SDVTracker
Real-Time Multi-Sensor Association and Tracking for Self-Driving Vehicles

Shivam Gautam, Gregory P. Meyer, Carlos Vallespi-Gonzalez and Brian C. Becker
Perception System

Object Detection ➔ Detections $D_t$ ➔ Objects $O_{t-1}$ ➔ Scoring (IoU/ L2/ Mahal distance) ➔ Assignment ➔ Pairwise Association ➔ State Update (EKF/ UKF) ➔ Objects $O_t$ ➔ Trajectory Prediction / Motion Planning

Object Tracking
Perception System
Perception System

Object Detection

Detections $D_t$

Objects $O_{t-1}$

Object Tracking
Perception System

Detections $D_t$

Objects $O_{t-1}$

Scoring (IoU/ L2/ Mahal distance)

Pairwise Association

Object Tracking
Perception System

Object Detection → Object Detection $D_t$ → Scoring (IoU, L2, Mahal distance) → Assignment → Pairwise Association → Object Tracking
Perception System

Object Detection

Objects $O_{t-1}$

Detections $D_t$

Scoring (IoU/ L2/ Mahal distance)

Assignment

Pairwise Association

State Update (EKF/ UKF)

Object Tracking
Perception System

Object Detection → Detections $D_t$ → Scoring (IoU/ L2/ Mahal distance) → Assignment → State Update (EKF/ UKF) → Objects $O_t$
Perception System

Object Detection

Detections $D_t$

Objects $O_{t-1}$

Scoring (IoU/ L2/ Mahal distance)

Assignment

State Update (EKF/ UKF)

Objects $O_t$

Pairwise Association

Object Tracking

Trajectory Prediction / Motion Planning
Why Robust Association and Tracking?
Typical Urban Scene
Typical Urban Scene - Dense
Typical Urban Scene - Dense

How many actors? Pedestrian? Bike?
Typical Urban Scene - Dense

2 Pedestrians, 1 Bike, Bike Actor not visible
Typical Urban Scene - Occlusions

*Not all occlusions are shown*
Typical Urban Scene - Motion Diversity
Proposed Method
SDVTracker: Real-Time Multi-Sensor Association and Tracking for Self-Driving Vehicles
SDVTracker: Real-Time Multi-Sensor Association and Tracking for Self-Driving Vehicles

**Learned Method**: Uses an LSTM network for actor-level association and tracking.
SDVTracker: Real-Time Multi-Sensor Association and Tracking for Self-Driving Vehicles

**Learned Method**: Uses an LSTM network for actor-level association and tracking.

**Real-Time**: Runs under 5ms (CPU) / 3ms (GPU) for 500 Actors.
SDVTracker: Real-Time Multi-Sensor Association and Tracking for Self-Driving Vehicles

**Learned Method**: Uses an LSTM network for actor-level association and tracking.

**Real-Time**: Runs under 5ms (CPU)/ 3ms (GPU) for 500 Actors.

**Multi-Sensor**: Associates cross-modality detections (Lidar, Camera) to a single tracked object.
SDVTracker: Real-Time Multi-Sensor Association and Tracking for Self-Driving Vehicles

Learned Method: Uses an LSTM network for actor-level association and tracking.

Real-Time: Runs under 5ms (CPU)/ 3ms (GPU) for 500 Actors.

Multi-Sensor: Associates cross-modality detections (Lidar, Camera) to a single tracked object.

Joint Association and Tracking: Jointly learns association and state estimation in a single model for peds/bikes/skateboarders.
Why robust Association and Tracking?

- **Association is hard**: occlusions, dense crowds, varying motions and detector false positives or false negatives.
Why robust Association and Tracking?

- **Association is hard**: occlusions, dense crowds, varying motions and detector false positives or false negatives.
- Association failures lead to **inaccurate state estimates**.
• Association failures lead to **inaccurate state estimates**.
• Association failures lead to **inaccurate state estimates**.
• Association failures lead to **inaccurate state estimates**.

