Robotics, Artificial Intelligence
and Real-time Systems

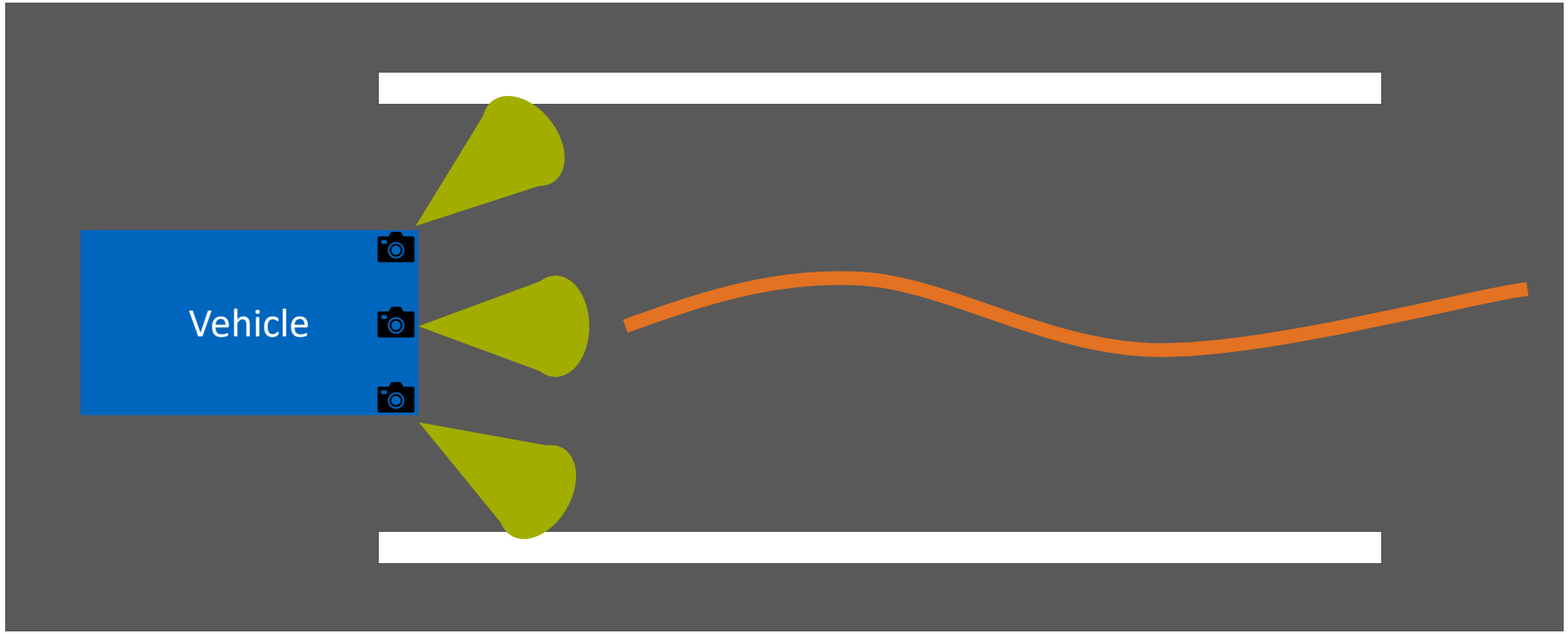
Masterpraktikum - Simulation-Based Autonomous Driving in Crowded City

# End to End Learning for Self-Driving Cars

Jiachen Lu
Garching bei München
02.03.2024

# Problem Statement

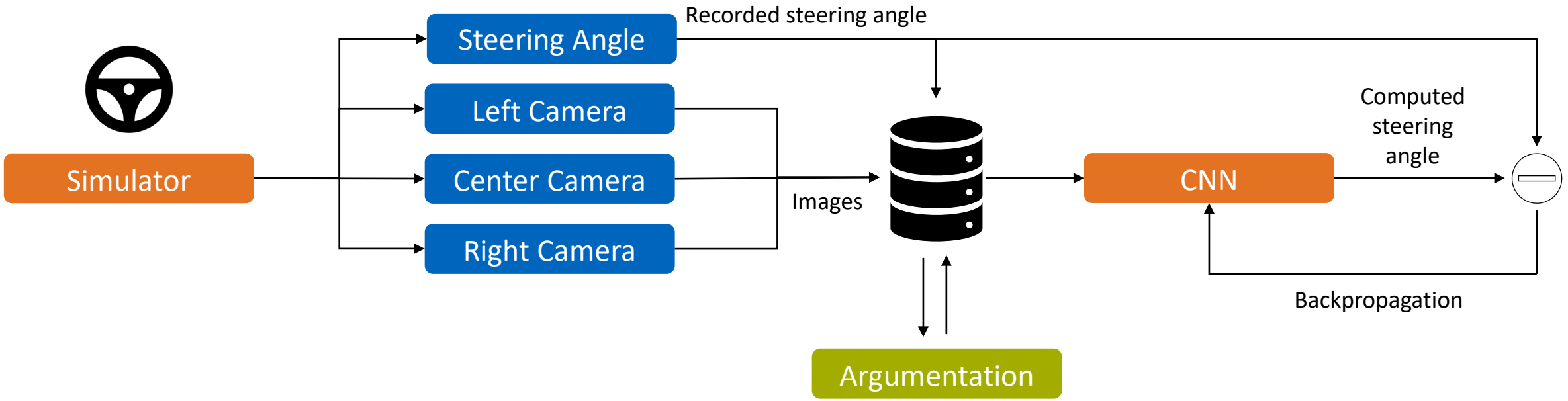
- Train a **end-to-end learning model** → **Car drive itself** in the simulator and the real world



