Prediction For Autonomous Driving In the Wild

Xinshuo Weng, 4th year PhD student
Robotics Institute, Carnegie Mellon University

October 29, 2021
Overview of My Research
Perception and Prediction for Autonomous Driving

2D Object Detection → 2D Object Tracking

2D Object Detection
- Lee, arXiv ‘16
- Weng, WACV ‘18

3D Object Detection
- Man, ICCV ‘21
- Park, BMVC ‘21

Monocular 3D Detection
- Weng, ICCVW ‘19

Ground Plane Estimation
- Man, ACM/MM ‘19

3D Multi-Object Tracking
- Weng, CVPR ‘20
- Weng, arXiv ‘21

Wang, ICRA ‘21

Collision Avoidance
- Manglik, IROS ‘19

Weng, IROS ‘20
- Weng, ECCVW ‘20

Trajectory Forecasting
- Weng, ICRA ‘21
- Weng, ECCVW ‘20

Point Cloud Forecasting
- Ye, ICCV ‘21

Object (Re-)Identification
- Sun, IROS ‘20
- Li, WACV ‘21

Ground Plane Estimation
- Man, ACMMM ‘19

Collision Avoidance

Camera Calibration
- Xu, CVPR ‘21

Dataset
- Weng, arXiv ‘21

Facial Landmark Detection
- Dong, CVPR ‘20, TPAMI ‘20
- Weng, BMVC ‘19

Visual Lipreading
Agenda

- Trajectory prediction in autonomous driving
- Perception-Prediction Integration
  - Perception-conditioned prediction
  - Forecast-then-perception pipeline

Key Idea: Robust Perception & Prediction Stack with Seamless Integration and Uncertainty Propagation
Perception and prediction are important components in the autonomous driving stack.
Standard Perception and Prediction Pipeline

Sensor Data

3D Object Detection

3D Multi-Object Tracking

Trajectory Forecasting
Standard Perception and Prediction Pipeline

1. Sensor Data
2. 3D Object Detection
3. 3D Multi-Object Tracking
4. Trajectory Forecasting

Detection results
Standard Perception and Prediction Pipeline

Sensor Data → 3D Object Detection → 3D Multi-Object Tracking → Trajectory Forecasting

Tracking results
Standard Perception and Prediction Pipeline

Sensor Data

3D Object Detection

3D Multi-Object Tracking

Trajectory Forecasting

Forecasting results
Standard Perception and Prediction Pipeline

- Sensor Data
- 3D Object Detection
- 3D Multi-Object Tracking
- Trajectory Forecasting

Perception
Prediction
Prediction Is Studied Separately from Other Modules!

- Assumption: GT (past + future) trajectories are given for testing
- Advantages:
  - Enable development of perception-agnostic prediction algorithms
  - Faster development cycle (isolation from other tasks)
- Downside: cannot be naively applied to real-world setting, why?
  - Idealized assumption: no GT trajectories available for deployment
  - Data distribution shift: GT trajectories vs. perception outputs
A Concrete Example: Social-GAN

- Motion state (Trajectories)
- Sequential models (LSTM)
- Interaction modeling (Pooling module)
- Multi-modal predictions (GAN)

Gupta, Johnson, Fei-Fei, Savarese, Alahi. Social GAN: Socially Acceptable Trajectories with Generative Adversarial Networks. CVPR 2018
Another Concrete Example: AgentFormer

- Allowing an agent's state at one time to directly interact with another agent's state at any timestamp (future or past)

Temporal-then-Social (Standard)

Temporal and Social together (Ours)

Yuan, Weng, Ou, Kitani. AgentFormer: Agent-Aware Transformers for Socio-Temporal Multi-Agent Forecasting. ICCV 2021
Challenge of Joint Social & Temporal Modeling

Agent-Aware Transformer
(Joint Social & Temporal Modeling + Preserve Time & Agent Information)

Challenges of applying Transformer to multi-agent trajectory sequence:

- Loss of Time Information
  - Time Encoding
- Loss of Agent Identity Information
  - Agent-Aware Attention

Yuan, Weng, Ou, Kitani. AgentFormer: Agent-Aware Transformers for Socio-Temporal Multi-Agent Forecasting. ICCV 2021
Agent-Aware Attention

Agent-Aware Attention
(Joint Social & Temporal Modeling + Preserve Time & Agent Information)

Agent 1 Agent 2 Agent 3

Two sets of keys & queries:

\[ Q_{\text{self}} = QW_{\text{self}}, \quad K_{\text{self}} = KW_{\text{self}} \]
\[ Q_{\text{other}} = QW_{\text{other}}, \quad K_{\text{other}} = KW_{\text{other}} \]

Mask \( M \)

