CNN Powered Autonomous Vehicle

Sumanth Donthula(A20519856)

- Prof. Yan Yan

Project Scope and Objectives:

Collecting real time image data from cars is a little bit tedious process and it costs a lot.

Behavioral cloning is a method of reproducing the setting of reproducing the real time scenarios using simulator.

To collect image data from the simulator and train selected models and compare performance metrics.

Architecture of real system cloned:

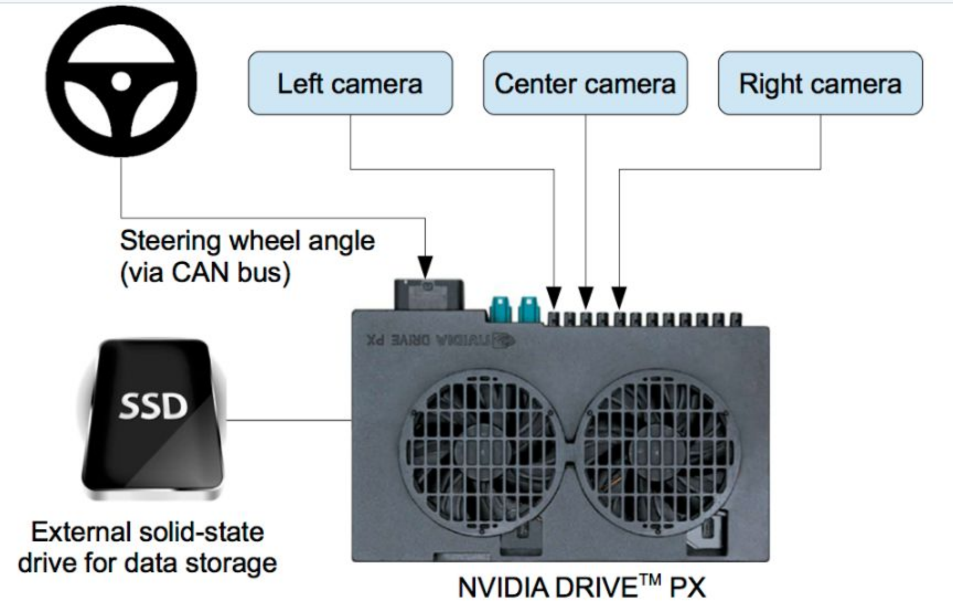


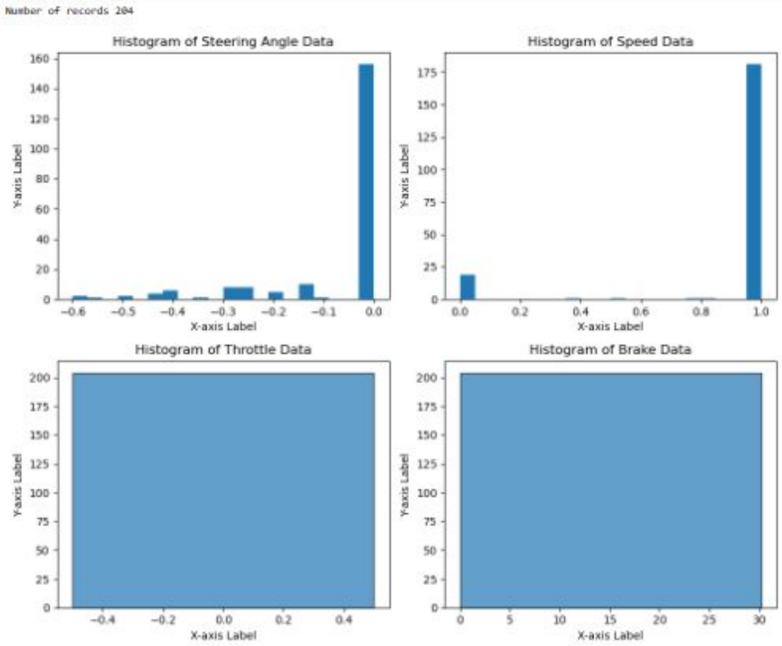
Fig: Unity Simulator and Track Used

Data Acquisition and Preparation:

When the simulator is being used in training mode it creates data logs and necessary images, the architecture of the process follows this methodology. The logs contains Image paths, features like Inverse of Steering Angle(1/R), Speed, Throttle and Break.

