Summer Reading 9: StopNet: Scalable Trajectory and Occupancy Prediction for Urban Autonomous Driving

Social Robot Navigation Project @ Bot Intelligence Group

Paper:
Summer Reading 9: StopNet: Scalable Trajectory and Occupancy Prediction for Urban Autonomous Driving
Jinkyu Kim1, Reza Mahjourian2, Scott Ettinger2, Mayank Bansal2,
Brandyn White2, Ben Sapp2, and Dragomir Anguelov2

Summary:

Abstract

- The authors propose StopNet
    - StopNet: A scalable motion forecasting method that accommodates sparse inputs in a whole-scene modeling framework, and co-trains trajectory and occupancy representations.
- A whole-scene sparse input representation allows StopNet to scale to predicting trajectories for hundreds of road agents with reliable latency.
- In addition to predicting trajectories, our scene encoder lends itself to predicting whole-scene probabilistic occupancy grids, a complementary output representation suitable for busy urban environments.
    - Occupancy grids allow the AV to reason collectively about the behavior of groups of agents without processing their individual trajectories.

Introduction

- An Autonomous Vehicles (AV) needs to continuously evaluate the space of all possible future motions from other road agents so that it can maintain a safe and effective motion plan for itself.
- consider driving next to a sports or music venue with lots of pedestrians.
    - Autonomous driving in such environments requires a motion forecasting and planning system that is:
        - Fast
        - Scales well with the number of agents.
- The existing motion forecasting methods do not meet the requirements discussed above. Models typically take upwards of 40-50ms for inference. This scalability issue is not addressed in public benchmarks and is often ignored in publications.
- Proposed methods often use raster (render-based) input representations which require costly CNNs for processing.
- Recently, methods have been proposed that use sparse point-based input representations. These methods offer improvements in accuracy and a reduction in the number of model parameters.
    - However, with a focus on accuracy, these methods use agent-centric scene representations, which require re-encoding road points and agent points from the view point of each individual agent.
- This work introduces **StopNet**, a motion forecasting method focused on latency and scalability.
    - The authors develop a novel whole-scene sparse input representation which can encode scene inputs pertaining to all agents at once.
        - Drawing from the 3D object detection literature, we develop a PointPillars-inspired scene encoder to concurrently process sparse points

        - sampled from all agents, leading to a very fast trajectory prediction model whose latency is mostly invariant to the number of agents.
    - StopNet's whole-scene encoder also supports predicting probabilistic occupancy grids (a dense output format capturing the probability that any given grid cell in the map is occupied by some agent part).
        - This output representation allows the AV planner to reason about the occupancy of entire regions in busy scenes without a need for processing individual trajectories—thereby requiring almost constant computation.
    -

Related Work

- Agent-Centric vs. Whole-Scene Modeling
    - Agent-centric models re-encode the world from the view point of every agent in the scene.
        - This process requires transforming road state and the state of all other agents into an agent-centric frame. Therefore, these methods scale linearly with the number of agents, which poses a scalability issue in dense urban scenes with hundreds of pedestrians and vehicles.
    - A popular alternative is whole scene modeling, where the bulk of the scene encoding is done in a shared coordinate system for all agents.
        - Whole-scene modeling has the very attractive advantage that the processing time is invariant to the number of agents.
- Dense vs. Sparse Input Representation
    - whole-scene models have always used a bird's-eye view (BEV) raster input representation to encode road elements, agent state, and agent interactions. This approach allows including a variety of heterogeneous inputs into a common raster format, and enables the use of well-established powerful CNN models. However, there are several disadvantages. The model's field of view (FOV) and resolution are constrained by the computational budget, and the ability to model spatially-distant interactions is dependent on the receptive field of the network.
    - On the other hand, with sparse inputs representations, the model inputs consist of vectors of continuous state attributes encoding the agent motion history, relation to road elements, and relation to neighboring agents.
        - This allows for arbitrary long-range interactions, and infinite resolution in continuous state attributes.
        - However, sparse inputs have always been combined with agent-centric models, posing scalability issues.
        - **StopNet** is the first method to address scalability by introducing a whole-scene sparse input representation and model.
- Trajectory vs. Occupancy Output Representation
    - A common approach to capturing trajectory uncertainty is to predict multiple trajectories per agent as well as Gaussian position uncertainty for each trajectory

    waypoint, which in busy scenes, amounts to a large set of constraints to process in the planning algorithm.
  - Moreover, the per-agent trajectories may be overlapping in space, and sampling from them independently may produce samples which violate physical occupancy constraints by placing agents on top of each other.
  - An alternative output representation is to predict the collective occupancy likelihood as discretized space-time cells in a grid view of the world.

Method

- Occupancy Prediction
  - Predicting occupancy grids with spatial dimensions WxH. Each cell in the occupancy grid contains a value in the range [0,1] representing the probability that any part of any agent box overlaps with that grid cell at time t.

