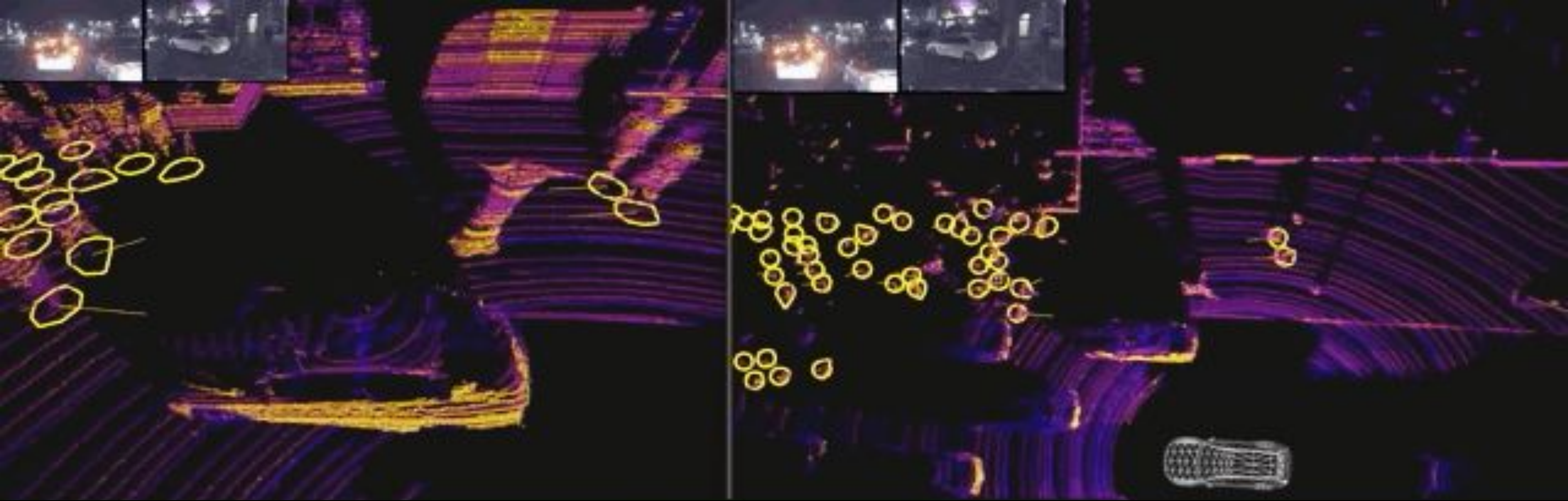


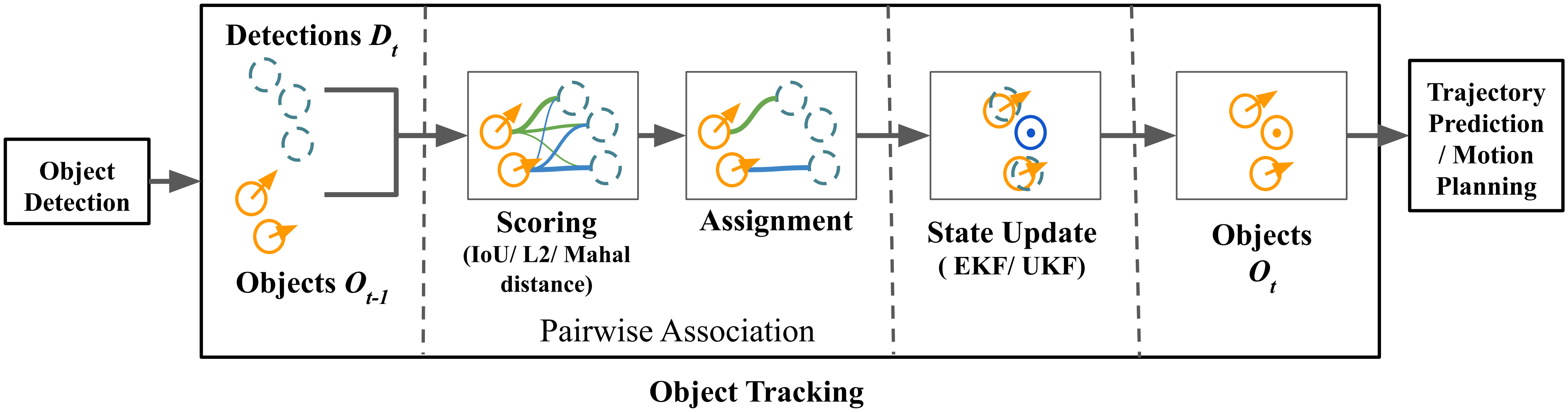
SDVTracker

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

Uber ATG

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

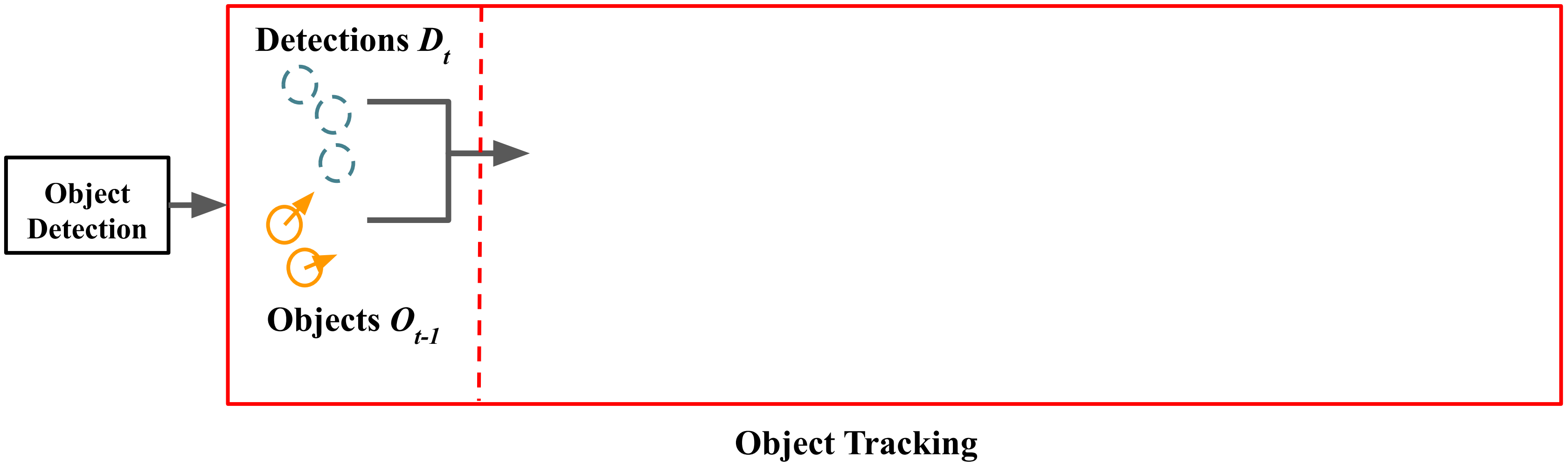
Perception System



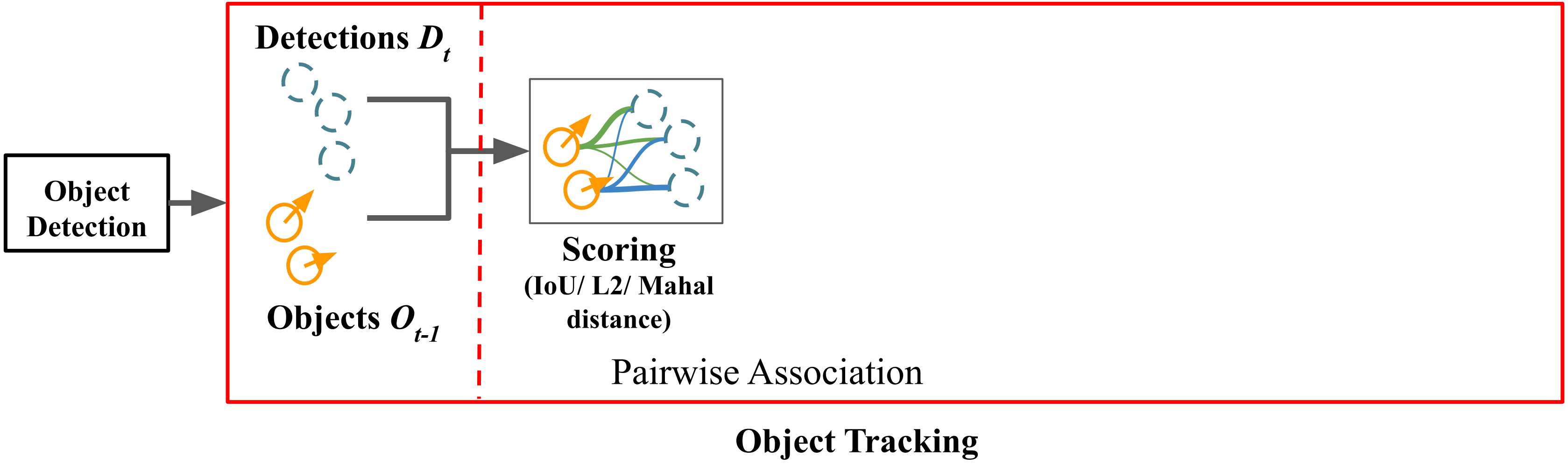
Perception System

Object Detection

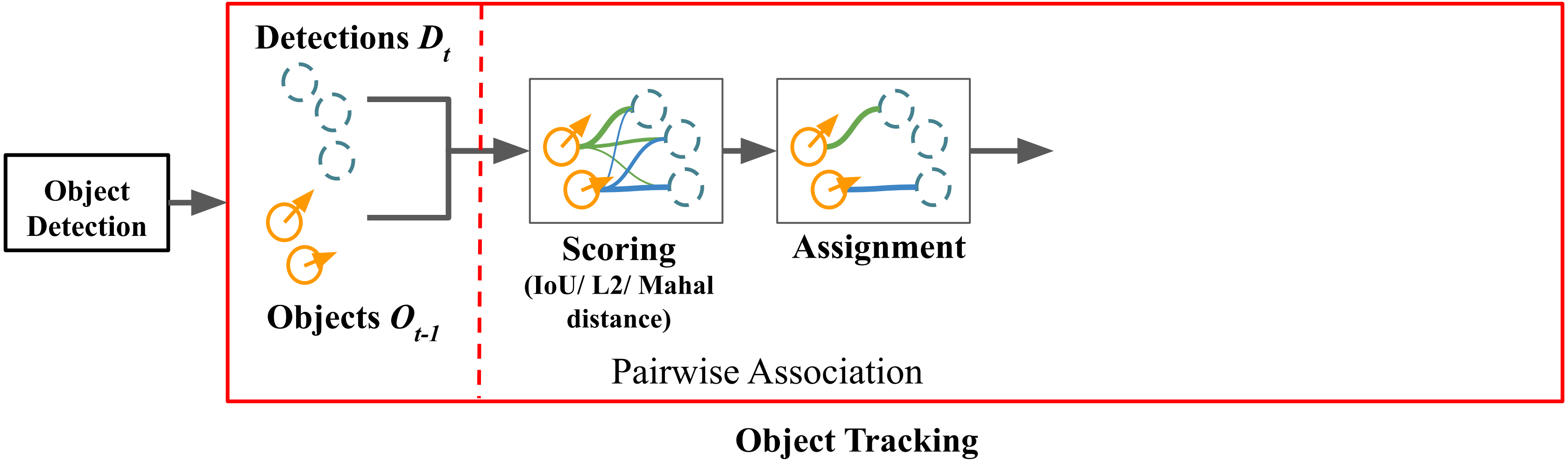
Perception System



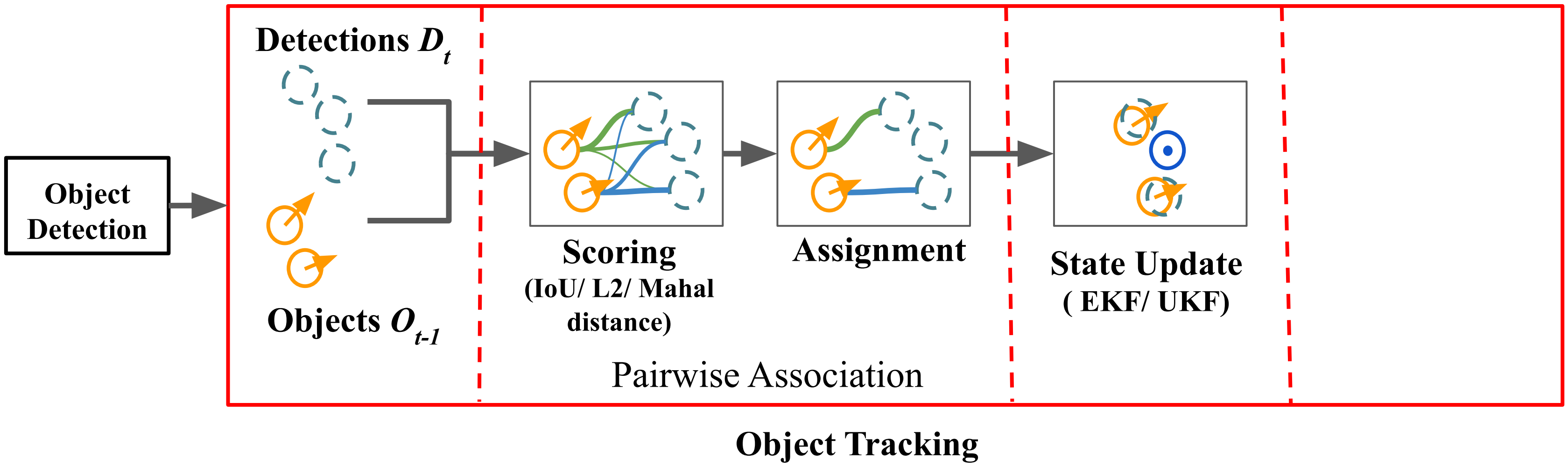