![Diagram showing association failures](image)

- T=0
- T=1
- T=2
Association failures lead to **inaccurate state estimates**.
• Association failures lead to **inaccurate state estimates**.
Why robust Association and Tracking?

- **Association is hard**: occlusions, dense crowds, varying motions and detector false positives or false negatives.
- Association failures lead to **inaccurate state estimates**.
- **Cascading effects** on downstream predictions, planning.
Why robust Association and Tracking?

- **Association is hard**: occlusions, dense crowds, varying motions and detector false positives or false negatives.
- Association failures lead to **inaccurate state estimates**.
- **Cascading effects** on downstream predictions, planning.
- Mis-associations break **single-source assumptions in probabilistic filtering** based methods.
Why robust Association and Tracking?

- **Association is hard**: occlusions, dense crowds, varying motions and detector false positives or false negatives.
- Association failures lead to **inaccurate state estimates**.
- **Cascading effects** on downstream predictions, planning.
- Mis-associations break **single-source assumptions in probabilistic filtering** based methods.
- Multiple Detectors: **Multiple False positives** present making associations hard.
Why robust Association and Tracking?

- **Association is hard**: occlusions, dense crowds, varying motions and detector false positives or false negatives.
- Association failures lead to **inaccurate state estimates**.
- **Cascading effects** on downstream predictions, planning.
  - Mis-associations break **single-source assumptions in probabilistic filtering** based methods.
  - Multiple Detectors: **Multiple False positives** present making associations hard.
  - Joint Tracking of VRUs: Need robust association to account for **different motion models** across pedestrians, bikes.
SDVTracker: Real-Time Multi-Sensor Association and Tracking for Self-Driving Vehicles
Previous Work

1. State Estimation and Tracking
   a. Filter-based tracking methods utilizing EKF/UKF are most common.
   b. IMM - Interacting Multiple Models utilizes multiple filters with unique motion models.
Previous Work

1. **State Estimation and Tracking**
   a. Filter-based tracking methods utilizing EKF/UKF are most common.
   b. IMM - Interacting Multiple Models utilizes multiple filters with unique motion models.

2. **Classical Association Methods:**
   a. **IoU score**: Thresholding on amount of overlap.
   b. **L2 score**: Thresholding on euclidean distance between observed detection and predicted position.
   c. **Mahalanobis score**: Covariance weighted distance between detection and object.
Previous Work

1. State Estimation and Tracking
   a. Filter-based tracking methods utilizing EKF/UKF are most common.
   b. IMM - Interacting Multiple Models utilizes multiple filters with unique motion models.

2. Classical Association Methods:
   a. IoU score: Thresholding on amount of overlap.
   b. L2 score: Thresholding on euclidean distance between observed detection and predicted position.
   c. Mahalanobis score: Covariance weighted distance between detection and object.

3. Other Learned Methods:
   a. Most learned methods focus on 2D tracking in Image space.
   b. Previous RNN Based methods require fixed number of tracks that are known beforehand.
   c. 3D Based Learned methods:
      i. Employ expensive feature extraction networks to perform association in 3D.
      ii. Are not multi-model in terms of detections from different sources.
      iii. Do not jointly learn association and tracking.
System Overview
System Overview

Detections $D_t$

Objects $O_{t-1}$
System Overview

Detections $D_t$

Objects $O_{t-1}$

LSTM Predict

Pairwise Association
System Overview

**Detections** $D_t$

**Objects** $O_{t-1}$

**LSTM Predict**

**Greedy Assignment**

**Pairwise Association**
System Overview

- Detections $D_t$
- Objects $O_{t-1}$
- LSTM Predict
- Greedy Assignment
- Pairwise Association
- IMM State Update
- State Update
System Overview

Detections $D_t$

Objects $O_{t-1}$

LSTM Predict

Greedy Assignment

Pairwise Association

IMM State Update

Output

Objects $O_t$
Model Details
Model Overview
Model Architecture
Model Targets

Association

State Estimate

\[
\begin{array}{c}
\text{Probability} \\
\begin{array}{c}
p_{\text{score}} \\
y_{\text{score}}
\end{array} \\
\begin{array}{cccc}
x_t & y_t & v_t^x & v_t^y \\
\sigma_x & \sigma_y & \sigma_v^x & \sigma_v^y
\end{array}
\end{array}
\]
Model Targets

Association

a. $p_{\text{score}}$: Probability that the current detection-object pair is a true association.

b. $y_{\text{score}}$: Learned score quantifying how good the association is.
Model Targets

Association

a. $p_{\text{score}}$ : Probability that the current detection-object pair is a true association.

b. $y_{\text{score}}$ : Learned score quantifying how good the association is.

Why Break it down this way:

1. Easy to remove highly unlikely associations: All matches below a certain probability can be removed.

2. Removes the need for arbitrary thresholds on scores: How do you threshold a score that works for peds, bikes in all scenarios?

3. Allows in identifying multiple false positives on the same detection.
Model Targets

Association

a. \( p_{\text{score}} \): Probability that the current detection-object pair is a true association.

b. \( y_{\text{score}} \): Learned score quantifying how good the association is.

State Estimate

a. \([x_t, y_t, v_x t, v_y t]\): Position and velocity in cartesian coordinates.

b. \([\sigma_{x t}, \sigma_{y t}, \sigma_{v x t}, \sigma_{v y t}]\): Corresponding uncertainty.
Model Loss

\[ l_{total} = l_{assoc} + w_{state} \cdot l_{state}, \]

Association

\[ l_{assoc} = l_{prob} + w_{score} \cdot l_{score}, \]

a. Cross Entropy loss on probability
b. L2 Loss on Score

State Estimate

\[ l_{state} = \sum_i \left( \frac{(s_{t,i} - s^{*}_{t,i})^2}{2\sigma_{t,i}^2} + \log \sigma_{t,i} \right) \]

\[ s_t = [x_t, y_t, v_t^x, v_t^y] \]

\[ \sigma_t = [\sigma_{x_t}, \sigma_{y_t}, \sigma_{v_t^x}, \sigma_{v_t^y}] \]
Results
Results

New Proposed Metrics: MOTVE & MOTVO

Significant Improvements
- MOTVE: 16% improvement over next best method.
- ID Switches: 6.23% improvement over next best method.

Lower MOTP ≠ Lower Velocity Error
- Trajectory Prediction more reliant on future states!
## Results

### TABLE I

**Comparison of Tracking Methods Across Multiple Sensor Modalities**

<table>
<thead>
<tr>
<th>Sensing Modalities</th>
<th>Method</th>
<th>MOTA ↑</th>
<th>MOTVO ↓</th>
<th>MOTVE ↓</th>
<th>FP ↓</th>
<th>FN ↓</th>
<th>IDSW ↓</th>
<th>MOTP ↓</th>
<th>MT ↑</th>
<th>ML ↓</th>
<th>Frag ↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>LiDAR + Camera</td>
<td>IoU-based Association</td>
<td>66.5809</td>
<td>4.236</td>
<td>2.549</td>
<td>0.192</td>
<td>0.295</td>
<td>416723</td>
<td>503642</td>
<td>51731</td>
<td>0.3586</td>
<td>0.384</td>
</tr>
<tr>
<td></td>
<td>L2 Association</td>
<td>68.1027</td>
<td>3.303</td>
<td>2.334</td>
<td>0.162</td>
<td>0.294</td>
<td>386467</td>
<td>497147</td>
<td>43991</td>
<td>0.3554</td>
<td><strong>0.388</strong></td>
</tr>
<tr>
<td></td>
<td>Mahalanobis Association</td>
<td>68.6031</td>
<td>3.565</td>
<td>2.118</td>
<td>0.160</td>
<td>0.287</td>
<td>366913</td>
<td>504202</td>
<td>39521</td>
<td>0.3498</td>
<td>0.385</td>
</tr>
<tr>
<td></td>
<td>SDVTTracker (Ours)</td>
<td><strong>69.4405</strong></td>
<td><strong>2.204</strong></td>
<td><strong>1.827</strong></td>
<td><strong>0.133</strong></td>
<td><strong>0.268</strong></td>
<td><strong>346651</strong></td>
<td><strong>504744</strong></td>
<td><strong>33118</strong></td>
<td><strong>0.3485</strong></td>
<td><strong>0.388</strong></td>
</tr>
</tbody>
</table>

Better at incorporating multiple sensor modalities

- **MOTVO**: 17% improvement over next best method.
- **ID Switches**: 16% improvement over next best method.
### Ablation Studies

- Jointly learning association and state targets improves performance.
- Recurrent network outperforms MLP.
- Adding multiple model filter after LSTM improves performance slightly.

#### Table II

**Effect of Learning Joint Tracking and Association**

<table>
<thead>
<tr>
<th>Network</th>
<th>IMM</th>
<th>Learning State</th>
<th>MOTA $\uparrow$</th>
<th>MOTVO $\downarrow$</th>
<th>MOTVE $\downarrow$</th>
<th>IDSW $\downarrow$</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLP</td>
<td>✓</td>
<td>✓</td>
<td>69.2221</td>
<td>2.448</td>
<td>0.1446</td>
<td>37594</td>
</tr>
<tr>
<td>MLP</td>
<td>✓</td>
<td>✓</td>
<td>69.3863</td>
<td>2.385</td>
<td>0.1413</td>
<td>34698</td>
</tr>
<tr>
<td>LSTM</td>
<td>✓</td>
<td>✓</td>
<td>69.2877</td>
<td>2.428</td>
<td>0.1419</td>
<td>35862</td>
</tr>
<tr>
<td>LSTM</td>
<td>✓</td>
<td>✓</td>
<td>69.3971</td>
<td><strong>2.240</strong></td>
<td>0.1528</td>
<td>34031</td>
</tr>
<tr>
<td>LSTM</td>
<td>✓</td>
<td>✓</td>
<td><strong>69.4405</strong></td>
<td>2.292</td>
<td><strong>0.1393</strong></td>
<td><strong>33118</strong></td>
</tr>
</tbody>
</table>
Ablation on Joint Learning and Targets

**TABLE III**

**EFFECT OF LEARNING PROBABILITY AND SCORE**

<table>
<thead>
<tr>
<th>Association Output</th>
<th>MOTA ↑</th>
<th>MOTVO ↓</th>
<th>MOTVE ↓</th>
<th>IDSW ↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probability Only</td>
<td>69.1837</td>
<td>2.544</td>
<td>0.1466</td>
<td>35419</td>
</tr>
<tr>
<td>Score Only</td>
<td>69.3618</td>
<td>2.551</td>
<td>0.1448</td>
<td>39461</td>
</tr>
<tr>
<td>Probability and Score</td>
<td><strong>69.4405</strong></td>
<td><strong>2.292</strong></td>
<td><strong>0.1393</strong></td>
<td><strong>33118</strong></td>
</tr>
</tbody>
</table>

- Learning Probability and Score is better than learning either one.
Impact of Pedestrian Density

45% Less velocity outliers in most crowded scenes
Impact of Pedestrian Density

Faster to run on CPU for a scene with <100 actors!
1. Discussed the **challenges for object association and tracking** in urban scenarios.

2. **Presented SDVTracker**, a learned method for object detection and tracking for autonomous driving.
   a. **Joint Association and Tracking** within single model.
   b. **Multi-Sensor** (LiDAR/ Camera) tracking.
   c. **Novel targets** for learning association.

3. Demonstrated the effectiveness of the model in **crowded scenarios**.

4. Justified **real-time performance on both CPU and GPU**.

5. **Future**:
   a. Experiments with increased capacity.
   b. Multi-hypothesis tracking
   c. Feature Descriptors from Detectors.
Thank You!

Questions: Shivam Gautam
           Autonomy Engineer, Uber ATG
           sgautam@uber.com