Attention Weight Matrix \( A \)

\[
A = M \odot (Q_{\text{self}}K_{\text{self}}^T) + (1 - M) \odot (Q_{\text{other}}K_{\text{other}}^T)
\]
AgentFormer: Model Overview

DLow Sampling
Diverse & Accurate Trajectories

Yuan, Weng, Ou, Kitani. AgentFormer: Agent-Aware Transformers for Socio-Temporal Multi-Agent Forecasting. ICCV 2021
AgentFormer: Visualization

Yuan, Weng, Ou, Kitani. AgentFormer: Agent-Aware Transformers for Socio-Temporal Multi-Agent Forecasting. ICCV 2021
State-of-the-art Performance on ETH/UCY/nuScenes

<table>
<thead>
<tr>
<th>Method</th>
<th>ETH</th>
<th>Hotel</th>
<th>Univ</th>
<th>Zara1</th>
<th>Zara2</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGAN [15]</td>
<td>0.81/1.52</td>
<td>0.72/1.61</td>
<td>0.60/1.26</td>
<td>0.34/0.69</td>
<td>0.42/0.84</td>
<td>0.58/1.18</td>
</tr>
<tr>
<td>Sophie [44]</td>
<td>0.70/1.43</td>
<td>0.76/1.67</td>
<td>0.54/1.24</td>
<td>0.30/0.63</td>
<td>0.38/0.78</td>
<td>0.54/1.15</td>
</tr>
<tr>
<td>Transformer-TF [12]</td>
<td>0.61/1.12</td>
<td>0.18/0.30</td>
<td>0.35/0.65</td>
<td>0.22/0.38</td>
<td>0.17/0.32</td>
<td>0.31/0.55</td>
</tr>
<tr>
<td>STAR [55]</td>
<td>0.36/0.65</td>
<td>0.17/0.36</td>
<td>0.31/0.62</td>
<td>0.26/0.55</td>
<td>0.22/0.46</td>
<td>0.26/0.53</td>
</tr>
<tr>
<td>PECNet [34]</td>
<td>0.54/0.87</td>
<td>0.18/0.24</td>
<td>0.35/0.60</td>
<td>0.22/0.39</td>
<td>0.17/0.30</td>
<td>0.29/0.48</td>
</tr>
<tr>
<td>Trajectron++ [45]</td>
<td>0.39/0.83</td>
<td>0.12/0.21</td>
<td>0.20/0.44</td>
<td>0.15/0.33</td>
<td>0.11/0.25</td>
<td>0.19/0.41</td>
</tr>
<tr>
<td>Ours (AgentFormer)</td>
<td><strong>0.26/0.39</strong></td>
<td><strong>0.11/0.14</strong></td>
<td><strong>0.26/0.46</strong></td>
<td><strong>0.15/0.23</strong></td>
<td><strong>0.14/0.24</strong></td>
<td><strong>0.18/0.29</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>$K = 5$ Samples</th>
<th>$K = 10$ Samples</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ADE$_5$ ↓</td>
<td>FDE$_5$ ↓</td>
</tr>
<tr>
<td>MTP [8]</td>
<td>2.93</td>
<td>-</td>
</tr>
<tr>
<td>MultiPath [5]</td>
<td>2.32</td>
<td>-</td>
</tr>
<tr>
<td>CoverNet [39]</td>
<td>1.96</td>
<td>-</td>
</tr>
<tr>
<td>DSF-AF [33]</td>
<td>2.06</td>
<td>4.67</td>
</tr>
<tr>
<td>DLow-AF [59]</td>
<td>2.11</td>
<td>4.70</td>
</tr>
<tr>
<td>Trajectron++ [45]</td>
<td>1.88</td>
<td>-</td>
</tr>
<tr>
<td>Ours (AgentFormer)</td>
<td><strong>1.59</strong></td>
<td><strong>3.14</strong></td>
</tr>
</tbody>
</table>

ETH/UCY

nuScenes

Code is on GitHub!
https://github.com/Khrylx/AgentFormer
Agenda

• Trajectory prediction in autonomous driving
• Perception-Prediction Integration
  • Perception-conditioned prediction
How about the data distribution shift?
GT past trajectories vs. perception outputs

How Big is the impact to prediction?
Prediction in the Wild: An Illustrative Example

- Key differences: Using tracking results as inputs
- Perception errors: Identity switches, missing tracking, noisy tracking, false positive tracks, etc
Prediction in the Wild: Real-World Results (SDD)

- Missing predictions, noisy & inconsistent predictions, ID-switched predictions
- Predictions are easily affected by the most immediate relative motion

Rui Yu and Zihan Zhou. Towards Robust Human Trajectory Prediction in Raw Videos. arXiv 2021
Similar Observations in Driving Data (nuScenes)

- Predictions can have wrong orientation due to noisy tracking

Using GT traj. as inputs

Using tracklets as inputs

Similar Observations in Driving Data (KITTI)

