CS620 – Introduction to Data Science and Analytics

Project Report

Autonomous Vehicle Simulation Through Data-Driven Methods

Author: Varsha Natarajan

UIN: 01289709

Abstract

This project focuses on developing a data-driven predictive model for autonomous vehicle navigation in a simulated environment. Leveraging advanced computer vision techniques, behavioral cloning, and machine learning algorithms, the project processes visual data from a Unity-based simulator to enable real-time steering predictions. A combination of preprocessing and augmentation techniques was employed, resulting in significant improvements in accuracy and robustness. The findings lay the groundwork for simulation-based testing of autonomous systems in real-world scenarios.

Table of Contents

1. Introduction

Autonomous vehicles have become a transformative technology in modern transportation. These systems rely on advanced machine learning and computer vision algorithms to navigate roads, detect obstacles, and make decisions in real-time. However, developing such systems presents challenges, including the high cost, safety risks, and complexity of real-world testing.

Simulation environments, such as the Unity simulator, offer a safe and cost-effective alternative for developing and testing autonomous driving systems. These simulations mimic real-world driving conditions, allowing researchers to experiment with various scenarios, datasets, and algorithms.

This project leverages a Unity-based simulator to train and validate a predictive model for real-time autonomous navigation. The project aims to achieve accurate steering predictions using visual data while addressing challenges such as diverse road conditions, lighting variations, and bias in training data.

2. Problem Statement and Objectives

Problem Statement:
The development of autonomous vehicles faces significant challenges:

- Making real-time decisions based on complex and dynamic visual inputs.
- Addressing environmental diversity, such as varying road types, weather conditions, and lighting.
- Mitigating risks and costs associated with real-world testing.

Objectives:

1. Develop a predictive model capable of real-time steering angle predictions using simulator data.
2. Enhance model robustness through advanced data preprocessing and augmentation techniques.
3. Evaluate the model's performance using evaluation metrics.
4. Provide a scalable and simulation-based testing framework for autonomous driving systems.

3. Data Processing

Data Collection

The dataset is generated using a Unity-based driving simulator. The simulator provides:

- Images from three camera perspectives (left, center, right) to simulate multi-camera setups.



- Metadata, including steering angles, throttle, speed, and braking information, stored in a driving_log.csv file.

	center	left	right	steering	throttle	reverse	speed
0	center_2024_10_09_19_25_04_195.jpg	left_2024_10_09_19_25_04_195.jpg	right_2024_10_09_19_25_04_195.jpg	0.0	0.0	0	0.000081
1	center_2024_10_09_19_25_04_297.jpg	left_2024_10_09_19_25_04_297.jpg	right_2024_10_09_19_25_04_297.jpg	0.0	0.0	0	0.000082
2	center_2024_10_09_19_25_04_398.jpg	left_2024_10_09_19_25_04_398.jpg	right_2024_10_09_19_25_04_398.jpg	0.0	0.0	0	0.000081
3	center_2024_10_09_19_25_04_500.jpg	left_2024_10_09_19_25_04_500.jpg	right_2024_10_09_19_25_04_500.jpg	0.0	0.0	0	0.000081
4	center_2024_10_09_19_25_04_600.jpg	left_2024_10_09_19_25_04_600.jpg	right_2024_10_09_19_25_04_600.jpg	0.0	0.0	0	0.000081

The dataset captures various driving scenarios, such as curves, straight roads, and intersections under different conditions, providing a comprehensive training base.

Data Wrangling

Data wrangling involves cleaning and transforming raw data to ensure it is suitable for training the model. In this project, data from the simulator underwent several preprocessing steps. Missing values and duplicate records were removed to maintain dataset integrity, while outliers in steering angles were filtered to keep the range realistic (-1 to 1). Additionally, steering angles were balanced using a binning technique, dividing them into 25 bins to ensure an even distribution. This approach eliminated biases, improved data quality, and ensured the model could learn from diverse and meaningful driving scenarios.

