network primarily used to analyze visual imagery. It is used in image recognition and classification tasks, as they are particularly good at capturing spatial hierarchies in images through convolutions. These convolutions allow networks to efficiently learn patterns such as edges, textures, and shapes at different levels of abstraction. The diagram illustrates the process flow of a machine learning classification project. It begins with the 'Dataset,' the collection of raw data that will serve as the foundation for the model. From there, the flowchart leads into 'Data Processing,' where the data is prepared for analysis. This includes 'Handling missing value,' an essential step to ensure the integrity of the dataset by addressing any incomplete or missing entries. After processing, the data is divided into the'Train set' and 'Test set.' The 'Train set' trains the machine learning algorithm, teaching it to recognize patterns and make predictions. The 'Test set,' however, is reserved for testing the model's accuracy against data it has not seen before. The flow then converges on 'Classification,' where the trained algorithm categorizes the data into predefined classes. The output of this phase is a 'Trained model,' a refined version of the algorithm learned from the training process. Finally, the 'Trained model' undergoes 'Evaluation,' which assesses performance. This stage is crucial for verifying the model's effectiveness and ensuring it meets the project's objectives. The model can be deployed if the evaluation is successful; otherwise, it may require further refinement. Figure 2 shows the mechanism used to illustrate the methodology.

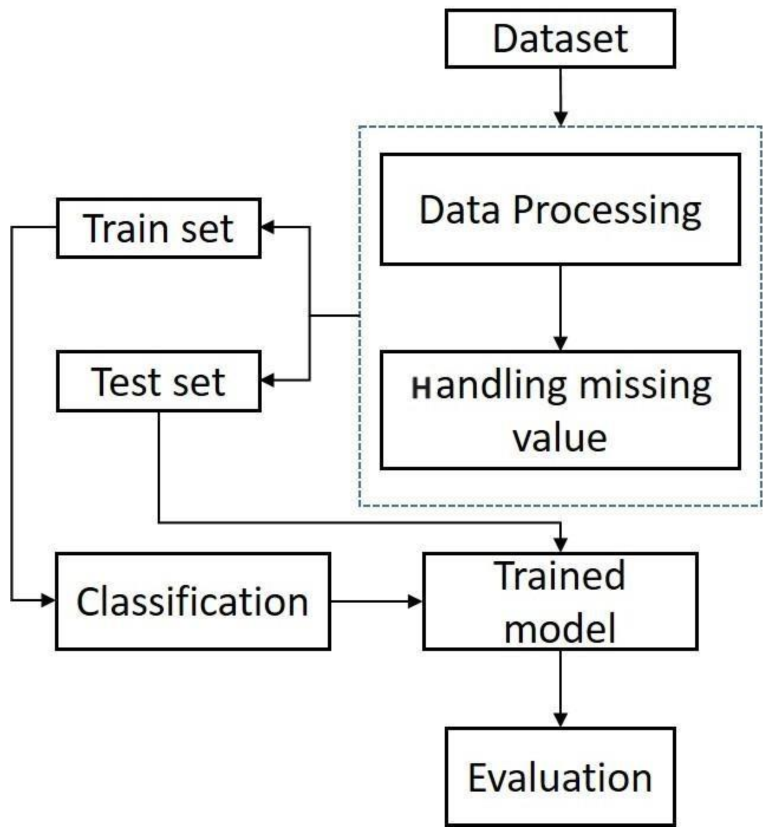


Figure 2: Flowchart of Machine Learning Classification.

4.1 Data Description

The Udacity datasets collected from the Roboflow repository are used in this study. The diversity of the data is essential to test the robustness of perceptron algorithms. This dataset includes many scene types, including city streets, residential areas, and highways. In addition, the videos were taken in various climatic conditions and at various times throughout the day and night. It includes 29,800 images, which are categorized into 11 classes. These classes represent a diverse set of road elements and scenarios that an autonomous vehicle might encounter, which are vital for developing robust detection systems. The classes used in this study are:

- Biker
- Car
- Pedestrian
- Traffic signs
- Traffic light green
- Traffic light green left

- Traffic light red
- Traffic light red left
- Traffic light yellow
- Traffic light yellow left
- Truck

4.2 Result and Discussion

The ResNet-50 architecture of convolutional neural network (CNN) is used to classify objects in the images. This Resnet-50 is a pre-trained model. It's normally called the transfer learning model. The model is pre-trained with around 1 million data, and the updated weights are used as the initial weights for this specific dataset. The dataset contains 29800 images of 224*224 pixels. Due to the computational constraints, only 50 epochs are run to analyze the improvement in the loss function. During this project, cross-entropy loss is used as a loss function. Cross entropy loss is assumed to be suitable for multiclass classification. It measures the performance of the classification model. Adam optimizer has been used in this study and is known for its adaptive learning capabilities, which makes it efficient in handling sparse gradients on noisy problems. The batch size, which is the number of samples that will be propagated through the network in one pass, offering a good balance between the speed of computation and the stability of the learning process, is set to 32. Figure 3 below shows the performance of training data and the loss for the number of epochs. It can be observed from the plot that the loss decreases as the number of epochs increases. They can be sufficiently minimized if parameters like the number of epochs, CNN architecture, and other hyperparameters are optimally selected.

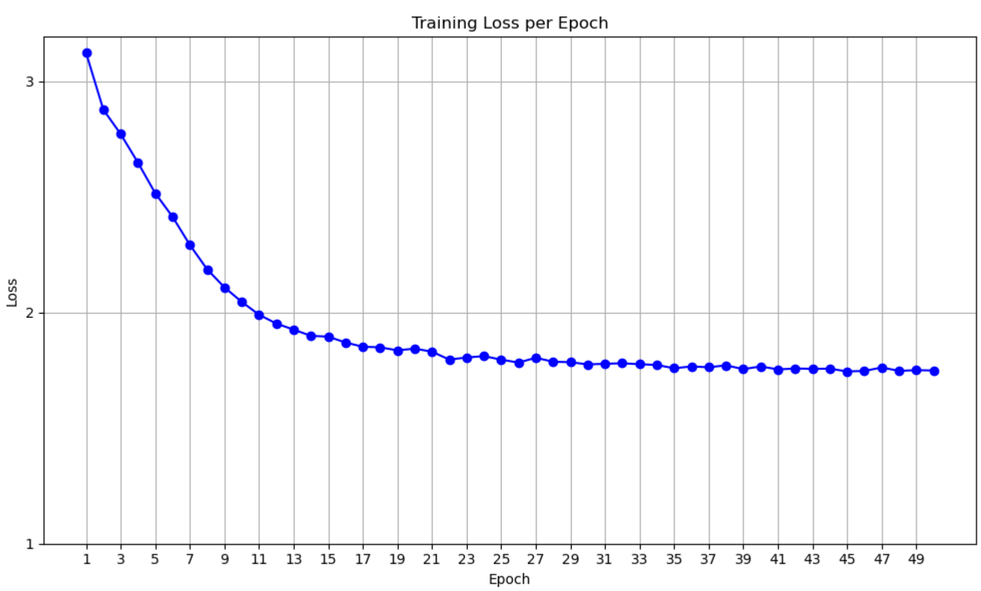


Figure 3: Training loss behavior.

Table 1 shows the classification metrics used to analyze the classifier's performance. The confusion matrices for the training and testing datasets are obtained, as shown in the figures below. It gives the total numbers of correctly and falsely classified datasets in different classes. As eleven classes are in our dataset, the confusion matrix is obtained as 11×11. Based on the truly predicted or falsely predicted, the confusion matrix elements are called true positive, true negative, false positive, or false negative.

Table 1: Classification metrics for the training and testing datasets

Metrics	**Training dataset**	**Test dataset**
Accuracy	0.7192	0.7178
Precision	0.2975	0.2704
Recall	0.4897	0.4424
F1 Score	0.2572	0.2258

Figure 4 compares the classification performance metrics across training and test datasets. It features four distinct categories of metrics: Accuracy, Precision, Recall, and F1 Score. The performance metrics for a classifier evaluated on training and test datasets are summarized below:

Accuracy

The classifier achieves a high level of accuracy on both datasets. For the training dataset, the accuracy is close to 71.92, and it is similarly high for the test dataset at around 71.78.

Precision

Precision is somewhat lower than accuracy across both datasets. The training dataset shows a precision just below 0.30, while the test dataset records a precision of around 0.27.

Recall

Recall measures the ability to detect relevant instances and is moderately high. For the training dataset, recall is near 0.49, decreasing to approximately 0.44 in the test dataset.

F1 Score

The F1 Score, which balances precision and recall, is the lowest among the metrics evaluated. The training dataset scores about 0.26, and the test dataset is slightly lower at about 0.23.

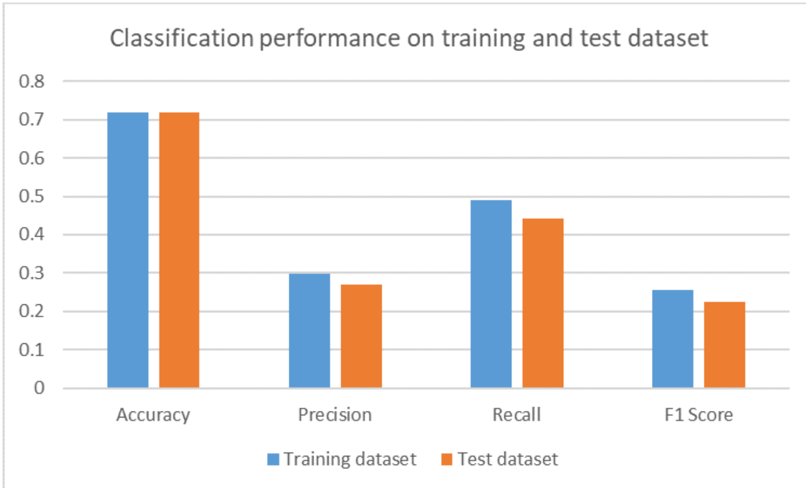
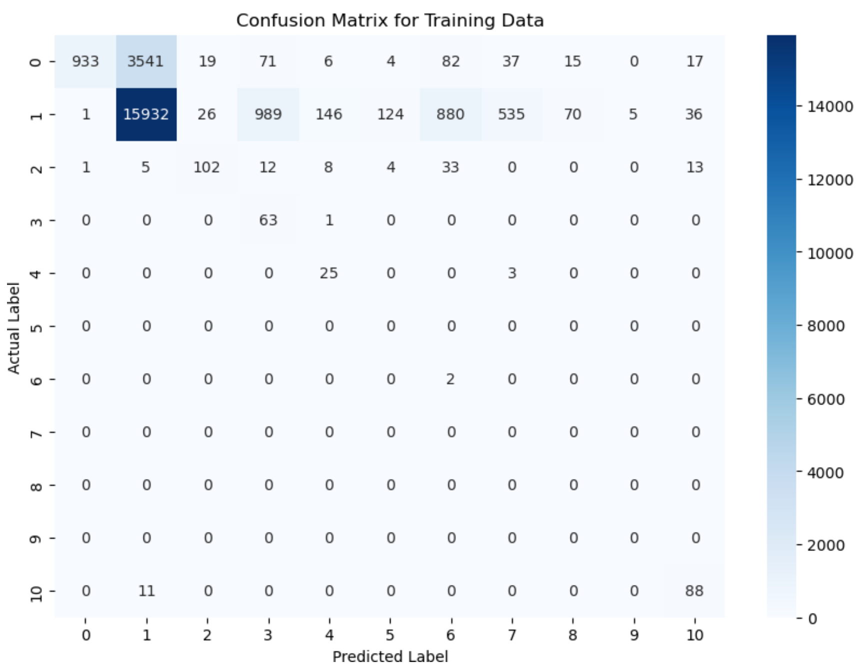


Figure 4: Comparative Analysis of Classification Metrics Across Training and Test Datasets.

Figures 5 and 6 show the confusion matrix for both the training and testing datasets that illustrate the performance measurement for our model.