Perception System



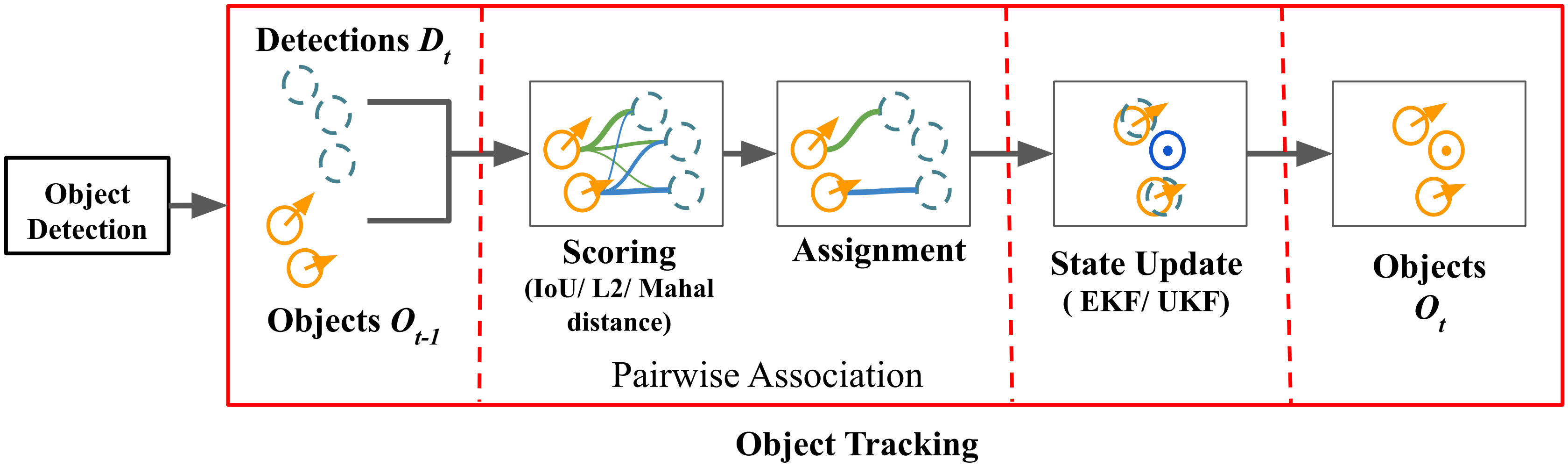
Perception System



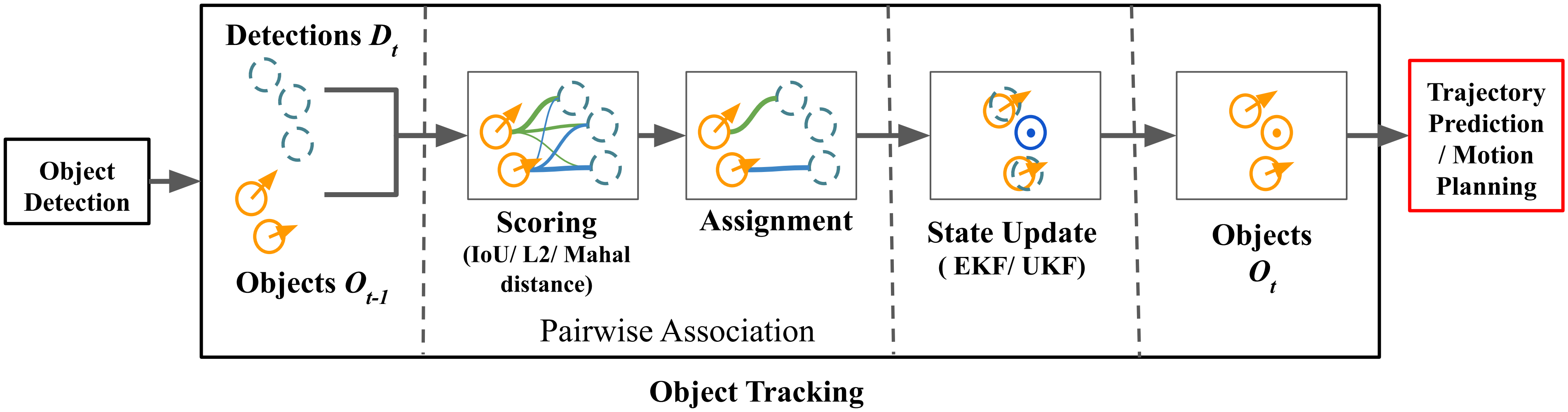
Perception System



Perception System



Perception System



Why Robust
Association and Tracking?

Typical Urban Scene

Typical Urban Scene - Dense

Typical Urban Scene - Dense

How many actors? Pedestrian? Bike ?

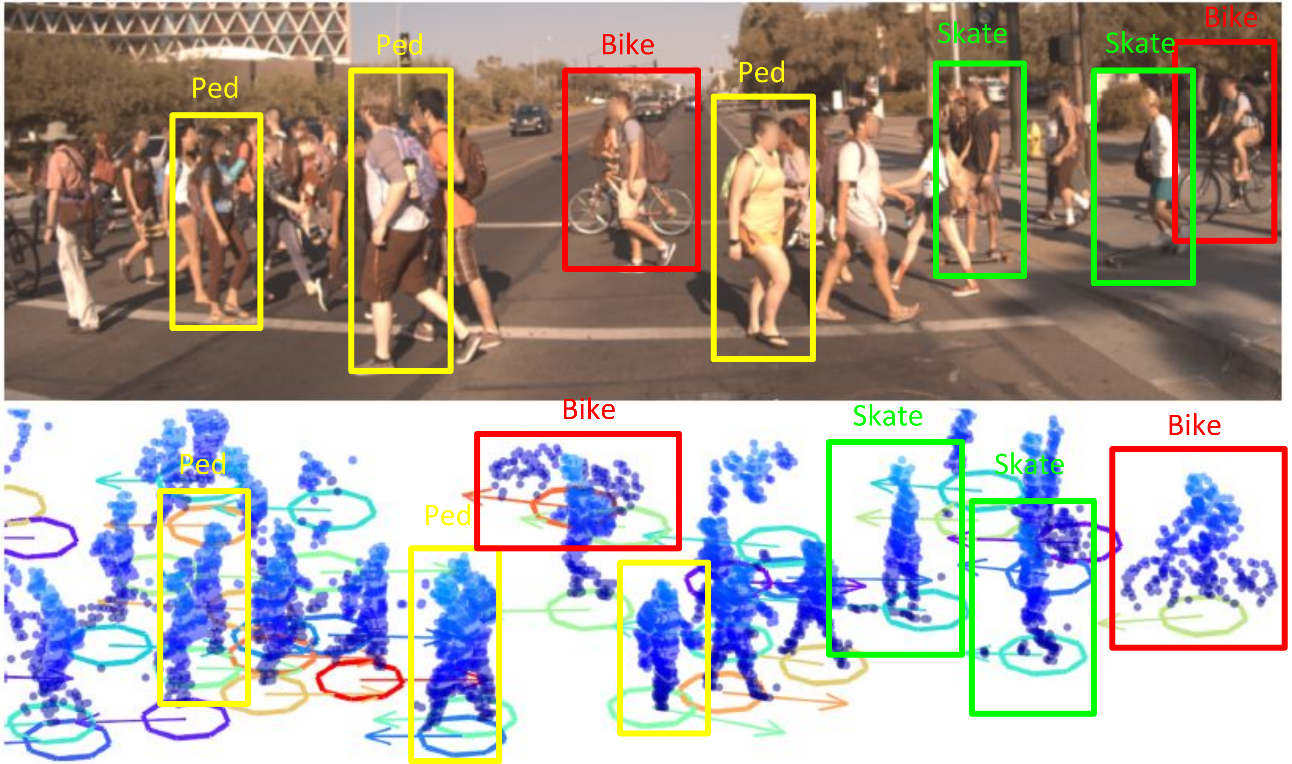
Typical Urban Scene - Dense

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.

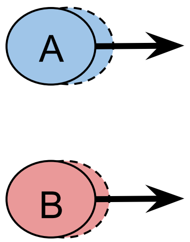
Why robust Association and Tracking?

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

Why robust Association and Tracking?

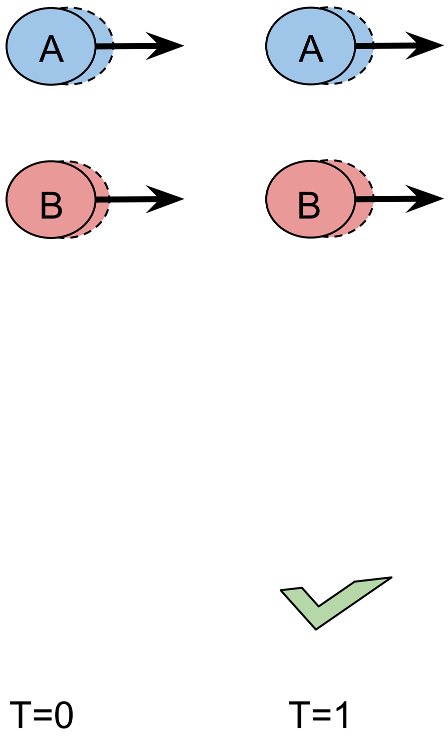
- **Association is hard** : occlusions, dense crowds, varying motions and detector false positives or false negatives.
- Association failures lead to **inaccurate state estimates**.

- Association failures lead to **inaccurate state estimates**.

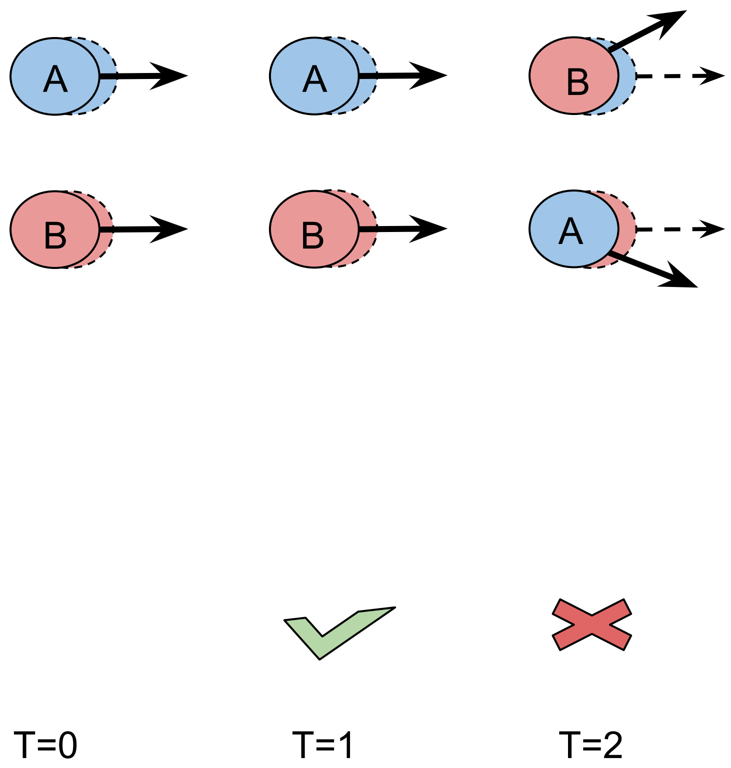


T=0

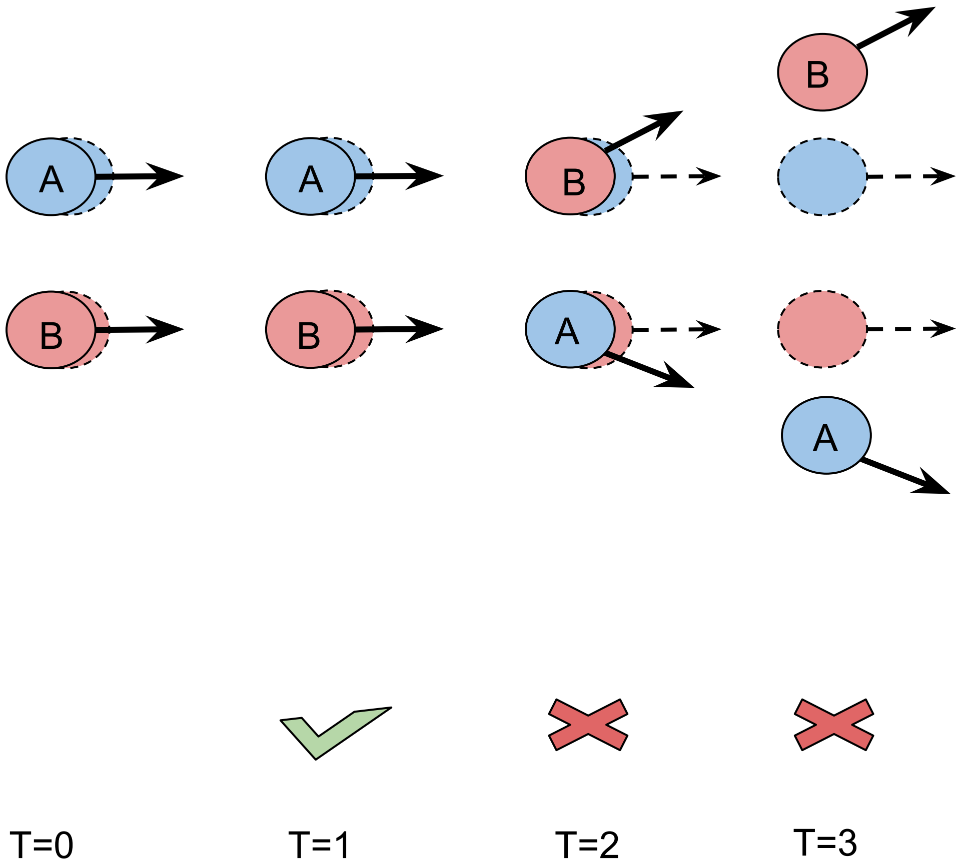
• Association failures lead to **inaccurate state estimates**.



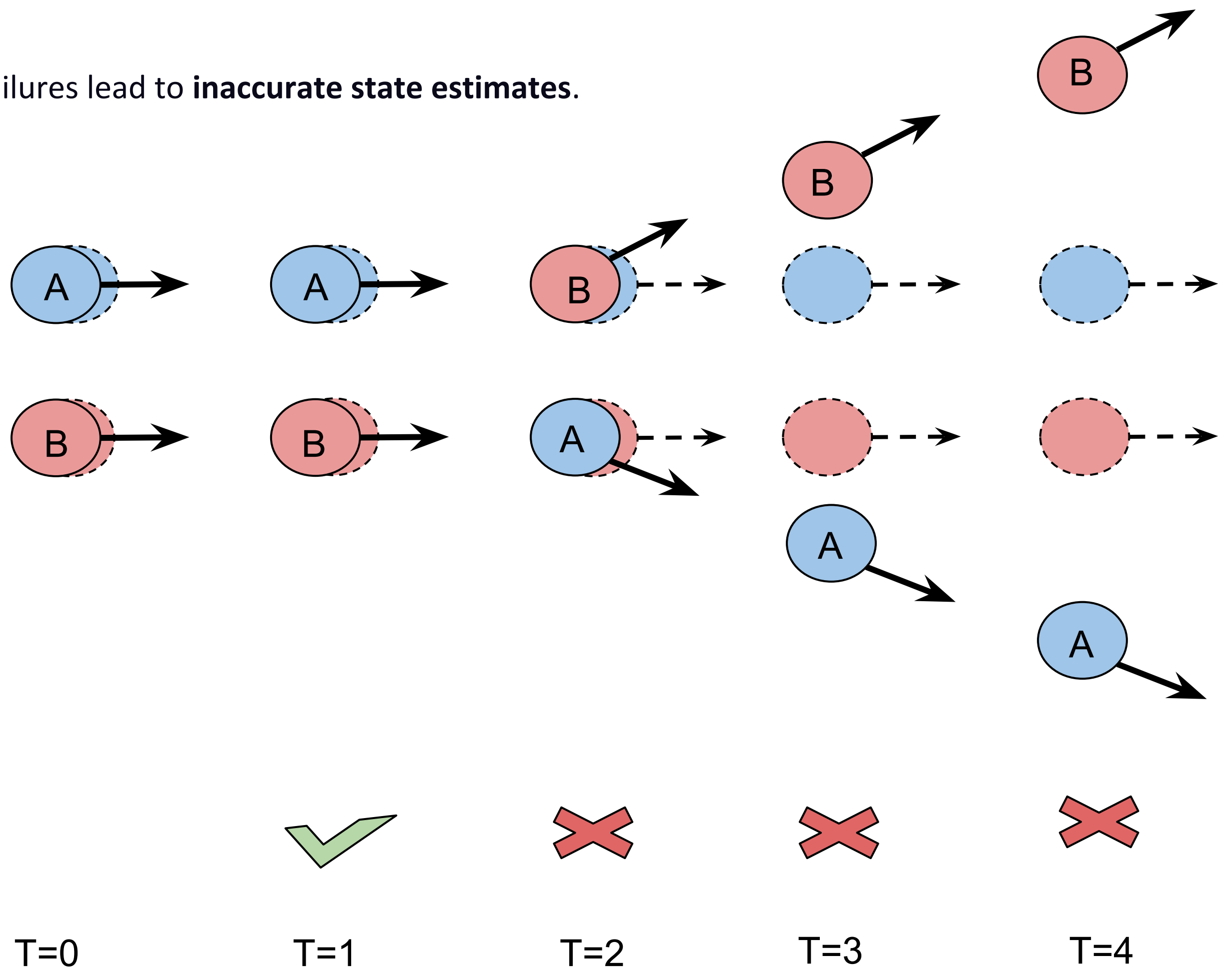
• Association failures lead to **inaccurate state estimates**.