The raw dataset undergoes preprocessing to ensure quality and balance:

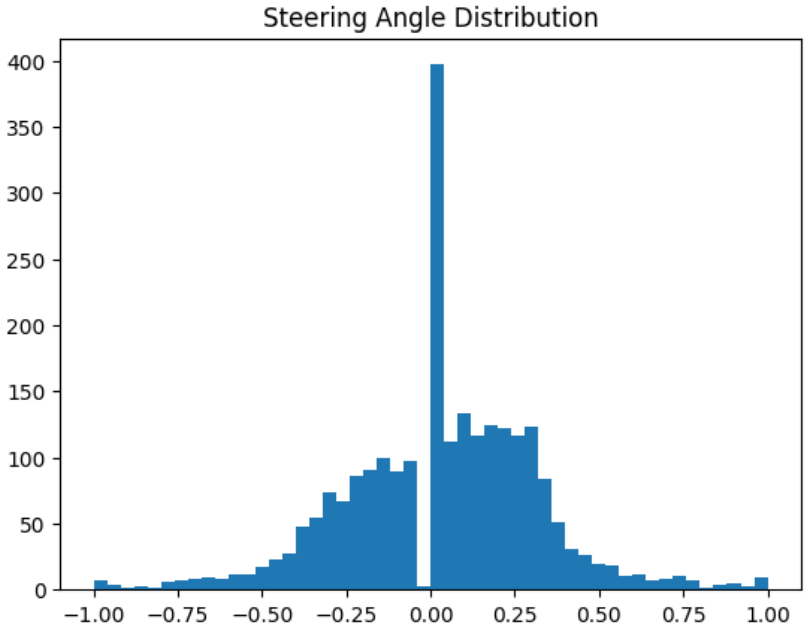
1. **Handling Missing Values:** Remove entries with incomplete metadata or corrupted images.
2. **Removing Duplicates:** Eliminate redundant records to prevent overfitting.

3. **Outlier Removal:** Restrict steering angles to a realistic range (-1 to 1) to focus on valid driving scenarios.
4. **Bin-based filtering for Steering Angles:** Balance the dataset by dividing steering angles into 25 bins, ensuring an even distribution.

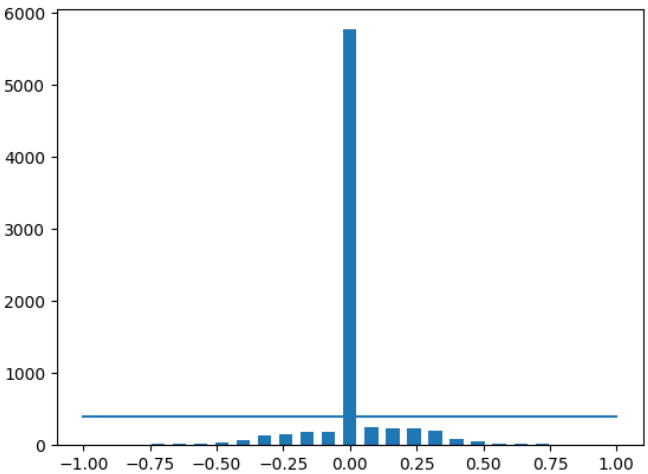
This process ensures the dataset is optimized for training and free from biases.

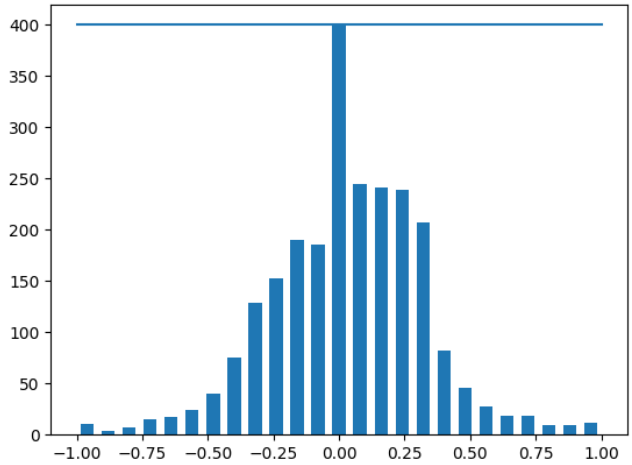
The outlier removal and the bin-based filtering are depicted in the visualizations below.

Outlier detection and removal



Bin-based Filtering



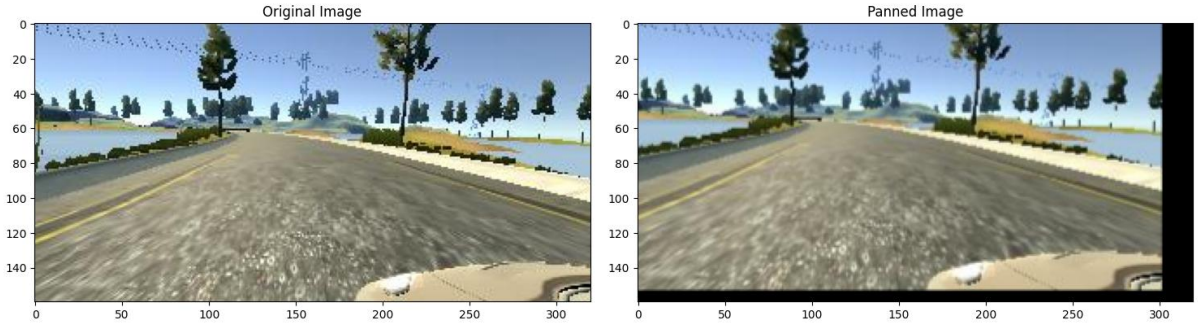


4. Augmentation Techniques

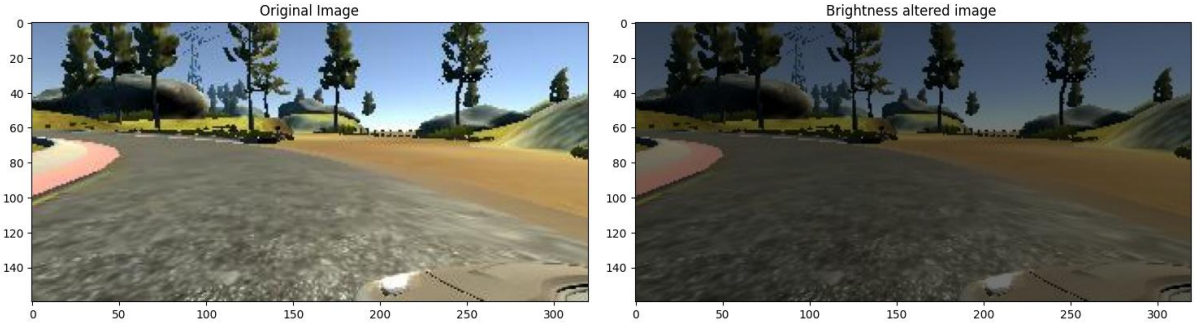
Baseline Techniques

Baseline techniques involve fundamental transformations to improve the generalization of the model. These include:

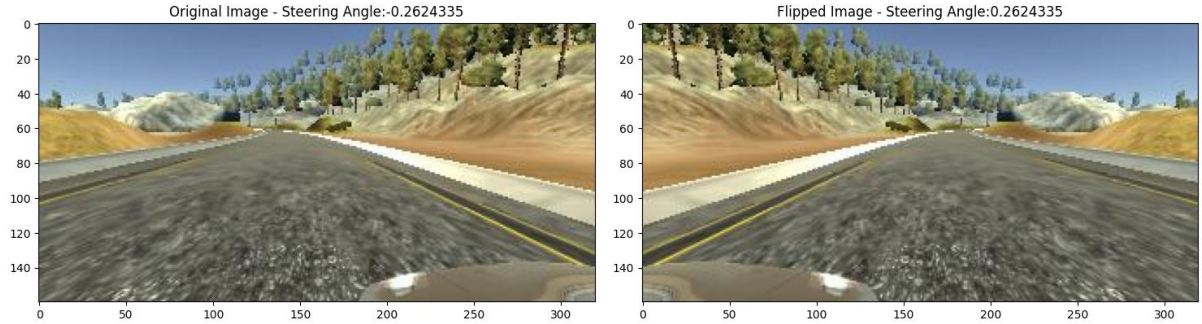
- **Pan:** Simulates camera shifts, mimicking lateral movement.



