Robust Real-time Multi-vehicle Collaboration on Asynchronous Sensors

Qingzhao Zhang[†*], Xumiao Zhang[†*], Ruiyang Zhu[†*],

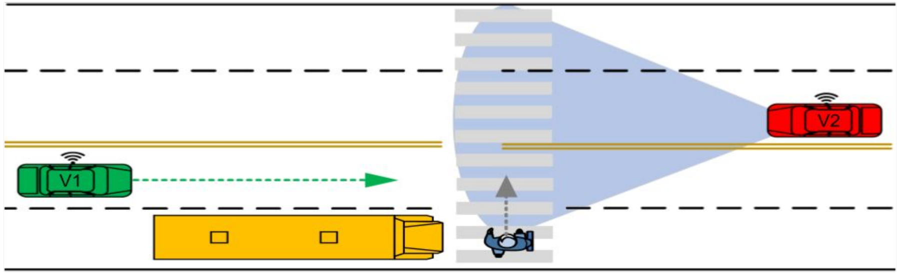
Fan Bai[‡], **Mohammad Naserian**[‡], Z. Morley Mao[†]

(*equal contributions) [†]University of Michigan [‡]General Motors

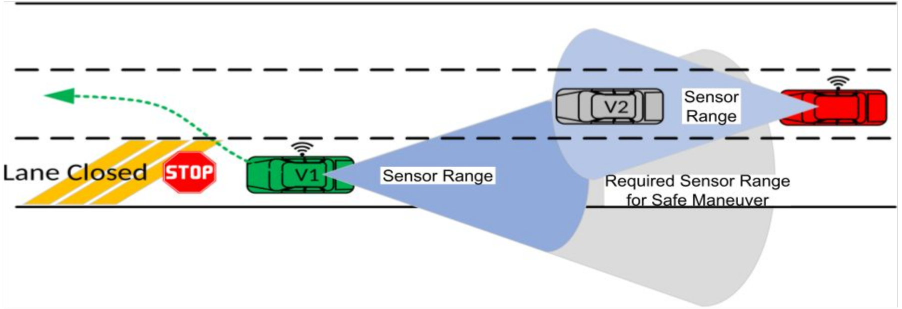
Oct. 3, 2023

Why cooperative perception?