- Inconsistent predictions (sudden velocity & orientation change) caused by identity switches

Using GT past trajectories as inputs

Using past tracklets from tracking as inputs

Tracking Errors Affect Prediction (Quantitative Measurement)

- If using past GT trajectories as inputs, prediction errors are low
- If using past tracklets from tracking as inputs, prediction errors are large (28x)

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Eval. Targets (# of obj)</th>
<th>Inputs to Prediction</th>
<th>$\text{minADE}_k$</th>
<th>$\text{minFDE}_k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>KITTI</td>
<td>Objects with IDS (33)</td>
<td>GT past trajectories</td>
<td>0.100</td>
<td>0.171</td>
</tr>
<tr>
<td></td>
<td>Objects with IDS (33)</td>
<td>past tracklets</td>
<td>2.820</td>
<td>4.514</td>
</tr>
<tr>
<td></td>
<td>Objects with FRAG (330)</td>
<td>GT past trajectories</td>
<td>0.177</td>
<td>0.306</td>
</tr>
<tr>
<td></td>
<td>Objects with FRAG (330)</td>
<td>past tracklets</td>
<td>1.621</td>
<td>2.155</td>
</tr>
<tr>
<td>nuScenes</td>
<td>Objects with IDS (4160)</td>
<td>GT past trajectories</td>
<td>0.473</td>
<td>0.825</td>
</tr>
<tr>
<td></td>
<td>Objects with IDS (4160)</td>
<td>past tracklets</td>
<td>8.345</td>
<td>13.892</td>
</tr>
<tr>
<td></td>
<td>Objects with FRAG (3365)</td>
<td>GT past trajectories</td>
<td>0.621</td>
<td>1.108</td>
</tr>
<tr>
<td></td>
<td>Objects with FRAG (3365)</td>
<td>past tracklets</td>
<td>14.520</td>
<td>21.815</td>
</tr>
</tbody>
</table>

Tracking Error Frequency & Spatial Distribution

- Non-negligible number of tracking errors
- About 200 errors are <5 meters, at a distance where planning is sensitive to
- Can cause safety-critical issue

How can we deal with this data distribution shift?

How can we fix this error propagation issue?
From “Track-then-Predict” to “Track-and-Predict”

- Prediction does not explicitly take into tracking results as inputs
- Any issue?

What about prediction in the next window?
Multi-hypothesis Tracking and Prediction (MTP)

- How to deal with error propagation in the sequence level?
- If one set of tracklets is not robust, what about more sets?
- Multi-hypothesis data association (MHDA)
- Cannot exhaustively search for all combinations of matching (combinatorial explosion)
  - Murty’s K-best assignment method

---

Multi-hypothesis Tracking and Prediction (MTP)

- Core idea: use MHDA to increase the likelihood of preserving accurate matching
- Assumption: different hypothesis is complimentary
Multi-hypothesis Tracking and Prediction (MTP)

- Different hypothesis is indeed complimentary
  - The No. of tracking errors persistent in all hypotheses are reduced with more hypotheses
- Significantly improve the accuracy metrics: minADE/FDE

4x of improvements with MTP

### TABLE II

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Targets</th>
<th>Methods</th>
<th>minADE&lt;sub&gt;k&lt;/sub&gt;</th>
<th>minFDE&lt;sub&gt;k&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>KITTI</td>
<td>IDS</td>
<td>STP, H=1, k=20</td>
<td>2.820</td>
<td>4.514</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MTP (Ours), H=5, k=100</td>
<td>1.099</td>
<td>1.768</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MTP (Ours), H=10, k=200</td>
<td>0.844</td>
<td>1.332</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MTP (Ours), H=20, k=400</td>
<td>0.707</td>
<td>1.093</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MTP (Ours), H=5, k=20, sampling</td>
<td>1.118</td>
<td>1.802</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MTP (Ours), H=10, k=20, sampling</td>
<td>0.876</td>
<td>1.390</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MTP (Ours), H=20, k=20, sampling</td>
<td>0.747</td>
<td>1.173</td>
</tr>
</tbody>
</table>

| KITTI    | FRAG    | STP, H=1, k=20    | 1.621             | 2.155              |
|          |         | MTP (Ours), H=5, k=100 | 1.436             | 1.862              |
|          |         | MTP (Ours), H=10, k=200 | 1.385             | 1.765              |
|          |         | MTP (Ours), H=20, k=400 | 1.305             | 1.627              |
|          |         | MTP (Ours), H=5, k=20, sampling | 1.448             | 1.888              |
|          |         | MTP (Ours), H=10, k=20, sampling | 1.404             | 1.801              |
|          |         | MTP (Ours), H=20, k=20, sampling | 1.335             | 1.688              |