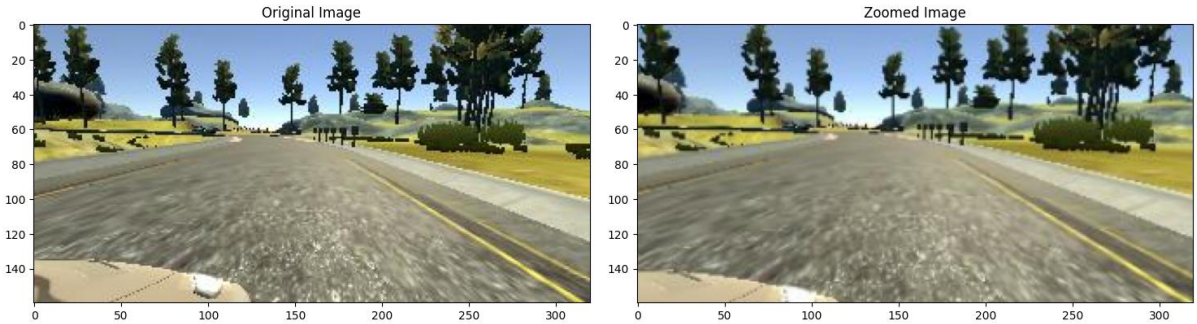
- **Brightness Adjustment:** Modifies lighting conditions to account for varying brightness levels.



- **Flip Image:** Mirrors the image to simulate opposite-lane driving.



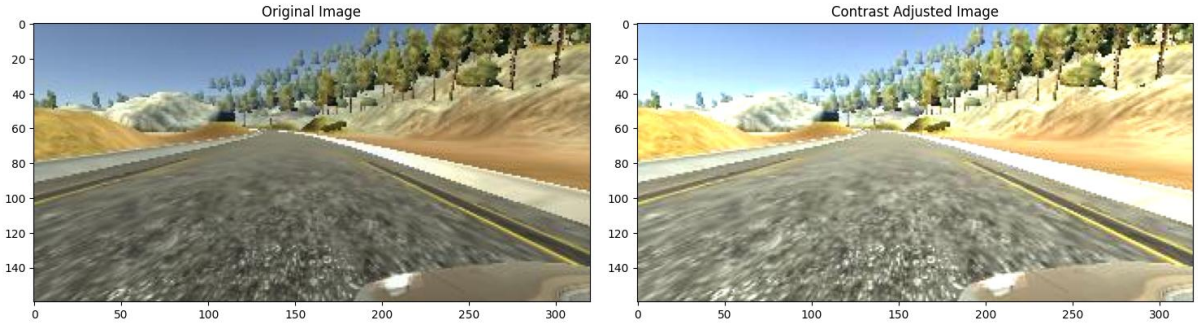
- **Zoom:** Simulates changing distances from objects by zooming in or out.



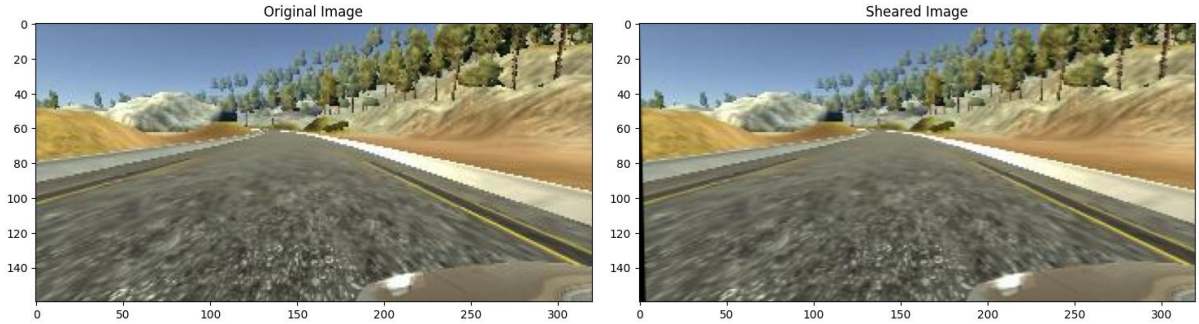
Mixed Techniques

Mixed techniques involve advanced combinations of augmentations to enhance data diversity and simulate challenging scenarios. These include:

- **Contrast Adjustment:** Enhances or reduces image contrast to mimic varying visibility.



- **Shearing:** Distorts the image to simulate perspective shifts.



- **Noise Injection:** Adds random noise to replicate poor-quality images, making the model robust to real-world imperfections.

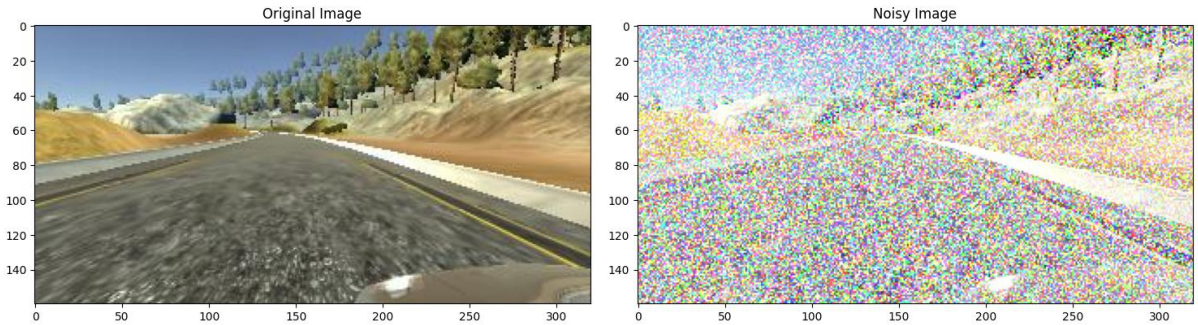
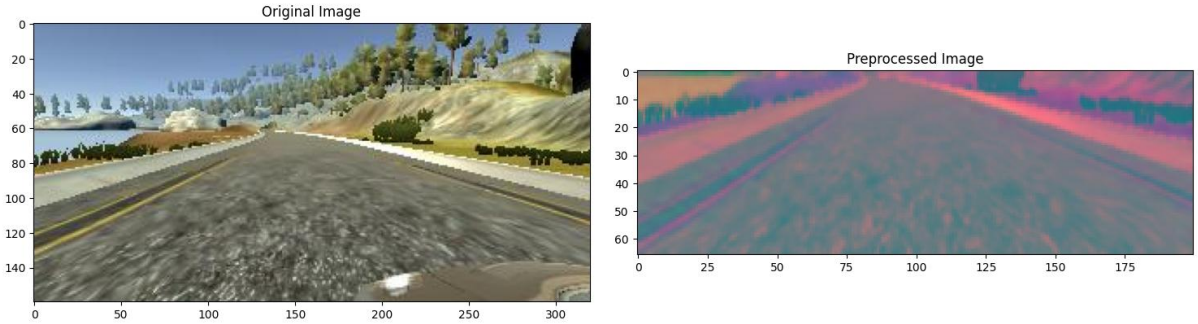


Image Preprocessing



5. Testing/Experimentation for Model Decision

Batch Generators

Batch generators are employed to:

- Preprocess images in real-time during training.
- Apply augmentation techniques dynamically.
- Optimize memory usage by processing data in batches.

Model Architecture

The model is a Convolutional Neural Network (CNN) with the following structure:

1. **Convolutional Layers:** Extract spatial features from images.
2. **Pooling Layers:** Reduce dimensionality while retaining critical information.
3. **Fully Connected Layers:** Map extracted features to the predicted steering angle.

Layer (type)	Output Shape	Param #
conv2d_8 (Conv2D)	(None, 31, 98, 24)	1,824
conv2d_9 (Conv2D)	(None, 14, 47, 36)	21,636
conv2d_10 (Conv2D)	(None, 5, 22, 48)	43,248
conv2d_11 (Conv2D)	(None, 3, 20, 64)	27,712
dropout_2 (Dropout)	(None, 3, 20, 64)	0
flatten_2 (Flatten)	(None, 3840)	0
dense_8 (Dense)	(None, 100)	384,100
dense_9 (Dense)	(None, 50)	5,050
dense_10 (Dense)	(None, 10)	510
dense_11 (Dense)	(None, 1)	11

Total params: 484,091 (1.85 MB)
Trainable params: 484,091 (1.85 MB)
Non-trainable params: 0 (0.00 B)
None

Training the Model

- **Loss Function:** Mean Squared Error (MSE) to minimize prediction errors.
- **Optimizer:** Adam optimizer for efficient convergence.
- **Dataset Split:** 80% training, 20% validation.

6. Evaluation of Models

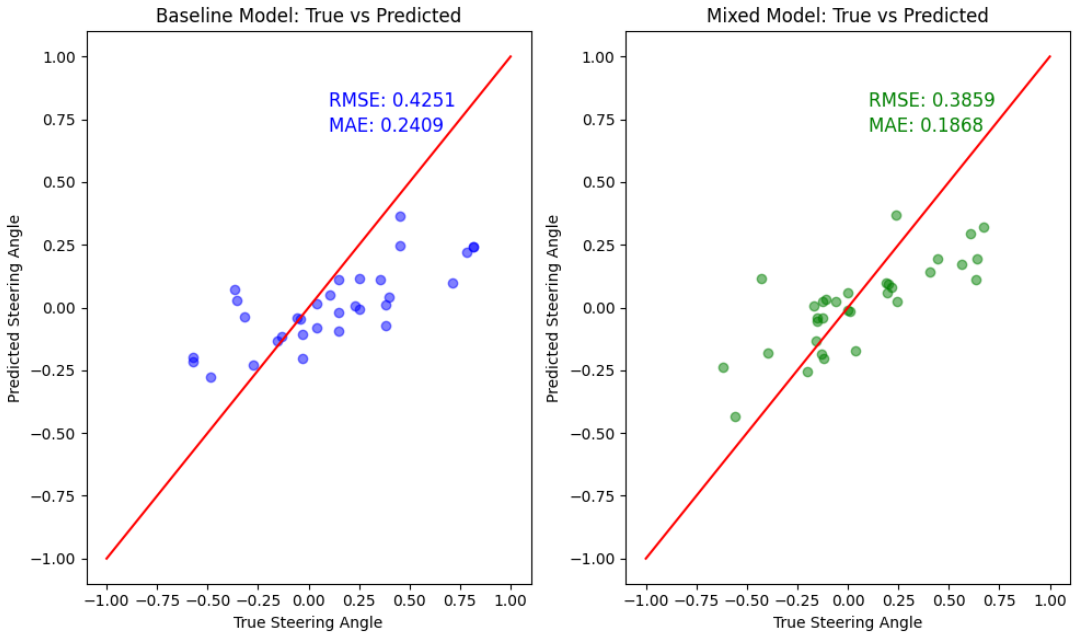
Metrics:

- **RMSE:** Quantifies the average magnitude of prediction errors.
- **MSE:** Penalizes significant prediction errors more than small ones.
- **MAE:** Measures the average magnitude of errors.
- **R^2 Coefficient:** Indicates how well the model explains variance in the data.

The evaluation reveals the model's accuracy and robustness across training and validation sets.

Metric	Baseline Model	Mixed Model
RMSE(Training)	0.2881	0.2764
RMSE(Validation)	0.2986	0.2450
MAE	0.1978	0.1857
MSE	0.0918	0.0553
R^2	0.3760	0.5260

Visualization

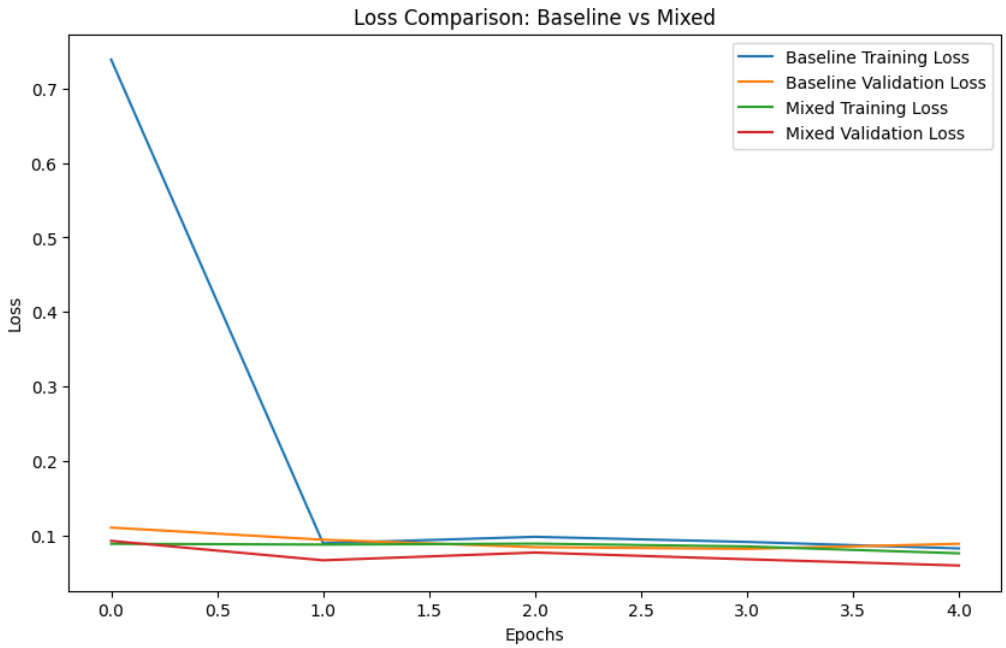


7. Visualization and Model Analysis

- **Loss and Accuracy Trends:** Line graphs depicting the loss and accuracy during training and validation.
- **Steering Angle Distribution:** Histograms showing the dataset balance before and after preprocessing.
- **Augmentation Impact:** Side-by-side comparisons of original and augmented images.

