

Master's project I: Trajectory Prediction (Pedestrian/Human, Cyclist, Car)

Task Description:

Trajectory prediction of different road users in mixed-traffic scenarios is attracting huge interest due to its importance in different artificial intelligence powered fields like autonomous vehicles and service/ co-working robots. The challenge lies within predicting accurate yet realistic trajectories (e.g., interactions, avoiding conflict/near-conflict situations).

Within this project, we will go through this process of predicting/forecasting trajectories with one road user as our prediction focus.

N.B by trajectory, we mean x and y coordinate positions of a certain road user at a certain time step. From now on we will use "pedestrian" as an example of a "road user".

Choose a dataset that provides the most important information (i.e., obstacle positions, destination points, group information).

Sample data source: <https://github.com/crowdbotp/OpenTraj>

Tasks:

1. Prepare a small report/presentation about the dataset (with visualizations).
2. Extract the required features:
 - a. pedestrians' speed (Newtonian) at each time-steps (if not given)
 - b. euclidean distance (and/or direction vector) from the destination points (if not given, think about how to extract it)
 - c. euclidean distance from the nearest obstacle point (if obstacle positions are given)
 - d. euclidean distance from the nearest pedestrian (neighbor). Here, a neighbor pedestrian can be defined as a pedestrian who's present in the scene at the same time-step.
 - e. euclidean distance from group members (if group info is given)
 - f. And so on. N.B. **It is not mandatory to extract all the features mentioned above (except a, b, and d).**
3. Design a trajectory prediction model that can: for each certain pedestrian predict the next 4.8 seconds of the trajectory (12 time-steps at 2.5 fps) with a maximum of 3.2 seconds (8 time-steps at 2.5 fps) of previous trajectory data (motion history) and other extracted features as input.
4. Use at least two different algorithms (Machine learning) to design the prediction model. i.e., two different versions of the prediction model with two different algorithms. Compare the performance of each of the models.
5. Use the following metrics to evaluate the model performance:
 - a. **Average displacement error (ADE):** ADE is the mean of all euclidean distance values between the real and predicted values at each time-steps. Lower is better.
 - b. **Final displacement error (FDE):** FDE is the distance between the last predicted coordinate (x, y) and the last real coordinate (x, y) of the primary pedestrian. Lower is better.

- c. **percentage of near-collisions:** Calculate the percentage of near-collisions in the model predicted trajectories of the pedestrians. A near-collision takes place when the Euclidean distance between the pedestrians is below 0.1 meters.
- 6. Take a sample scenario (with at least four pedestrians) from the dataset and visualize (simple) the predicted and real trajectories to qualitatively compare model performance.