# Problem Statement

- Train a **end-to-end learning model** → **Car drive itself** in the simulator and the real world

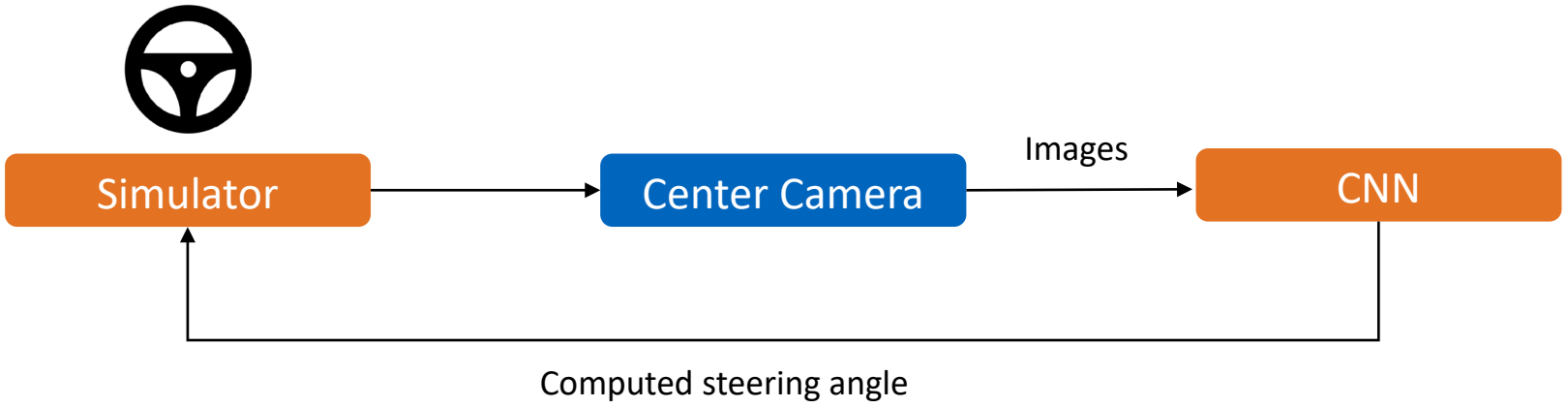
- When Training



# Problem Statement

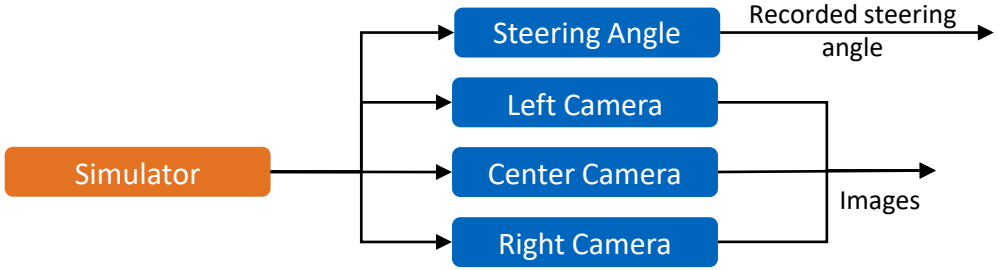
- Train a **end-to-end learning model** → **Car drive itself** in the simulator and the real world

- When Testing

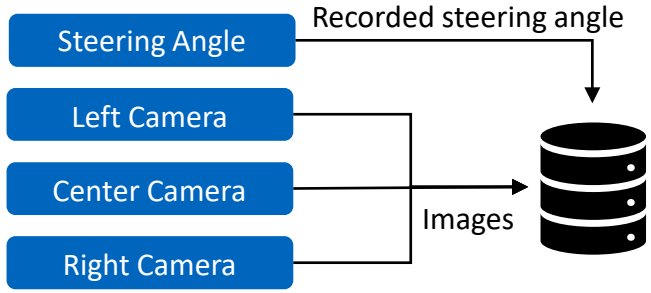


# Simulator

- UDACITY Simulator
  - Training Mode
    - Manual driving
    - Data acquisition
  - Autonomous Mode
    - Automated driving
    - Model testing
  - Two tracks

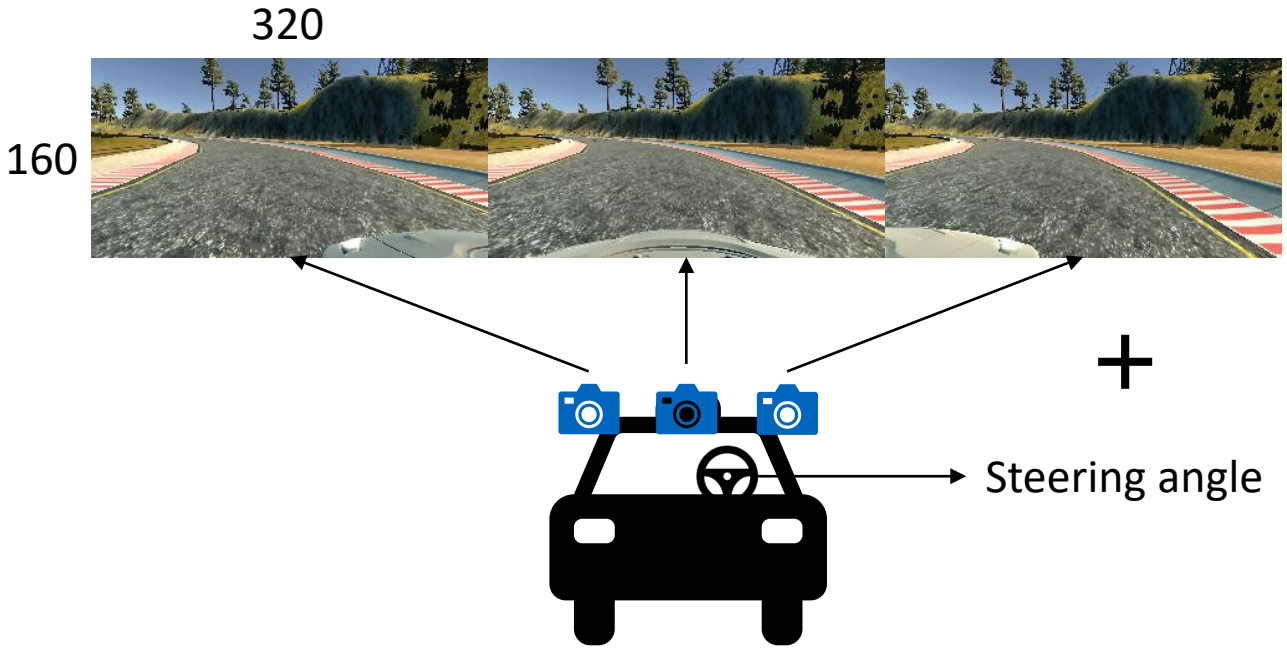



# Data Acquisition




- At each time step $t$

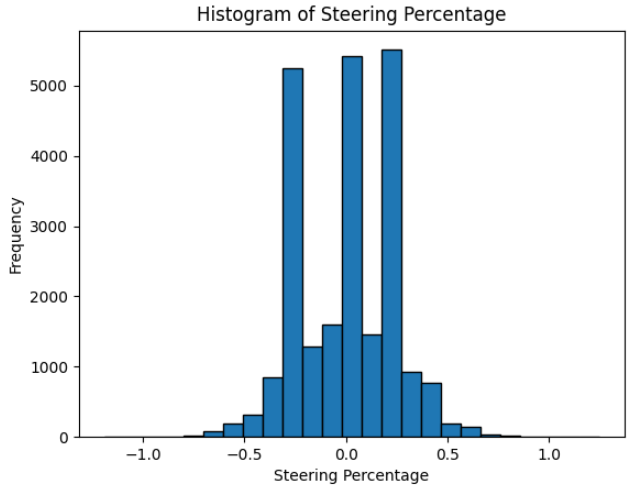
| Images | Steering angle |
|---|---|
| $I_{Center}$ | $S = S_{Record}$ |
| $I_{Left}$ | $S = S_{Record} + 6.25°$ |
| $I_{Right}$ | $S = S_{Record} - 6.25°$ |



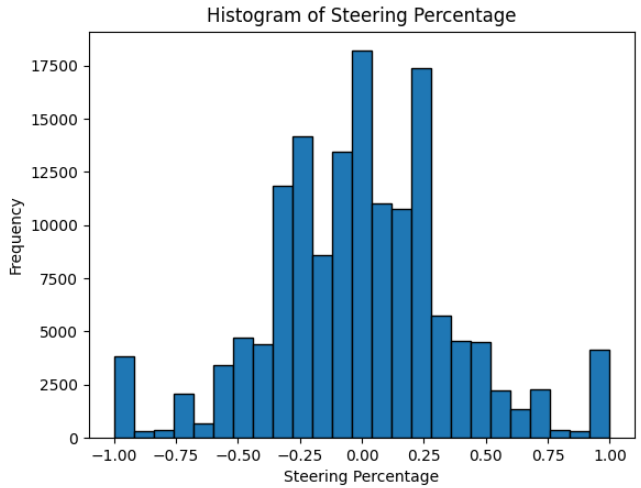
# Data Acquisition

- Data set from UDACITY
  - **Downloaded**
  - **Small Dataset**: 24 thousand
  - **Data imbalance:**
    - Too much frames at $\mathbf{Steering\ angle} = \mathbf{0}°$
    - Too less frames at $|\mathbf{Steering\ angle}| > \mathbf{12.5}°$

- Data set from manual acquisition
  - **Xbox Controller**
  - **Dataset**: 150 thousand




Data set from UDACITY




Data set from manual acquisition


# Data Augmentation

- Data set from manual acquisition
  - Different Track → **Influencing factors**
    - Lighting, shadows
    - road conditions, road markings

- Data Augmentation
  - Flip
  - Translate
  - Brightness
  - Random shadow
  - Random erasing

Original

Flip

Translate

Brightness

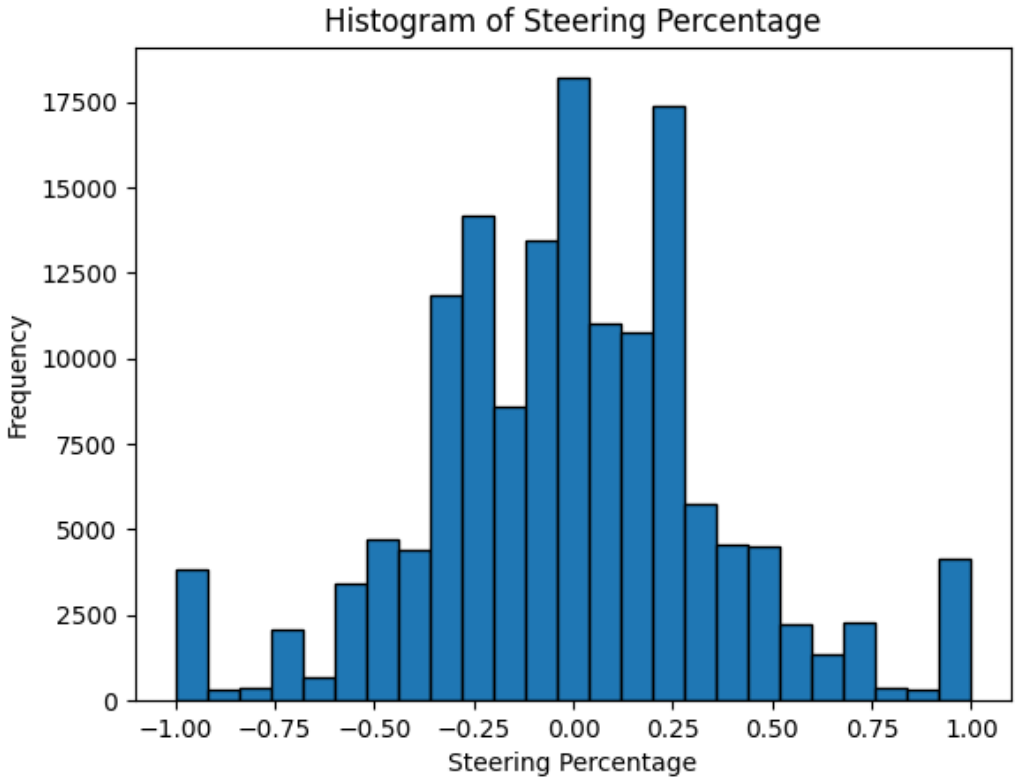
Random shadow

Random erasing


# Data Augmentation

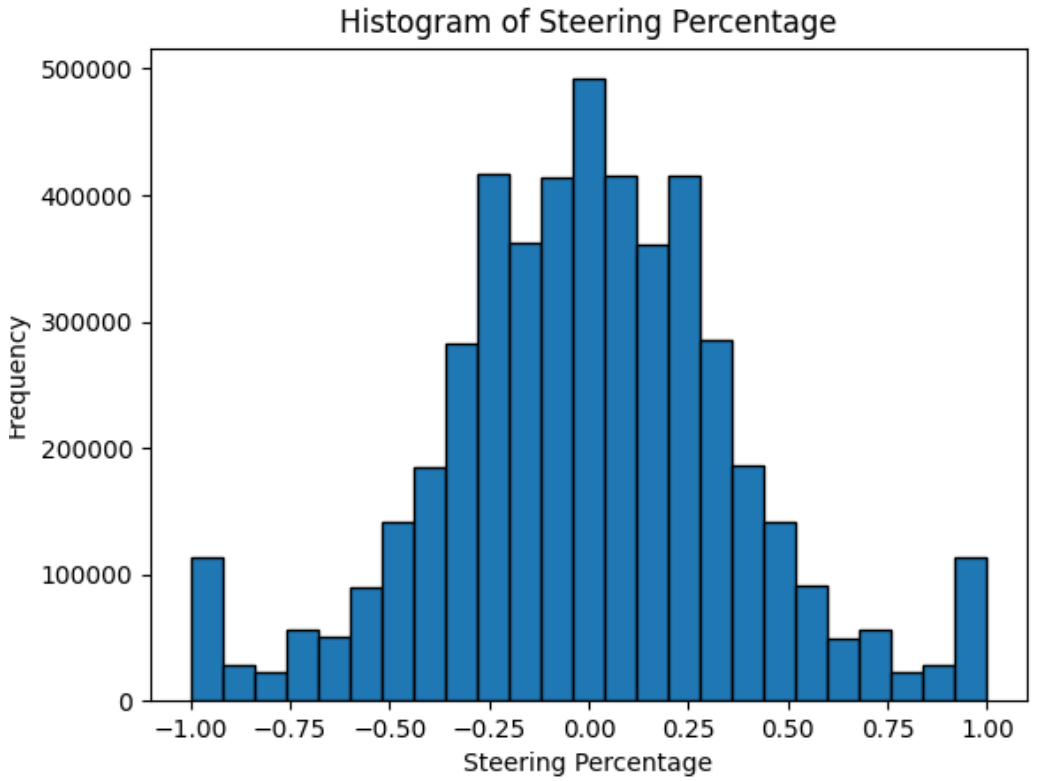
## Data set before Augmentation

- **Dataset:** 150 thousand
- **Steering angle imbalance**

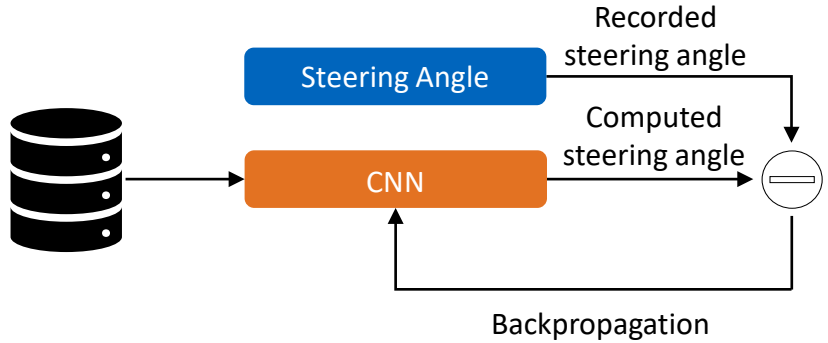


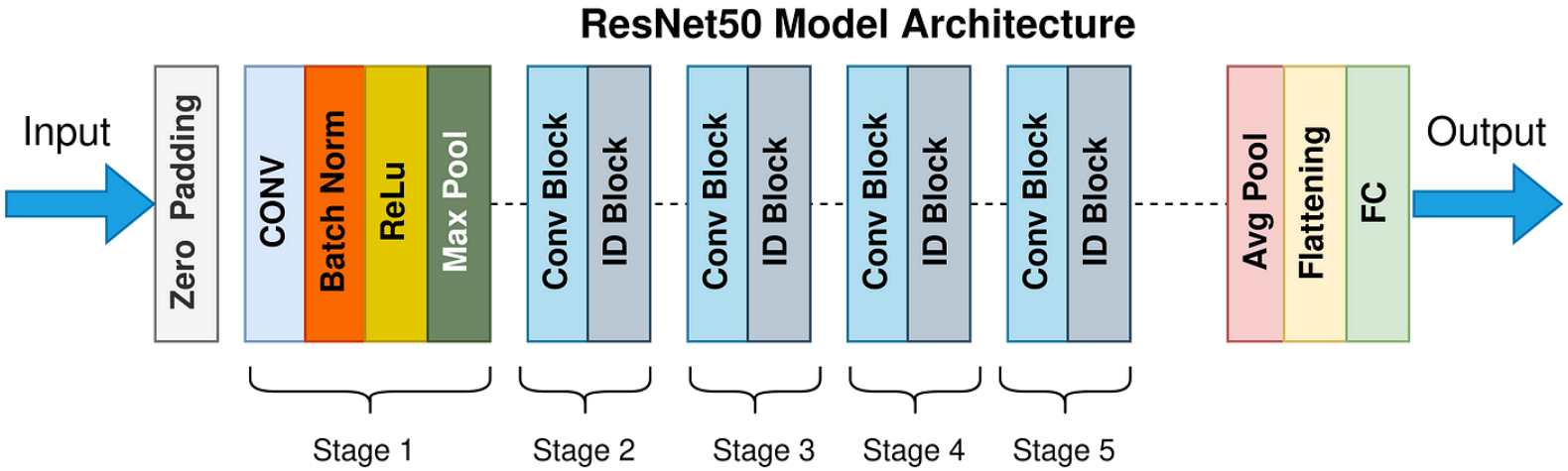

## Data set after Augmentation

- **Dataset:** 4.82 million
- **Steering angle balance**



# ResNet Backbone Network



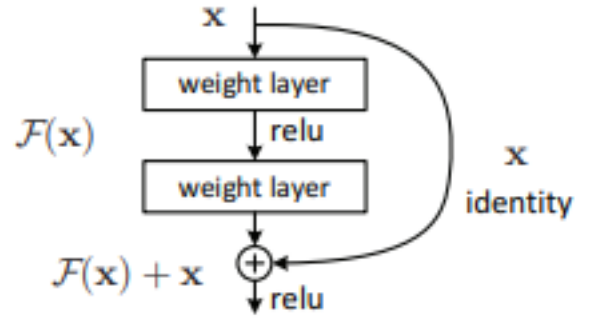
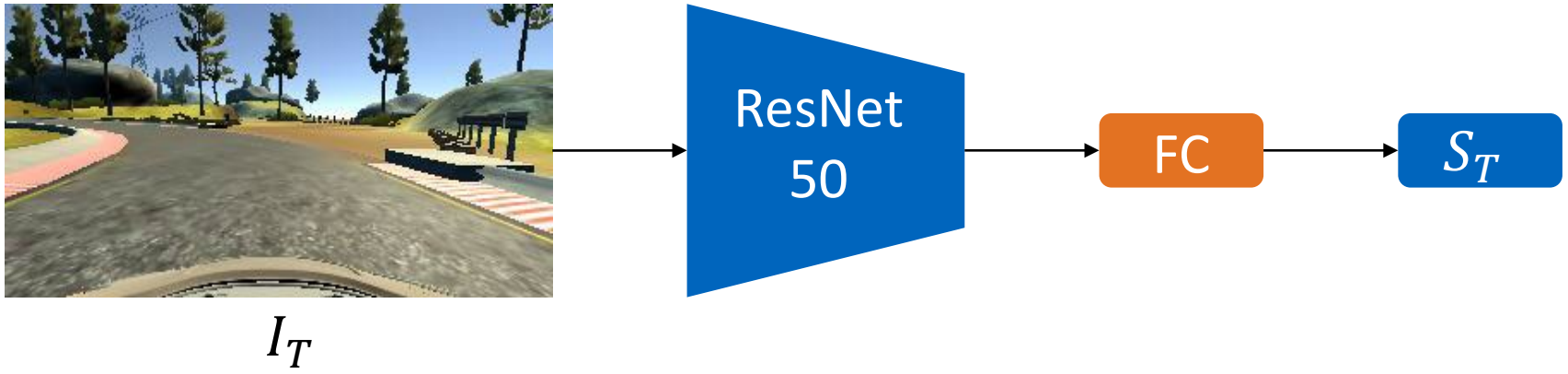




Figure 2. Residual learning: a building block.

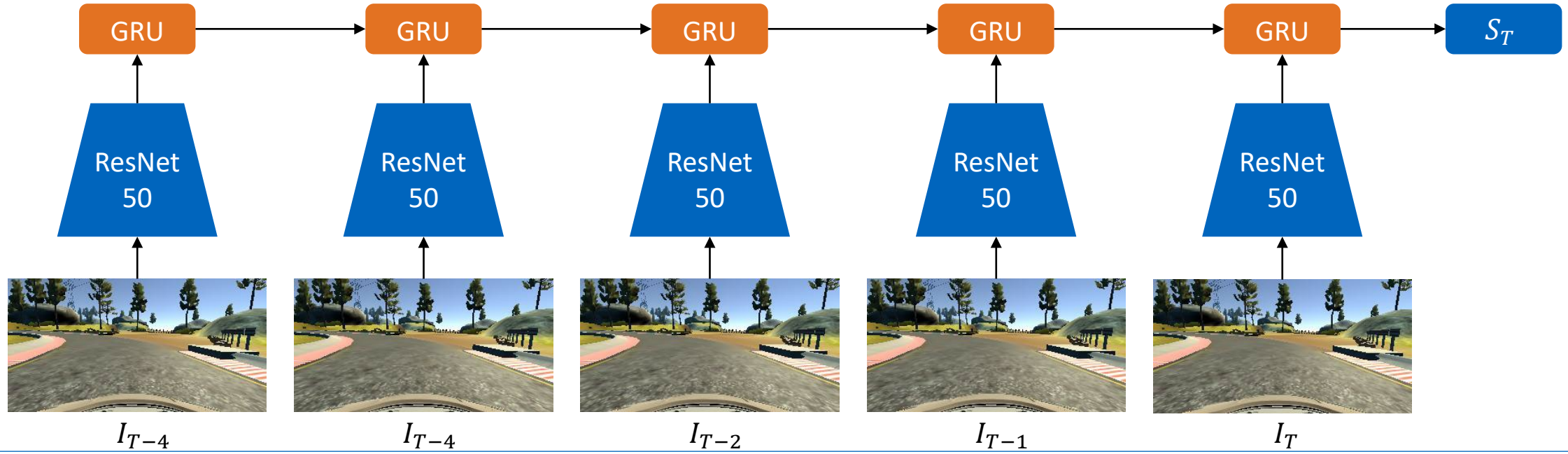
# ResNet50

- **Idea**: Directly use the image of the current time step $I_t$ to predict the steering angle $S_t$
- **Methodology**:
  - $I_t$ is fed into the **ResNet50 backbone network** to extract image features
  - Then, the additional FC layers will predict the steering angle $S_t$



