Near Real-Time Generation of Microscopic Traffic Simulations from Object Lists

Master‘s Thesis of Lekha Kosuri

Mentoring:
M. Sc. Alexander Kutsch
M. Sc. Mario Ilic

Motivation and Objectives

Digital Twins of traffic data offer a range of benefits in the fields of advanced traffic management, intelligent transportation systems, and autonomous vehicle testing. To achieve it, a high-fidelity and real-time traffic simulation of sensor data is required. However, object lists derived from sensor data using deep learning methods contain various errors and require a pipeline that can accurately match the coordinates, filter out errors, and simulate them in real-time within simulation software like SUMO. The primary objective of this thesis is to develop a robust pipeline that processes raw object lists from drone data into microscopic traffic simulations within SUMO in near real-time. Additionally, the thesis aims to establish a framework for identifying and filtering out object list errors, as well as setting up appropriate coordinate transformations.

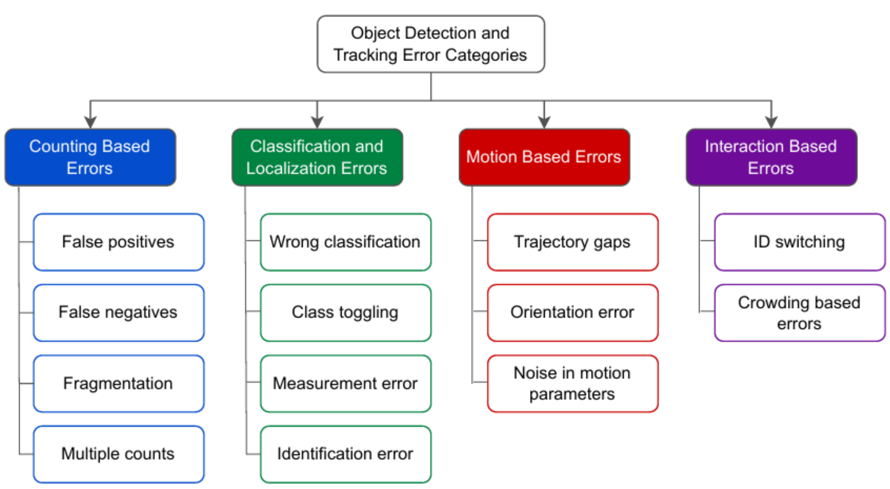

Fig. 1. Object detection and tracking based error categories

Taxonomy of Errors

Object lists and trajectories produced from deep learning-based techniques often contain various errors that are difficult to correct within the techniques. Creating a logic-based filtering mechanism for these errors addresses this problem, along with the development of robust safety mechanisms for enhanced reliability, designed based on the ground truth. This thesis categorizes possible errors into four categories based on the similarity in the consequences of the errors and their solutions as shown in **Fig. 1**. Counting-based errors occur when a false density of traffic is reported, for example, due to false positives or multiple counts. Classification and localization errors lead to inaccurate class and size identifications. Motion-based errors lead to faulty trajectories and vehicle movements. Interaction-based errors are primarily caused by the interaction of traffic objects due to their closeness or crowding. Logic-based solutions that utilize physical constraints and counteract typical error patterns are suggested to eliminate them.

Methodology

The methodology of this thesis is illustrated in Fig. 2. A SUMO network file is created using the boundaries from the OSM map, along with a decal of an orthophoto of the test site and reference coordinate points to align the decal. The pipeline is created as a Python script using the TraCI interface, using two object insertion methodologies. The pure-

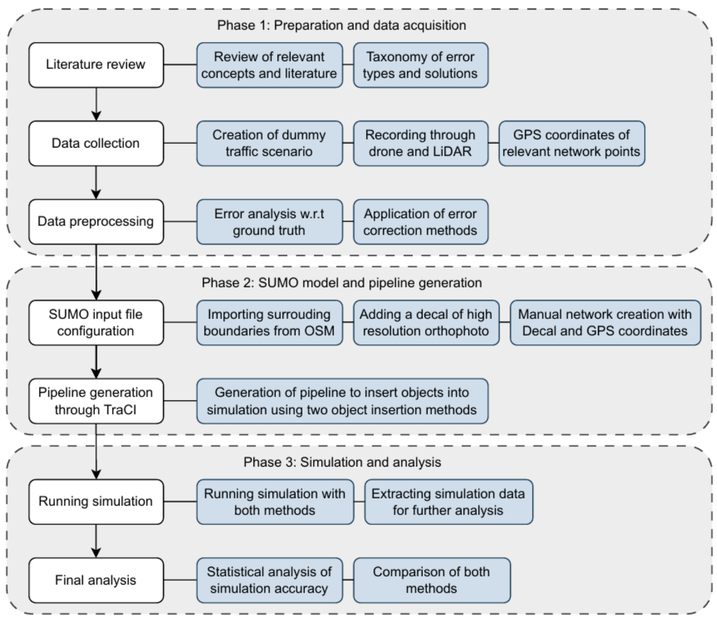

Fig. 2. Methodological workflow

positional method inserts objects based solely on their true coordinates. The network-constrained method utilizes a specific argument (keepRoute = 0) to snap objects continuously to the nearest allowed edges within a 5-meter radius. The pipeline includes additional filters to remove parked vehicles and to perform coordinate position correction. Additional mechanisms are used to insert pedestrians due to their different functioning in the interface.

Results and Conclusion

The pipeline is able to successfully simulate the ground truth scenario by online manipulation of objects during the simulation run. The results, as shown in Fig. 3, show that the pure-positional method is able to more accurately recreate the object trajectories, but due to network-related errors, might appear to have erroneous placement with respect to the network infrastructure. While the network-constrained method can properly align objects with respect to road boundaries, it introduces jitters in movements during online snapping. The first method is suggested by the thesis, with future research to focus on improvements in more accurate network creation. In conclusion, possible improvements and future research directions involving integration with other sensors are discussed.

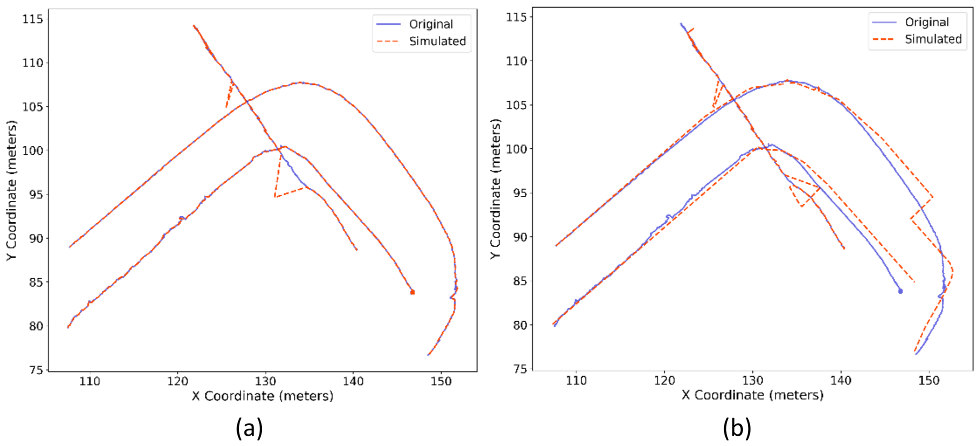

Fig. 3. (a) Trajectories after pure positional injection of Scenario 1 dataset (b) Trajectories after network- constrained injection of Scenario 1 dataset