

# **Self-Driving Car – Final Project Report**

Course: CVI – Computer Vision

Project: End-to-End Steering for Simulator Car

Group Members:

- Pouya Rad
- Param Jayesh Katrodia
- Yashkumar Bharatkumar Patel

## 1. Project Goal

The goal of this project was to train a convolutional neural network that takes an image from the front camera of the simulator car and predicts a steering angle in real time. The model then runs in autonomous mode and drives the car around Track 1 without going off the road.

## 2. Data Collection

We used the official Udacity self-driving car simulator on Track 1. Driving was done in manual mode while recording. We collected about five full laps of data, including straight driving, left turns, right turns, and recovery maneuvers. The simulator produced a `driving_log.csv` file and an `IMG` folder containing captured images.

## 3. Data Pre-Processing

All pre-processing was done in `data_preprocess.py`. Images were cropped, converted to YUV, blurred, resized, and normalized. Three camera images were used (center, left, right) with steering corrections.

## 4. Steering Angle Histogram

A histogram of steering angles was generated (`steering_histogram.png`). The dataset had many straight-driving frames. Left/right camera correction and augmentation helped balance this.

## 5. Model Architecture

The NVIDIA end-to-end driving model was used:

- Conv layers with 24, 36, 48, 64, 64 filters
- Flatten + Dense layers (100, 50, 10, 1)
- Dropout of 0.5

- MSE loss, Adam optimizer

## 6. Training Procedure

Training was done using `train_model.py`:

- Epochs: 10
- Batch size: 64
- Early stopping + model checkpointing

The final models are `model.h5` and `model_final.h5`.

## 7. Autonomous Driving (`drive.py`)

A PI controller regulates throttle. The simulator sends images, which are preprocessed and fed into the CNN. The model outputs the steering angle, and throttle is computed based on target speed (6 mph). The car can drive most of Track 1 autonomously.

## 8. Challenges

- Car drifting off-track at start: fixed by more data + left/right images
- Sudden steering jumps: fixed with PI controller + lower speed
- Data imbalance: fixed with augmentation and steering correction
- Environment issues: resolved library version conflicts

## 9. How to Run the Project

Training:

```
python train_model.py
```

Driving:

```
python drive.py model.h5
```

## 10. Conclusion

The project demonstrates a complete pipeline for autonomous steering using CNNs. The car can successfully drive around Track 1, showing that the model learned meaningful visual-driving relations.