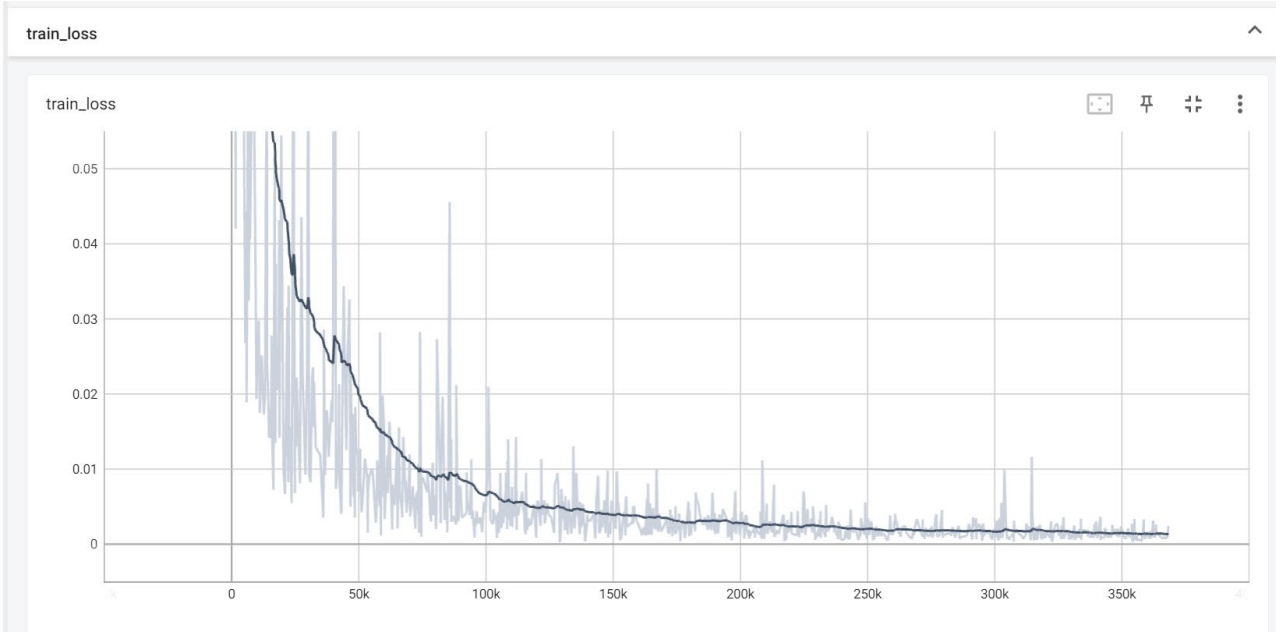
# ResNet50 + GRU

- **Idea**: Steering angle $S_t$ should be based on the images from the past seconds, i.e. $I_{t-k:t}$
- **Methodology**:
  - $I_{t-4:t}$ are fed into the **ResNet50 backbone network** to extract image features
  - Then, these features will be fed to the **GRU module** to predict the steering angle $S_t$



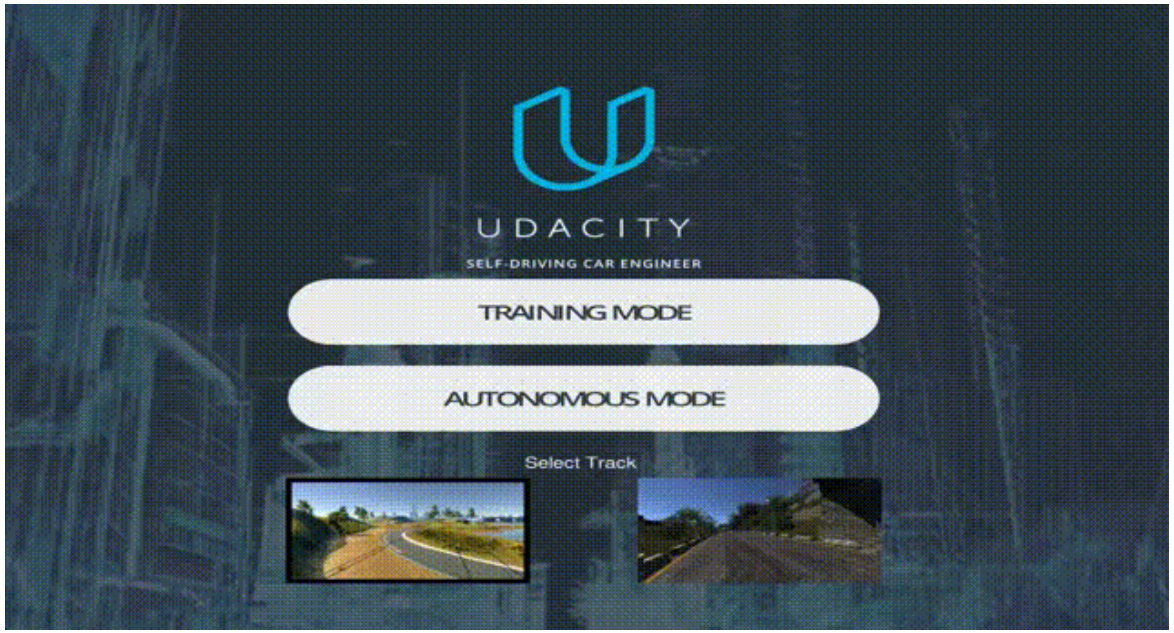
# Training

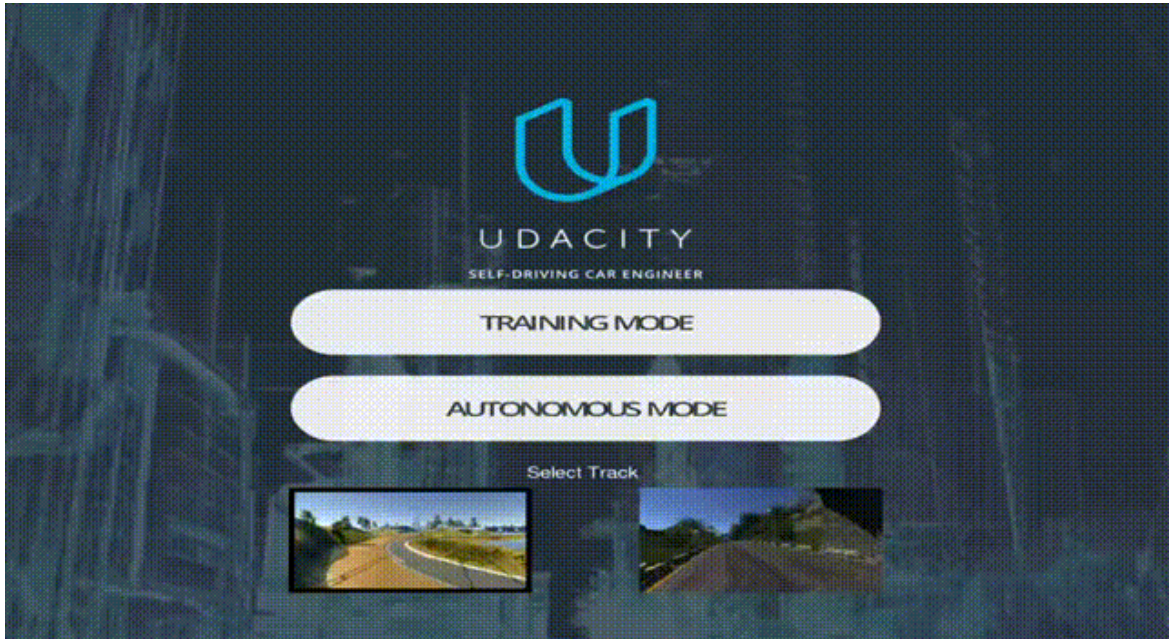
- **Environment**
  - Python 3.8
  - Pytorch 1.11.0 + Pytorch-lightning 2.0.2
  - GPU: 2 x 3090 24GB
- **Training Parameter**
  - Loss: L2 Loss
  - Optimizers: Adam
  - Learning rate: 0.001
  - Batch size: 64
  - Epoch: 20