- Sparse Whole-Scene Input Representation
  - The model inputs consist of 3 sets of points P_r, P_l, and P_a, each associated feature vectors.
  - P = [P_r] U [P_ l] U [P_a]
  - P_r: The road element points
  - P_l: Traffic light points
  - P_a: Agent points

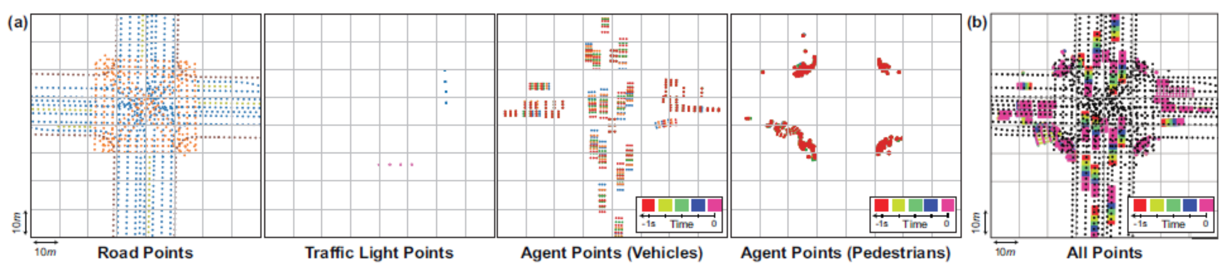

Fig. 2. Sparse Whole-Scene Input Representation. (a) Input point sets $\mathcal{P}^r$, $\mathcal{P}^l$ and $\mathcal{P}^a$ (vehicles and pedestrians) for an example scene. (b) All points.

  -

- Whole-Scene Encoder
  - StopNet consists of an encoder, a ResNet backbone, and 2 heads for decoding trajectory and occupancy predictions from the shared scene features.
    1. Inspired by PointPillars, the StopNet encoder discretizes the point set P into an evenly-spaced grid of MxN pillars in the x-y plane.
    2. A simplified version of PintNet is then applied to encode and aggregate the features from all points in each pillar.
    3. A max operation (max pooling) is then applied across all the points within each pillar to compute a single feature vector per pillar.
    4. The MxN feature map produced by the encoder is then processed through a ResNet backbone, reshaped to WxH, and concatenated with

binary occupancy grids rendered from the current positions of scene agents.

5. The resulting feature map is then shared by a trajectory decoder and an occupancy grid decoder to produce the final predictions of the model.

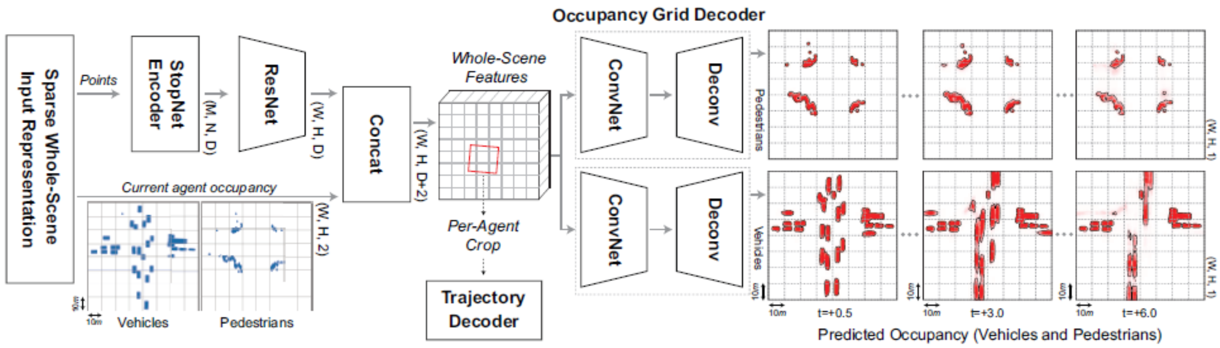

Fig. 3. An overview of the StopNet architecture. The encoder processes the input point set $\mathcal{P}$ and produces a feature map, which is used to predict both per-agent trajectories and whole-scene occupancy grids for each agent type. Input agent boxes at $t = 0$ are also rendered in BEV as binary features and fed to the trajectory and occupancy grid decoders.

Experiments

- Datasets
    - Crowds Dataset.
        - This dataset is a revision of the Waymo Open Motion Dataset focused on crowded scenes. It contains over 13 million scenarios spanning over 500 hours of real-world driving in several urban areas across the US.
        - The scenarios contain dynamic agents, traffic lights and road network information. All scenarios contain at least 20 dynamic agents.
- Training Setup
    - The authors train 3 variants of their model:
        - M_T is trained only with a trajectory loss
        - M_O is trained only with an occupancy loss
        - M_TO uses co-training and a consistency loss
- Metrics
    - Trajectory Metrics:
        - The authors use 2 standard Euclidean distance-based metrics:
            - Minimum Average Displacement Error (min ADE)
            - Minimum Final Displacement Error (min FDE)
    - Occupancy Metrics:
        - Mean Cross Entropy Error between the predicted occupancy grids and the ground-truth
- Results
    - Occupancy Grids vs. Trajectories
        - Trajectories

        - Trajectory models often output tens of potential trajectories per agent, which need to be taken into consideration as constraints in the planning algorithms.
        - The size of the trajectory outputs grows linearly with the number of agents in the scene, while the number of potential agent interactions grows quadratically.
    - Occupancy Grids
        - Occupancy grids require fixed compute to generate and consume regardless of the number of agents in the scene.
        - Occupancy grids also capture the full extents of agent bodies, as opposed to just center locations, and this simplifies calculating overlap probabilities.
        - The occupancy representation is particularly useful in busy urban scenes, where trajectory prediction models face challenges caused by noisy detection and poor tracking due to occlusions.

Conclusion

- The authors proposed StopNet, a scalable motion forecasting method that accommodates sparse inputs in a whole-scene modeling framework, and co-trains trajectory and occupancy representations.

Glossary: