SELF DRIVING CARS SOFTWARE

Perception: Detect Every Obstacle & Road Situation

Obstacle Detection (Vehicles, Pedestrians, Traffic Lights, Signs, ...)

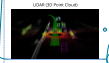
COMPUTER VISION ENGINEER

Camera (Images)

- Classify Obstacles
- Localize Obstacles
  - Using 2D Bounding Boxes
  - Using 3D Bounding Boxes
- Track Obstacles through time

PERCEPTION ENGINEER (LIDAR)

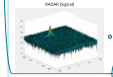
LIDAR (3D Point Cloud)



- Classify Obstacles
- Localize Obstacles
- Estimate obstacles velocities
- Track Obstacles through time

PERCEPTION ENGINEER (RADAR)

RADAR (Signal)



- Classify Obstacles
- Localize Obstacles
- Estimate obstacles velocities

Road & Lane Line Detection

COMPUTER VISION ENGINEER

Camera (Images)

- Drivable Area Segmentation (2D)
- Lane Line Detection

PERCEPTION ENGINEER (LIDAR)

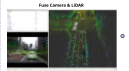
LIDAR (3D Point Cloud)

- Lane Line Detection
- Drivable Area Segmentation (3D)

Sensor Fusion:
Fuse Data from all sensors to have a unique estimation for each obstacle

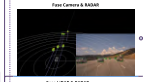
SENSOR FUSION ENGINEER

Fuse Camera & LIDAR



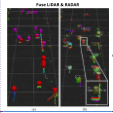
Get fused position and classification of each obstacle

Fuse Camera & RADAR



Get fused classification and velocity of each obstacle

Fuse LIDAR & RADAR



Get fused position and velocity of each obstacle

Skills Needed

- Computer Vision Engineer
  - Artificial Intelligence
    - Machine Learning
      - Deep Learning
        - Convolutional Neural Networks
          - Object Detection
          - Image Classification
          - Image Segmentation
    - Hungarian Matching
      - Obstacle Association
    - Bayes Filtering
      - Kalman Filters
        - Obstacle Tracking
- Perception Engineer (LIDAR)
  - 3D Segmentation Algorithms
    - RANSAC
    - Geometric Deep Learning
  - Obstacle Clustering Algorithms
    - Machine Learning
      - Clustering
        - DBSCAN
        - HDBSCAN
        - KMEANS
        - EUCLIDEAN
        - ...
- Perception Engineer (RADAR)
  - FMCW Waveform generation
  - CFAR
  - Fast Fourier Transform
  - Range Doppler Maps
- Languages
  - C++
  - Python
  - MATLAB
- ROS

Localization: Understand where is the vehicle in the world

Determine the position to the center of the lane

- Use Computer Vision to estimate the distance to center

LOCALIZATION ENGINEER

Determine the position of the vehicle in a Map

- Starting Point
  - Odometry
    - Estimate the distance driven in the last x meters
- No Starting Point (Kidnapped Vehicle)
  - GPS

    - Get a position with 1-10 m accuracy
  - GPS RTK (Real Time Kinematics)

    - Correct the position using triangulation
      - Get a position with 1-10 cm accuracy
  - Bayesian Filtering

    - PREDICTION
    - UPDATE
    - Particle Filters (Monte Carlo Localization)
      - Improve the position to 1-3 cm accuracy
        - Advantage: Multiple estimations possible
    - Markov Localization
    - Kalman Filters
      - Linear Kalman Filters
      - Extended Kalman Filters
      - Unscented Kalman Filters
      - Improve the position to 1-3 cm accuracy
        - Drawback: One estimation possible
- Ultra Wideband
  - Trilateration (similar to GPS) in areas where GPS is not working (tunnels, ...)
    - Estimate the position using UWB modules

Determine the position of the vehicle without a Map (SLAM)

- Localization
  - Extended Kalman Filters
  - Sparse Extended Information Filters
  - Extended Information Form
    - Estimate the relative position to obstacles
  - Fast SLAM
  - Graph SLAM
  - ...
- Mapping
  - Occupancy-Grid Mapping
    - Discretize the world
      - Estimate the position in a cell
  - 2.5D Maps
  - Extended Elevation Maps
  - ...

Skills Needed

- Artificial Intelligence
  - Bayesian Filtering
    - Kalman Filters
      - Linear Kalman Filters
      - Extended Kalman Filters
      - Unscented Kalman Filters
    - Particle Filters
  - Markov Localization
- C++
- ROS

Planning: Plan a route from A to B, respecting traffic rules and other vehicles

Behavioral Planning

- Determine the future positions of obstacles based on past scenarios
  - Data Driven Approach
- Determine the future position of obstacles based on current observations
  - Kalman Filters
- Determine the future positions of obstacles based on constraints
  - Model Based Approach

Search/ Decision Making

Discrete Planning: Discretize the environment

- Uninformed algorithms (no idea if getting closer to the objective)
  - Generate a path from A to B
  - Breadth First Search (BFS) - Search the grid broadly
  - Depth First Search (DFS) - Search the grid in depth
  - Uniform Cost Search (UCS) - Search the grid by computing costs
- Informed algorithms (know if they are getting closer)
  - Generate a path from A to B
  - A* - Search the grid using 'distance to goal' function
  - Hybrid A* - Search the grid using 'distance to goal' function and by making the world a bit more continuous

Sample Based Planning: Use realistic constraints

- Generate a path from A to B
  - Probabilistic Roadmap (PRM)
  - Rapidly Exploring Random Tree (RRT)

Probabilistic Planning: Use probabilities & reward functions

- Reinforcement Learning
  - Generate a path from A to B
  - Value Iteration Algorithm
  - Q Learning
  - Policy Iteration Algorithm
  - ...

Trajectory Generation

- Generate a curve/spline between nodes or point A and B

MOTION PLANNING ENGINEER

Control: Control the steering wheel and acceleration pedal to move the car

Control Command

- PID (Proportional Integral Derivative)
- MPC (Model Predictive Control)
- Linear Quadratic Regulator (LQR)
- ...

CONTROL COMMAND ENGINEER

- Generate a steering wheel and acceleration value to follow the trajectory

Skills Needed

- Artificial Intelligence
  - Reinforcement Learning
    - Value Iteration
    - Policy Iteration
    - ...
  - Bayes Filtering
    - Kalman Filters
    - ...
  - Machine Learning
    - Basic Machine Learning Algorithms for classification, regression, clustering
  - Problem Solving
    - Search Algorithms
    - Graph Theory
- C++, Python
- ROS