- Association failures lead to **inaccurate state estimates**.



- Association failures lead to **inaccurate state estimates**.



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

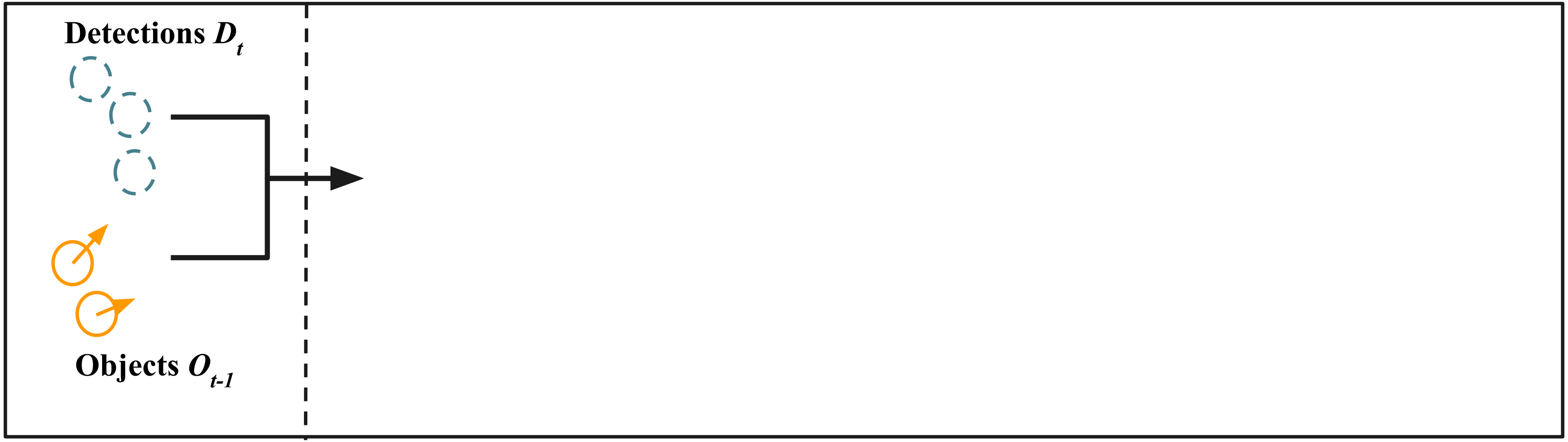
1. **State Estimation and Tracking**
   a. Filter-based tracking methods utilizing EKF/UKF are most common.
   b. IMM - Interacting Multiple Models utilizes multiple filters with unique motion models.
2. **Classical Association Methods:**
   a. **IoU score**: Thresholding on amount of overlap.
   b. **L2 score**: Thresholding on euclidean distance between observed detection and predicted position.
   c. **Mahalanobis score**: Covariance weighted distance between detection and object.

Previous Work

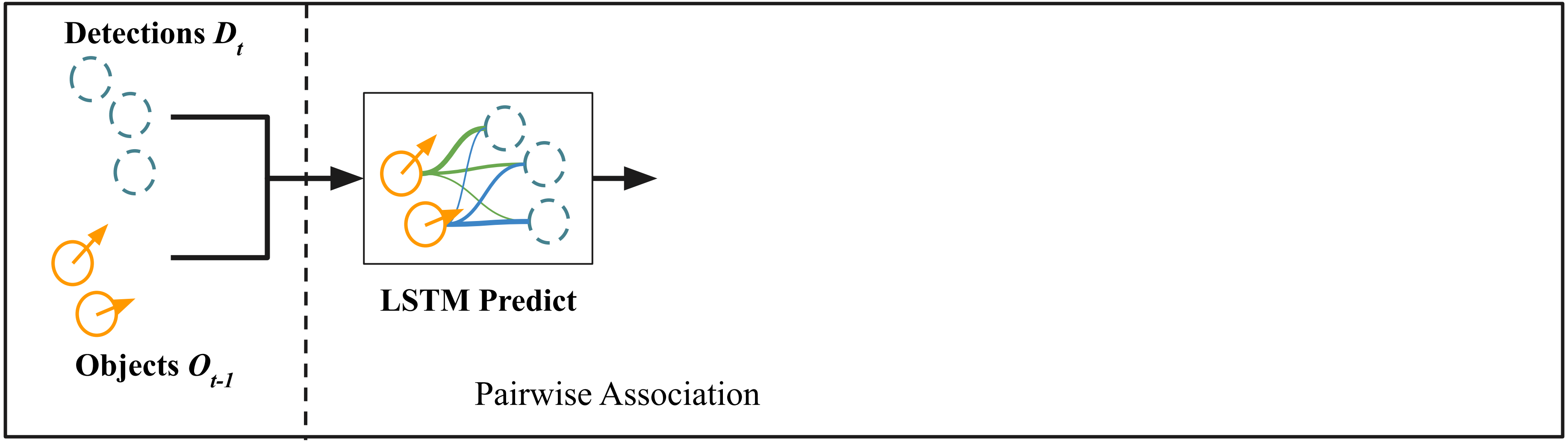
1. **State Estimation and Tracking**
   a. Filter-based tracking methods utilizing EKF/UKF are most common.
   b. IMM - Interacting Multiple Models utilizes multiple filters with unique motion models.
2. **Classical Association Methods:**
   a. **IoU score**: Thresholding on amount of overlap.
   b. **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

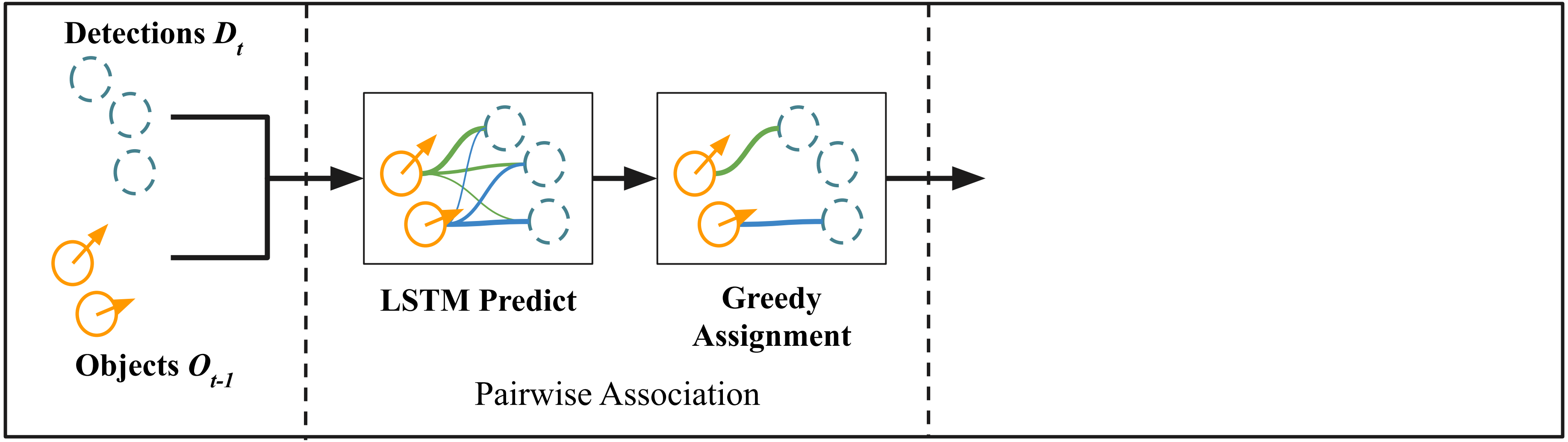
System Overview



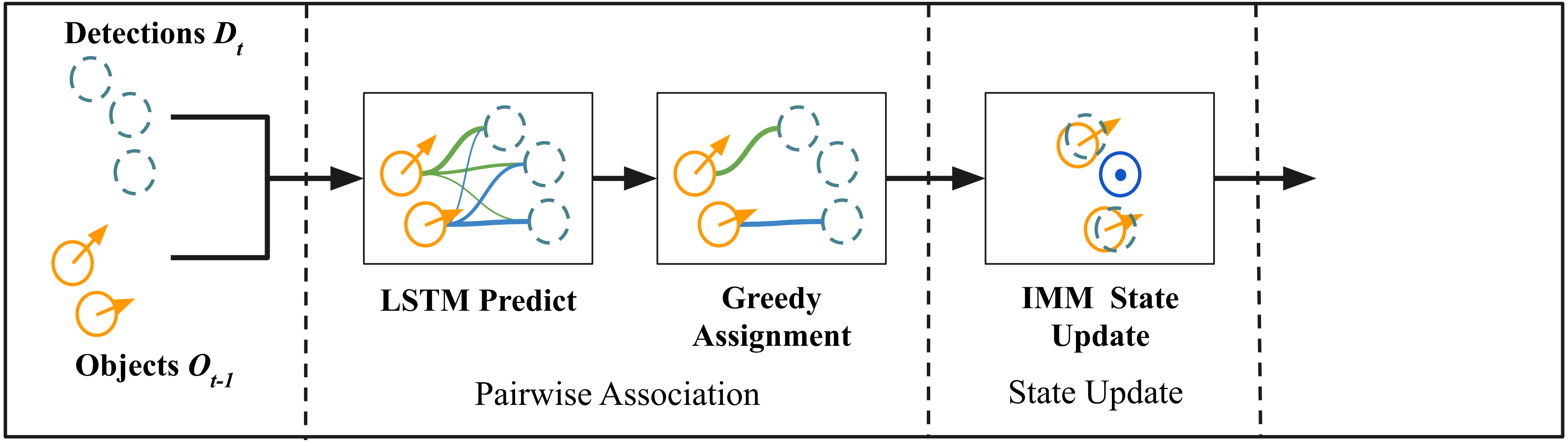
System Overview



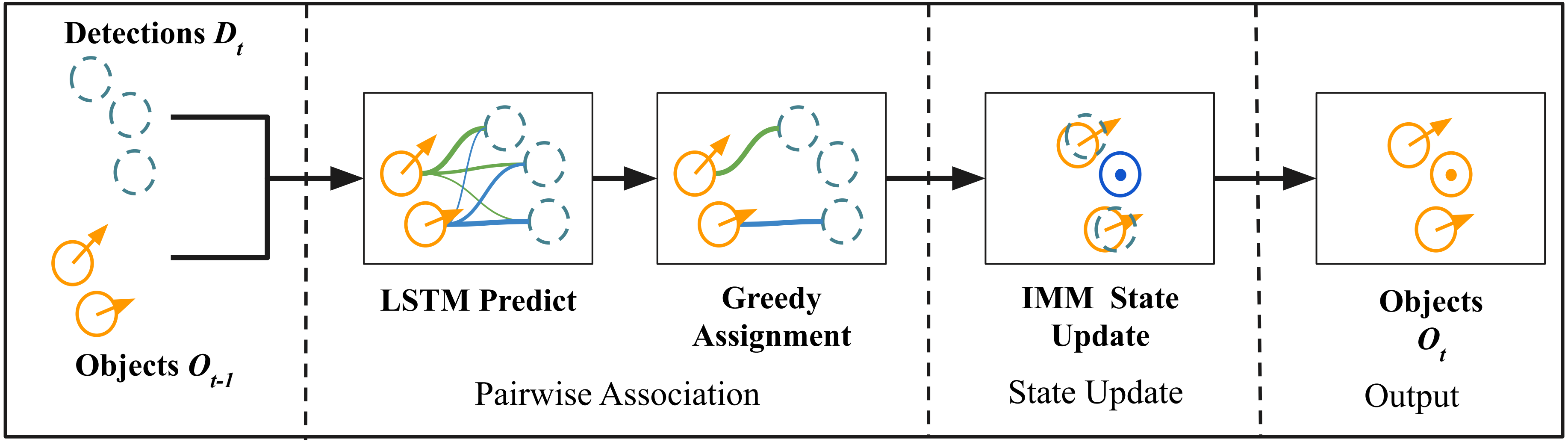
System Overview



System Overview

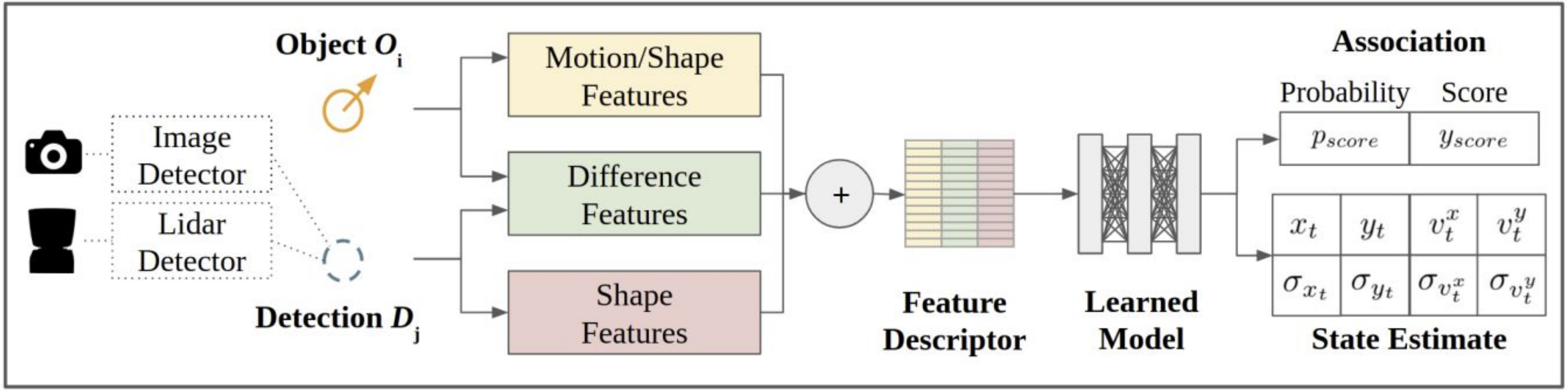


System Overview

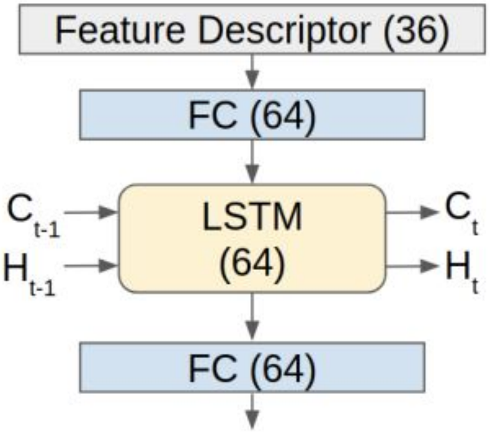


Model Details

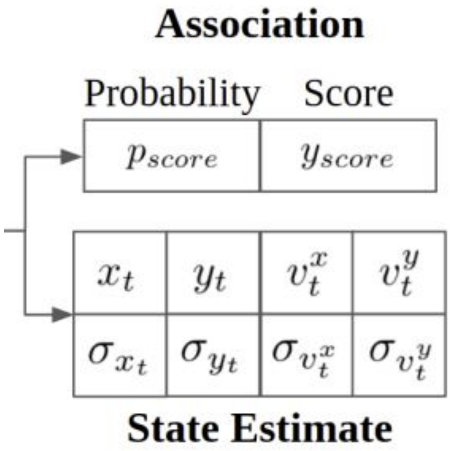
Model Overview



Model Architecture



Model Targets

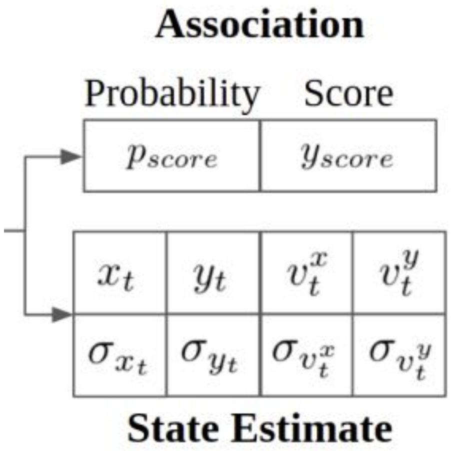


Model Targets

Association

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

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

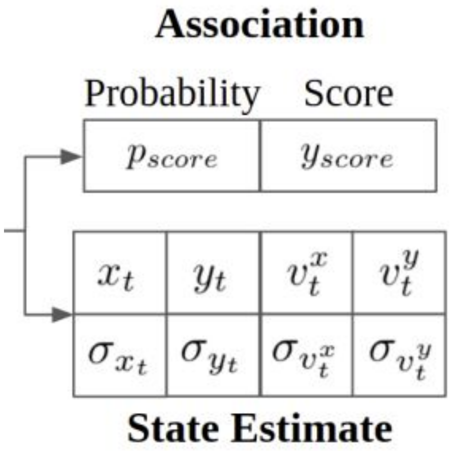


Model Targets

Association

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

b. $\mathbf{y_{score}}$: 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

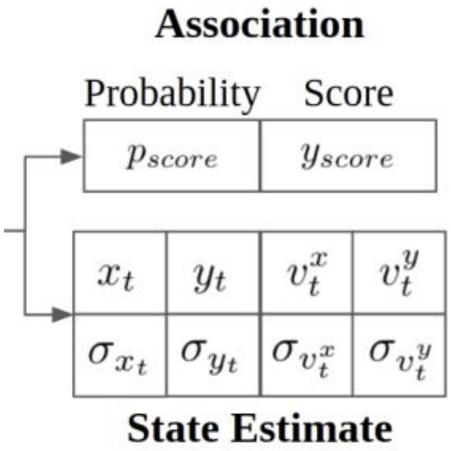
a. $\mathbf{p_{score}}$: Probability that the current detection-object pair is a true association.

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

State Estimate

a. $\mathbf{[x_t, y_t, vx_t, vy_t]}$: Position and velocity in cartesian coordinates.

b. $[\boldsymbol{\sigma}_{xt}, \boldsymbol{\sigma}_{yt}, \boldsymbol{\sigma}_{vxt}, \boldsymbol{\sigma}_{vyt}]$: Corresponding uncertainty.



Model Loss

$$\ell_{total} = \ell_{assoc} + w_{state} \cdot \ell_{state},$$

Association

$$\ell_{assoc} = \ell_{prob} + w_{score} \cdot \ell_{score},$$

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

State Estimate

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

$$\boldsymbol{s}_t = [x_t, y_t, v_t^x, v_t^y]$$

$$\boldsymbol{\sigma}_t = [\sigma_{x_t}, \sigma_{y_t}, \sigma_{v_t^x}, \sigma_{v_t^y}]$$

Results

Results

TABLE I

COMPARISON OF TRACKING METHODS ACROSS MULTIPLE SENSOR MODALITIES

Sensing Modalities	Method	MOTA ↑	MOTVO ↓		MOTVE ↓		FP ↓	FN ↓	IDSW ↓	MOTP ↓	MT ↑	ML ↓	Frag ↓
			Ped	Bike	Ped	Bike							
LiDAR	IoU-based Association	67.6486	3.572	2.377	0.170	0.287	394840	510918	44855	0.3527	0.389	0.164	39385
	L2 Association	67.9379	3.204	2.373	0.158	0.281	391298	**508561**	42092	0.3519	0.390	0.163	38850
	Mahalanobis Association	68.4670	2.956	2.041	0.157	0.271	370060	516321	37788	**0.3466**	0.386	0.165	40026
	SDVTracker (Ours)	**68.9816**	**2.199**	**1.633**	**0.131**	**0.248**	**362560**	510970	**35433**	0.3475	**0.391**	**0.162**	**38438**

- **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

Sensing Modalities	Method	MOTA ↑	MOTVO ↓		MOTVE ↓		FP ↓	FN ↓	IDSW ↓	MOTP ↓	MT ↑	ML ↓	Frag ↓
			Ped	Bike	Ped	Bike							
LiDAR + Camera	IoU-based Association	66.5809	4.236	2.549	0.192	0.295	416723	503642	51731	0.3586	0.384	0.167	43417
	L2 Association	68.1027	3.303	2.334	0.162	0.294	386467	**497147**	43991	0.3554	**0.388**	0.163	40572
	Mahalanobis Association	68.6031	3.056	2.118	0.160	0.287	366913	504202	39521	0.3498	0.385	0.165	41251
	SDVTracker (Ours)	**69.4405**	**2.204**	**1.827**	**0.133**	**0.268**	**346651**	504744	**33118**	**0.3485**	**0.388**	**0.162**	**40008**

Better at incorporating multiple sensor modalities

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

Ablation Studies

TABLE II

EFFECT OF LEARNING JOINT TRACKING AND ASSOCIATION

Network	IMM	Learning State	MOTA ↑	MOTVO ↓	MOTVE ↓	IDSW ↓
MLP	✓		69.2221	2.448	0.1446	37594
MLP	✓	✓	69.3863	2.385	0.1413	34698
LSTM	✓		69.2877	2.428	0.1419	35862
LSTM		✓	69.3971	**2.240**	0.1528	34031
LSTM	✓	✓	**69.4405**	2.292	**0.1393**	**33118**