# Qualitative Results

**All results are based on the ResNet50**



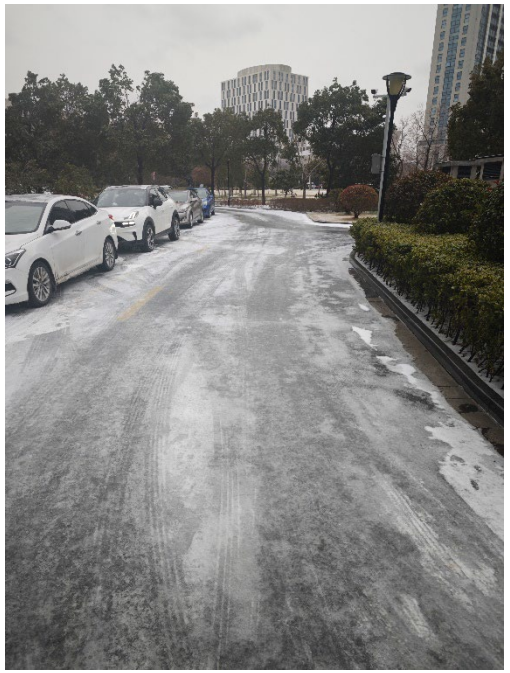



# Real World Cars

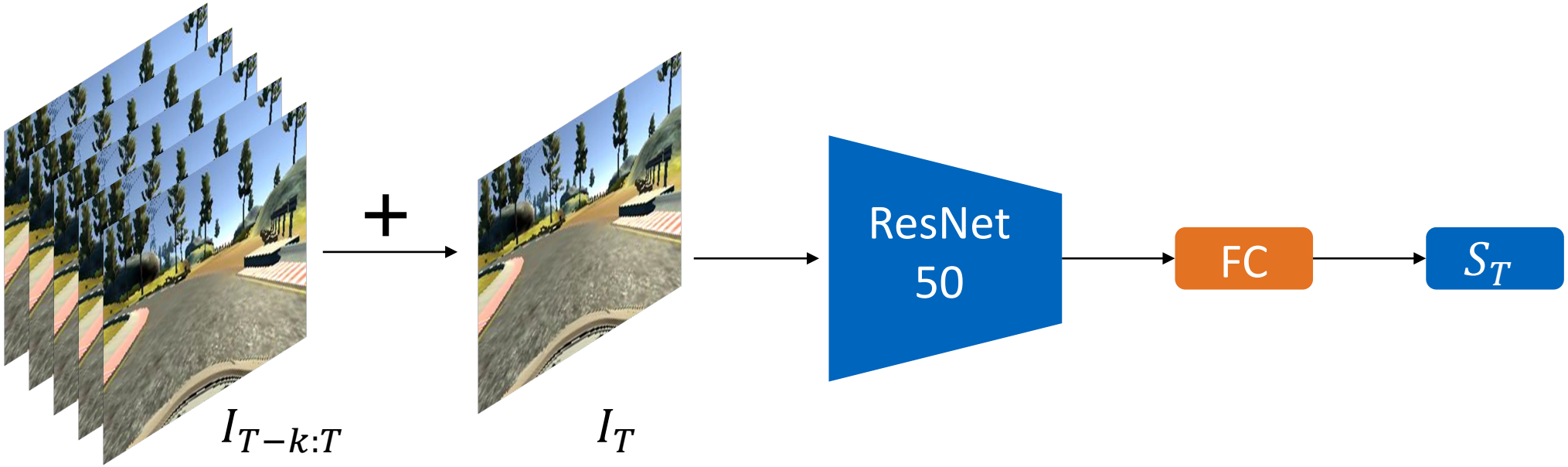
- Real World Cars
  - Similar to the simulator
  - Train ResNet50-based Network
  - Poor generalization
  -

# Real World Cars

- Real World Cars: **Problem**
  - More complex decisions
  - More complex environments
  - Sense-Plan-Act



# Further Work



# Thanks for your listening!