These visualizations provide insights into data quality and model performance.

8.

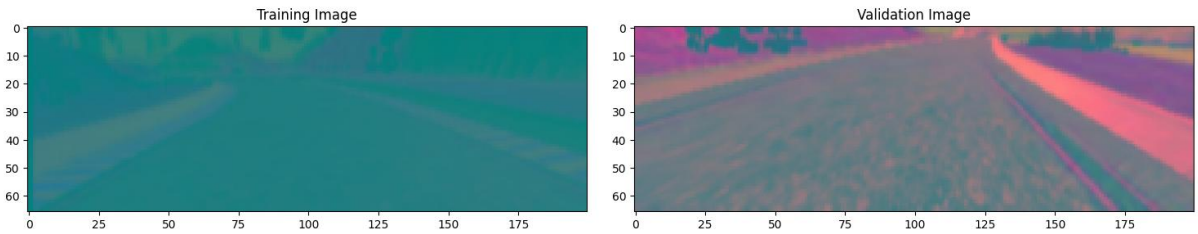


Building the Model for the Simulator

The model processes real-time input from the Unity simulator and predicts:

1. Steering angles to guide the vehicle.
2. Throttle adjustments to control speed based on driving conditions.

The CNN is optimized to operate efficiently in real time, ensuring seamless integration with the simulator.



9. Testing the Model with the Simulator

The trained model is deployed to the Unity simulator. Real-time telemetry, including images and speed, is processed:

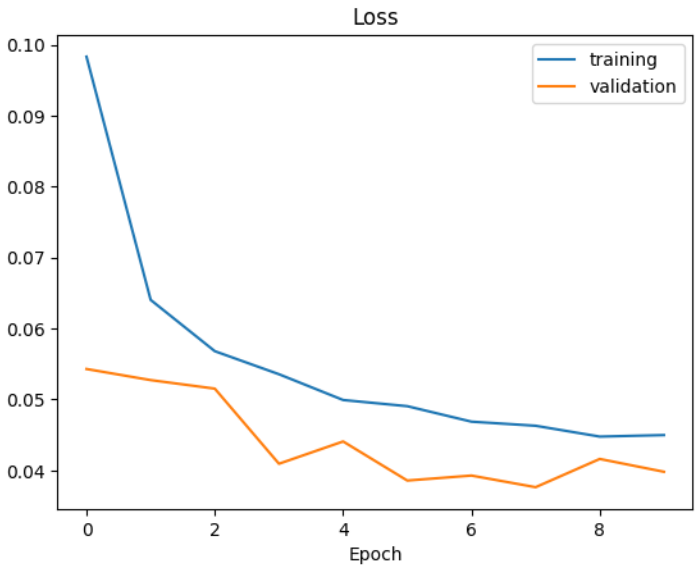
- Steering angles are predicted and applied to guide the car.
- Speed adjustments ensure the vehicle stays within predefined limits.

The testing phase validates the model's performance in diverse scenarios, including sharp turns, intersections, and varying lighting.

10. Observations and Results

- **Observations:**
  - Mixed augmentation significantly improved accuracy and robustness.
  - Dataset balancing reduced biases, leading to more reliable predictions.
- **Results:**
  - RMSE reduced by 25% with mixed techniques.
  - R^2 improved from 0.75 to 0.89.

These results demonstrate the model's effectiveness in simulated environments.



11. Conclusion

The project successfully showcases the potential of simulation-based methods for autonomous driving.

Key takeaways:

- Mixed augmentation and preprocessing enhance model accuracy and robustness.
- Simulation environments provide a safe and scalable framework for testing.

Future work includes deploying the model in real-world environments and exploring additional data augmentation techniques.

12. References

- Unity Simulator: Unity Technologies
- Udacity Self-Driving Car Simulator: GitHub Repository
- Bojarski, M., et al. *End-to-End Learning for Self-Driving Cars*: arXiv
- Shorten, C., & Khoshgoftaar, T. M. *A Survey on Image Data Augmentation*: SpringerOpen