MTP Increases the Likelihood of Preserving Accurate Predictions

Separated perception and prediction
Inaccurate predictions

MTP (Ours)
More accurate predictions

Agenda

• Trajectory prediction in autonomous driving
• Perception-Prediction Integration
  • Perception-conditioned prediction
  • Forecast-then-perception pipeline
SPF$^2$: Sequential Pointcloud Forecasting for Sequential Pose Forecasting

- Track-then-forecast pipeline:
  - Detection $\rightarrow$ MOT $\rightarrow$ Trajectory Forecasting

- Forecast-then-track pipeline
  - Sequential Pointcloud Forecasting $\rightarrow$ Detection $\rightarrow$ MOT

- Differences
  - Invert the order of forecasting
  - Forecast at the sensor level (not object positions)

Weng et al. Inverting the Pose Forecasting Pipeline with SPF$^2$: Sequential Pointcloud Forecasting for Sequential Pose Forecasting. CoRL 2020
SPF$^2$ Provides Stable Object Trajectory Prediction

- Predicted point clouds preserve object information
- Equivalent to results obtained from standard pipeline, i.e., object trajectories
Performance Scaling with More Point Cloud Data

• Advantages?
  • Does not require human annotation for prediction
  • Point cloud data is prevalent nowadays
• The key is this new task -- Sequential Pointcloud Forecasting (SPF)

Weng et al. Inverting the Pose Forecasting Pipeline with SPF2: Sequential Pointcloud Forecasting for Sequential Pose Forecasting. CoRL 2020
Performance Scaling with More Point Cloud Data

- CD: Chamfer distance (lower is better)
- EMD: Earth Mover’s Distance (lower is better)
SPF: Sequential Pointcloud Forecasting

• Goal: a sequence of past clouds -> a sequence of future clouds
• Predict the entire scene, including background
• Deal with large-scale points (1.5M) rather than 1k points

Weng et al. Inverting the Pose Forecasting Pipeline with SPF2: Sequential Pointcloud Forecasting for Sequential Pose Forecasting. CoRL 2020
SPFNet

• How to learn representations from large-scale point cloud sequences?
• Baseline: FC-LSTM autoencoder model
• Four modules
  • Frame-wise point cloud encoder
  • FC-LSTM for temporal modeling
  • Frame-wise point cloud decoder
  • Losses: chamfer distance

Weng et al. Inverting the Pose Forecasting Pipeline with SPF2: Sequential Pointcloud Forecasting for Sequential Pose Forecasting. CoRL 2020
Predicted Point Clouds Preserves Objects

Weng et al. Inverting the Pose Forecasting Pipeline with SPF2: Sequential Pointcloud Forecasting for Sequential Pose Forecasting. CoRL 2020
Detection on Predicted Point Clouds

- Green: detected 3D boxes
- Yellow: GT 3D boxes
- Detections mostly match with GT on predicted point clouds from SPFNet
Tracking on Predicted Point Clouds

• Color represents the object identity
• Predicted point clouds from SPFNet preserves objects

Weng et al. Inverting the Pose Forecasting Pipeline with SPF2: Sequential Pointcloud Forecasting for Sequential Pose Forecasting. CoRL 2020
Takeaway

Joint module design is the key to robustness

Perception

Prediction
Open Problem
Prediction-Conditioned Planning & Planning-Informed Prediction

- How to efficiently leverage multi-modal predictions in contingency planning
- Robust planning: also leverage the probability of predictions
- Planning-aware prediction: not all predictions should be treated equal
- Ego-conditioned prediction
Presented works (in presented order)

- Ye Yuan, Xinshuo Weng, Yanglan Ou, Kris Kitani. AgentFormer: Agent-Aware Transformers for Socio-Temporal Multi-Agent Forecasting. ICCV 2021

- Xinshuo Weng*, Ye Yuan*, Kris Kitani. PTP: Parallelized Tracking and Prediction With Graph Neural Networks and Diversity Sampling. ICRA / RA-L 2021

- Xinshuo Weng, Boris Ivanovic, Marco Pavone. MTP: Multi-hypothesis Tracking and Prediction for Reduced Error Propagation. arXiv 2021

- Xinshuo Weng, Jianren Wang, Sergey Levine, Kris Kitani, Nick Rhinehart. Inverting the Pose Forecasting Pipeline with SPF2: Sequential Pointcloud Forecasting for Sequential Pose Forecasting. CoRL 2020
Interested in Autonomous Driving Workshop?

• (December 13) Workshop at NeurIPS ’21
• Interested in participating in / organizing?

Past workshop events this year:
IROS’ 21, IJCAI’ 21, ICCV’21
Prediction For Autonomous Driving In the Wild

Xinshuo Weng, 4th year PhD student
Robotics Institute, Carnegie Mellon University

October 29, 2021