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

Ablation on Joint Learning and Targets

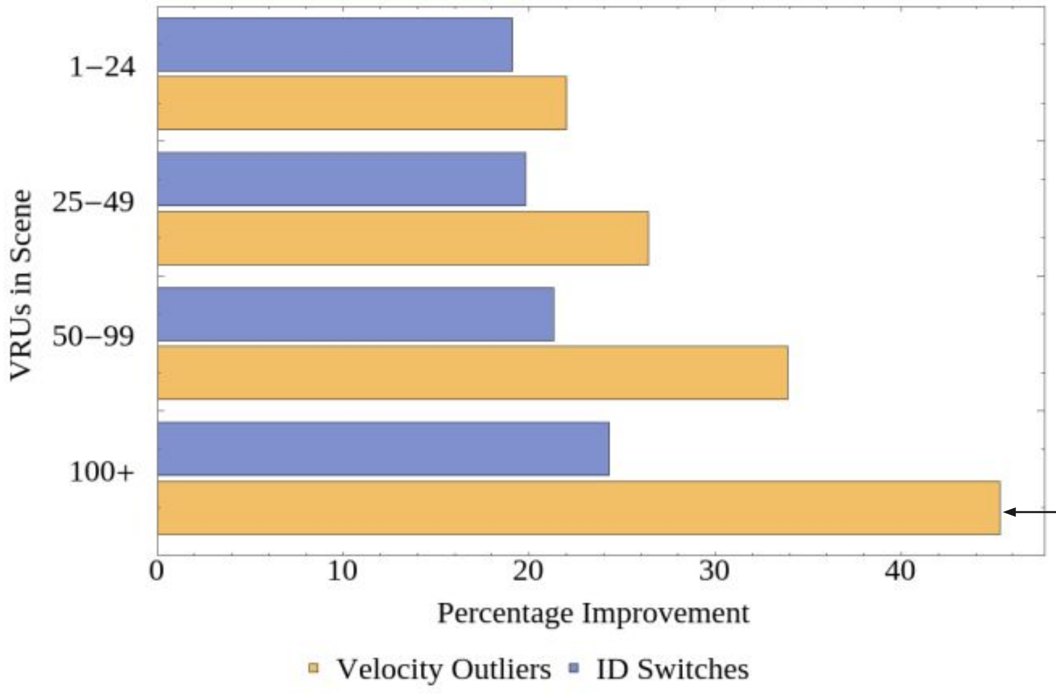
TABLE III

EFFECT OF LEARNING PROBABILITY AND SCORE

Association Output	MOTA ↑	MOTVO ↓	MOTVE ↓	IDSW ↓
Probability Only	69.1837	2.544	0.1466	35419
Score Only	69.3618	2.551	0.1448	39461
Probability and Score	**69.4405**	**2.292**	**0.1393**	**33118**

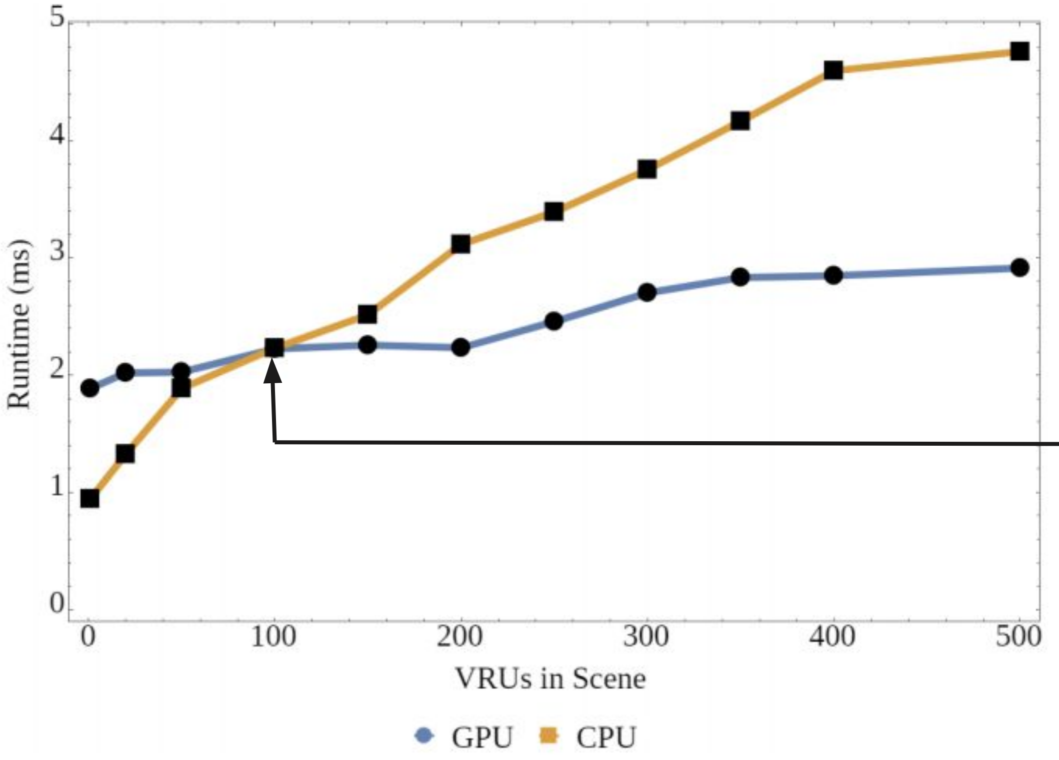
- Learning Probability and Score is better than learning either one.

Impact of Pedestrian Density

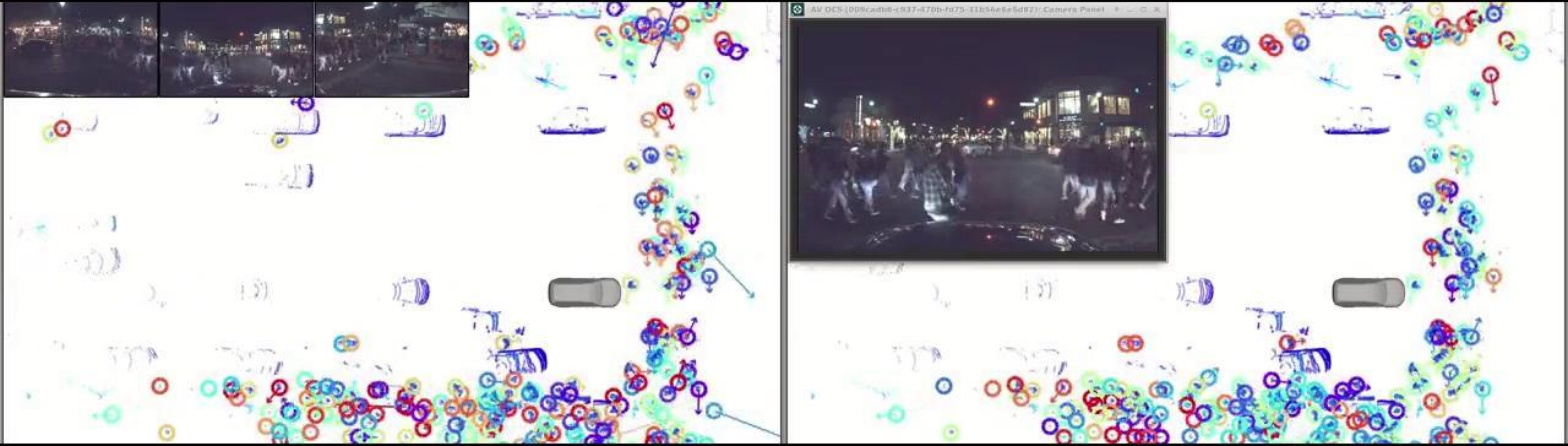


45% Less velocity outliers in most crowded scenes

Impact of Pedestrian Density



Faster to run on CPU for a scene with <100 actors!


Camera Panel

Summary

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