\home\cc\data\IMG\center_2023_11_10_10_21_18_838.j	\home\cc\data\IMG\left_2023_11_10_10_21_18_838.j	\home\cc\data\IMG\right_2023_11_10_10_21_18_838.j	0	0	0	7.93E-05
\home\cc\data\IMG\center_2023_11_10_10_21_18_946.j	\home\cc\data\IMG\left_2023_11_10_10_21_18_946.j	\home\cc\data\IMG\right_2023_11_10_10_21_18_946.j	0	0	0	7.80E-05
\home\cc\data\IMG\center_2023_11_10_10_21_19_067.j	\home\cc\data\IMG\left_2023_11_10_10_21_19_067.j	\home\cc\data\IMG\right_2023_11_10_10_21_19_067.j	0	0	0	7.87E-05
\home\cc\data\IMG\center_2023_11_10_10_21_19_168.j	\home\cc\data\IMG\left_2023_11_10_10_21_19_168.j	\home\cc\data\IMG\right_2023_11_10_10_21_19_168.j	0	0	0	7.78E-05
\home\cc\data\IMG\center_2023_11_10_10_21_19_294.j	\home\cc\data\IMG\left_2023_11_10_10_21_19_294.j	\home\cc\data\IMG\right_2023_11_10_10_21_19_294.j	0	0	0	7.84E-05
\home\cc\data\IMG\center_2023_11_10_10_21_19_414.j	\home\cc\data\IMG\left_2023_11_10_10_21_19_414.j	\home\cc\data\IMG\right_2023_11_10_10_21_19_414.j	0	0	0	7.80E-05
\home\cc\data\IMG\center_2023_11_10_10_21_19_535.j	\home\cc\data\IMG\left_2023_11_10_10_21_19_535.j	\home\cc\data\IMG\right_2023_11_10_10_21_19_535.j	0	0	0	7.88E-05
\home\cc\data\IMG\center_2023_11_10_10_21_19_671.j	\home\cc\data\IMG\left_2023_11_10_10_21_19_671.j	\home\cc\data\IMG\right_2023_11_10_10_21_19_671.j	0	0	0	7.77E-05
\home\cc\data\IMG\center_2023_11_10_10_21_19_795.j	\home\cc\data\IMG\left_2023_11_10_10_21_19_795.j	\home\cc\data\IMG\right_2023_11_10_10_21_19_795.j	0	0	0	7.82E-05
\home\cc\data\IMG\center_2023_11_10_10_21_19_907.j	\home\cc\data\IMG\left_2023_11_10_10_21_19_907.j	\home\cc\data\IMG\right_2023_11_10_10_21_19_907.j	0	0	0	8.00E-05
\home\cc\data\IMG\center_2023_11_10_10_21_20_037.j	\home\cc\data\IMG\left_2023_11_10_10_21_20_037.j	\home\cc\data\IMG\right_2023_11_10_10_21_20_037.j	0	0.2468073	0	0.155001
\home\cc\data\IMG\center_2023_11_10_10_21_20_180.j	\home\cc\data\IMG\left_2023_11_10_10_21_20_180.j	\home\cc\data\IMG\right_2023_11_10_10_21_20_180.j	0	0.6304296	0	0.935436
\home\cc\data\IMG\center_2023_11_10_10_21_20_278.j	\home\cc\data\IMG\left_2023_11_10_10_21_20_278.j	\home\cc\data\IMG\right_2023_11_10_10_21_20_278.j	0	0.972492	0	1.977818
\home\cc\data\IMG\center_2023_11_10_10_21_20_379.j	\home\cc\data\IMG\left_2023_11_10_10_21_20_379.j	\home\cc\data\IMG\right_2023_11_10_10_21_20_379.j	0	1	0	3.157355
\home\cc\data\IMG\center_2023_11_10_10_21_20_500.j	\home\cc\data\IMG\left_2023_11_10_10_21_20_500.j	\home\cc\data\IMG\right_2023_11_10_10_21_20_500.j	-0.1405746	0.636028	0	4.406077
\home\cc\data\IMG\center_2023_11_10_10_21_20_623.j	\home\cc\data\IMG\left_2023_11_10_10_21_20_623.j	\home\cc\data\IMG\right_2023_11_10_10_21_20_623.j	0	0.270622	0	4.866705
\home\cc\data\IMG\center_2023_11_10_10_21_20_739.j	\home\cc\data\IMG\left_2023_11_10_10_21_20_739.j	\home\cc\data\IMG\right_2023_11_10_10_21_20_739.j	-0.176424	0	0	4.868411
\home\cc\data\IMG\center_2023_11_10_10_21_20_853.j	\home\cc\data\IMG\left_2023_11_10_10_21_20_853.j	\home\cc\data\IMG\right_2023_11_10_10_21_20_853.j	-0.5167023	0	0	4.765398
\home\cc\data\IMG\center_2023_11_10_10_21_20_978.j	\home\cc\data\IMG\left_2023_11_10_10_21_20_978.j	\home\cc\data\IMG\right_2023_11_10_10_21_20_978.j	-0.1394973	0	0	4.712336

EDA on Bad Data< One Lap



EDA on Good Data: 3 to 4 laps

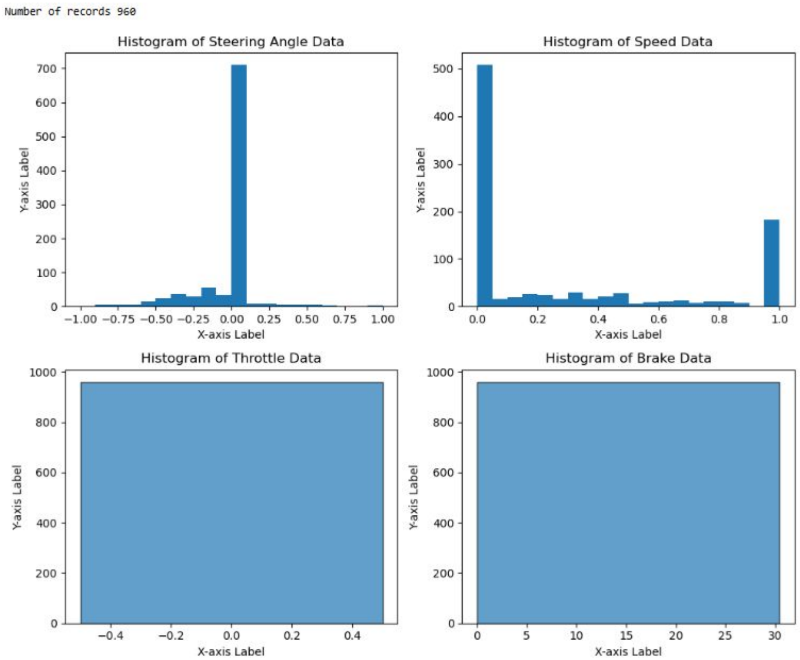
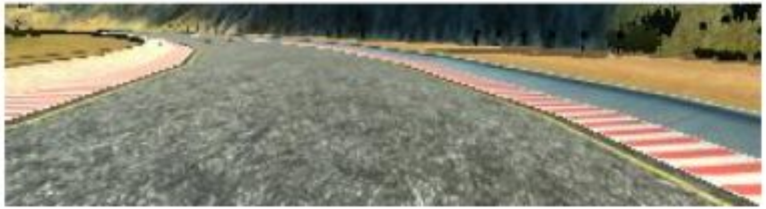
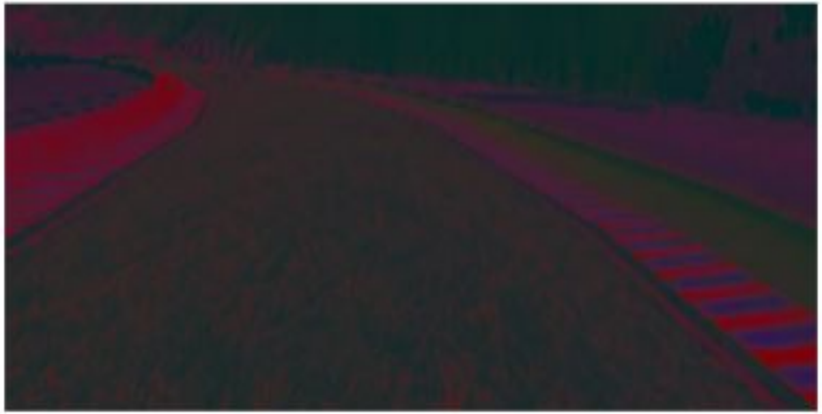
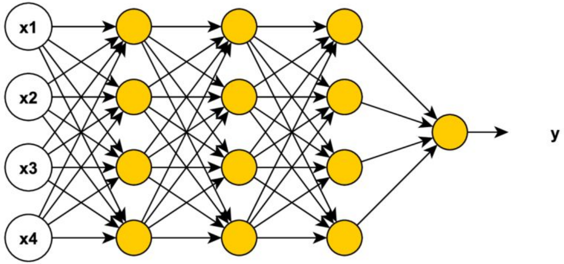


Image PreProcessing





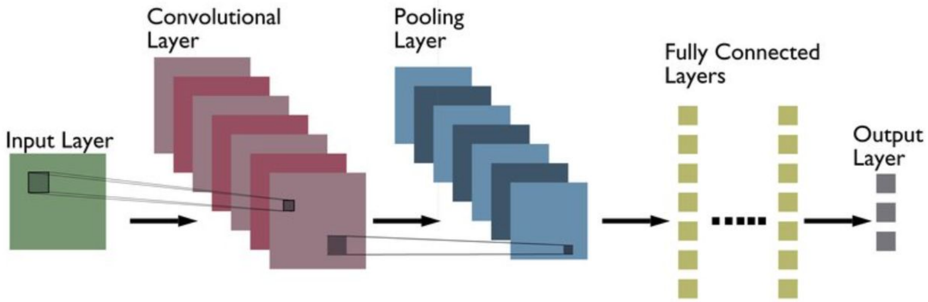
Model Architecture, DNN




```
def buildModel(keepProb):
    model = Sequential()  # Create a Sequential model (a linear stack of layers).
    model.add(Lambda(lambda x: x / 127.5, input_shape=inputShape))
    # Lambda layer to normalize the input data.

    # Add several fully connected layers with ELU activation.
    model.add(Dense(24, activation='elu'))
    model.add(Dense(36, activation='elu'))
    model.add(Dense(48, activation='elu'))
    model.add(Dense(64, activation='elu'))
    model.add(Dense(64, activation='elu'))
    model.add(Flatten())  # Flatten the output from the previous layers.
    # Add more fully connected layers with ELU activation.
    model.add(Dense(100, activation='elu'))
    model.add(Dense(50, activation='elu'))
    model.add(Dense(10, activation='elu'))

    model.add(Dense(1))  # Output layer with 1 unit (for regression tasks).

    model.summary()  # Display a summary of the model architecture.

    return model  # Return the constructed neural network model.
```

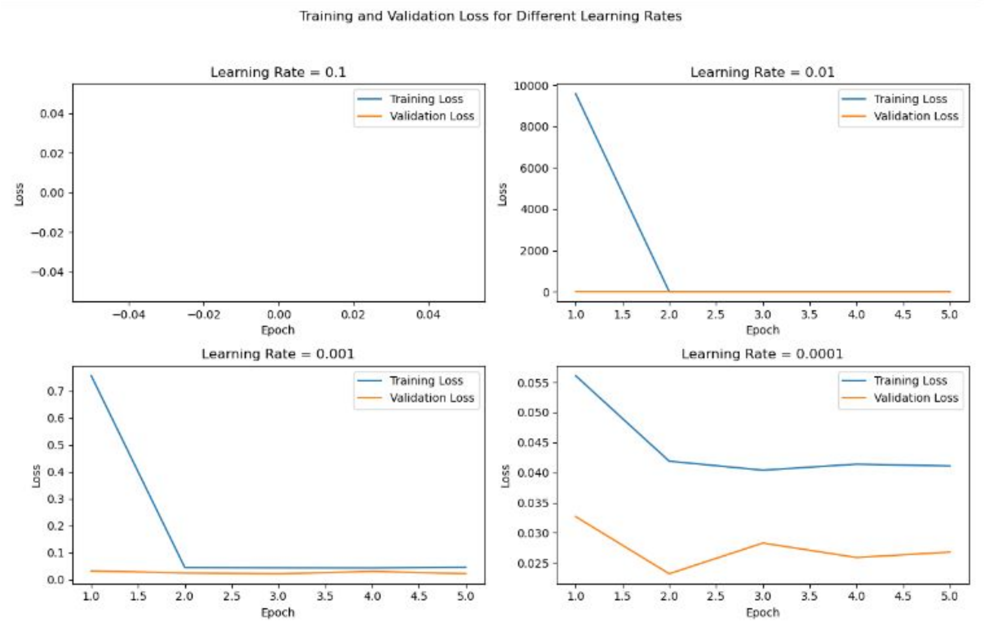
Training Process and Optimization, DNN:

Model Architecture, CNN



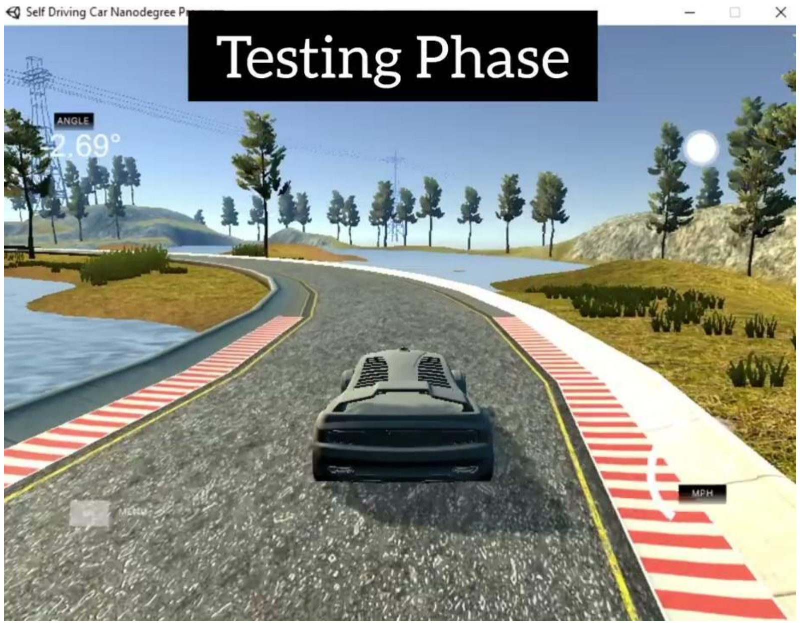

```
def buildModel(keepProb):
    model = Sequential()
    model.add(Lambda(lambda x: x / 127.5, input_shape=inputShape))
    model.add(Conv2D(24, (5, 5), activation='elu', strides=(2, 2)))
    model.add(Conv2D(36, (5, 5), activation='elu', strides=(2, 2)))
    model.add(Conv2D(48, (5, 5), activation='elu', strides=(2, 2)))
    model.add(Conv2D(64, (3, 3), activation='elu'))
    model.add(Conv2D(64, (3, 3), activation='elu'))
    model.add(Flatten())
    model.add(Dense(100, activation='elu'))
    model.add(Dense(50, activation='elu'))
    model.add(Dense(10, activation='elu'))
    model.add(Dense(1))
    model.summary()
    return model
```

Training Process and Optimization, CNN:



Results and Performance Metrics:

- throttle = 0.4- steeringAngle**2 - (speed/speedLimit)**2
- sendControl(steeringAngle, throttle)



Challenges & Future Work:

- Working with Self Driving cars, is a very sensitive use case and even small error in the model will result in trivial situations.
- Future work involves refining real-time adaptability to handle increasingly dynamic environments and unpredictable scenarios, ensuring continued precision in autonomous navigation.
- Exploring the integration of emerging algorithms, particularly reinforcement learning variants, stands as a future avenue to further enhance the adaptability and intelligence of our autonomous driving system.
- Continued research in future iterations will focus on advancing multi-sensor data fusion techniques, pushing the boundaries of perception accuracy for improved resilience in varied driving conditions.
- For future scalability, our roadmap includes optimizing hardware efficiency, exploring edge computing solutions, and actively engaging with regulatory frameworks to propel our autonomous driving project into the forefront of industry evolution.