	0	1	2	3	4	5	6	7	8	9	10
0	933	3541	19	71	6	4	82	37	15	0	17
1	1	15932	26	989	146	124	880	535	70	5	36
2	1	5	102	12	8	4	33	0	0	0	13
3	0	0	0	63	1	0	0	0	0	0	0
4	0	0	0	0	25	0	0	3	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	2	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0
10	0	11	0	0	0	0	0	0	0	0	88

Figure 5: Training data confusion matrix.

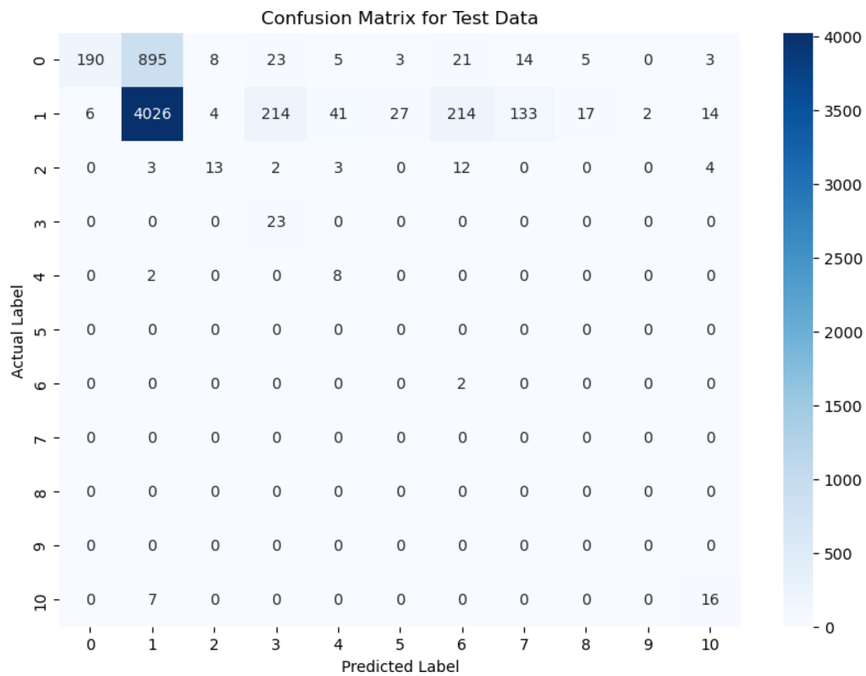


Figure 6: Test data confusion matrix.

5 Conclusions

The primary aim of our project is to develop a robust system capable of predicting the speed of surrounding objects and accurately classifying them in real-time. Furthermore, this Project demonstrates the application of a convolution neural network to classify objects that autonomous vehicles may encounter on roadways. The project explores the effectiveness of Convolutional Neural Networks (CNN) in solving classification problems, with a focus on their application in autonomous vehicle environments. Hyperparameter tuning, Regularization techniques, and Transfer learning with larger models could be used in future work to enhance our model performance. This work will be improved more before submitting it to Jordan conference (IEEE conference) in August.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Advances in neural information processing systems*, vol. 25, 2012.

[2] Q.-C. Mao, H.-M. Sun, L.-Q. Zuo, and R.-S. Jia, “Finding every car: a traffic surveillance multi-scale vehicle object detection method,” *Applied Intelligence*, vol. 50, pp. 3125–3136, 2020.

[3] A. Agafonov and A. Yumaganov, “3d objects detection in an autonomous car driving problem,” in *2020 International Conference on Information Technology and Nanotechnology (ITNT)*, pp. 1–5, IEEE, 2020.

[4] N. A. M. Mai, P. Duthon, L. Khoudour, A. Crouzil, and S. A. Velastin, “3d object detection with sls-fusion network in foggy weather conditions,” *Sensors*, vol. 21, no. 20, p. 6711, 2021.

[5] M. J. Mirza, C. Buerkle, J. Jarquin, M. Opitz, F. Oboril, K.-U. Scholl, and H. Bischof, “Robustness of object detectors in degrading weather conditions,” in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pp. 2719–2724, IEEE, 2021.