- Limited sensing on occluded or far-away objects



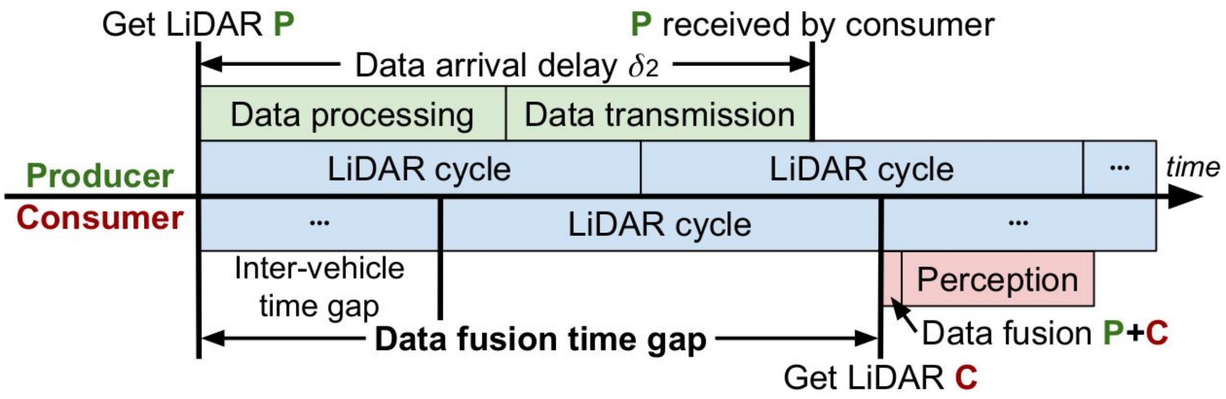
Occluded pedestrian



Far-away obstacles

Motivation 1: synchronization problem

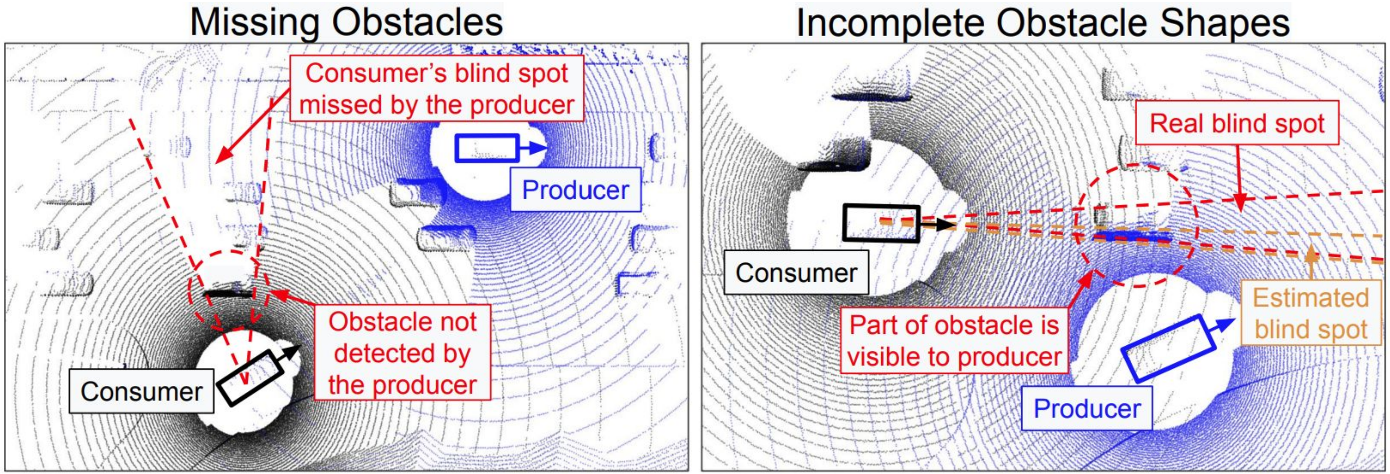
- In multi-vehicle collaboration, the LiDAR images to be merged is not captured on the same timestamp.



Consumer is the vehicle receiving LiDAR data; **provider** is the vehicle providing LiDAR data.

Motivation 2: inaccurate blind spot estimation

- Existing systems trend to share sensor data about blind spots only.
  - *However, inaccurate blind spot estimation compromise the sharing efficiency*
  - *e.g., AutoCast[1] estimate blind spots based on observed objects and naive ray*



[1] Qiu, Hang, et al. "AutoCast: scalable infrastructure-less cooperative perception for distributed collaborative driving." *Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services*. 2022.

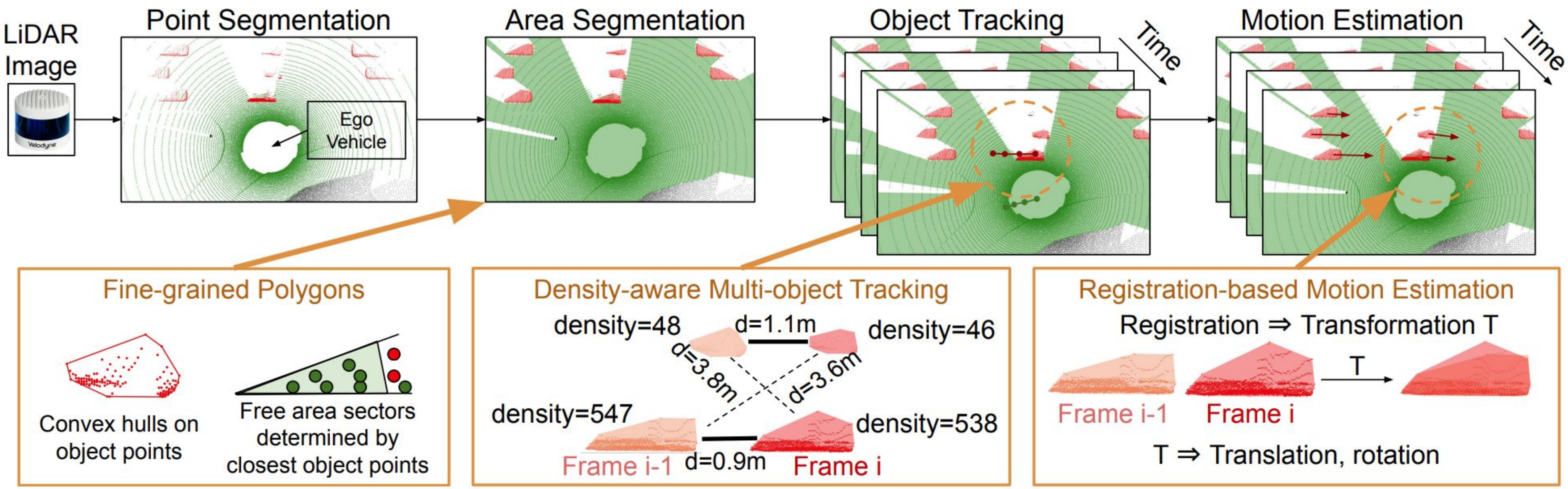
Overview

- Q: Synchronization problem?
- A: Prediction
  - *Leverage prediction algorithms to synchronize LiDAR point clouds.*

- Q: Accurate blind spot estimation?
- A: On-demand data sharing
  - *Let consumers proactively request data they need.*

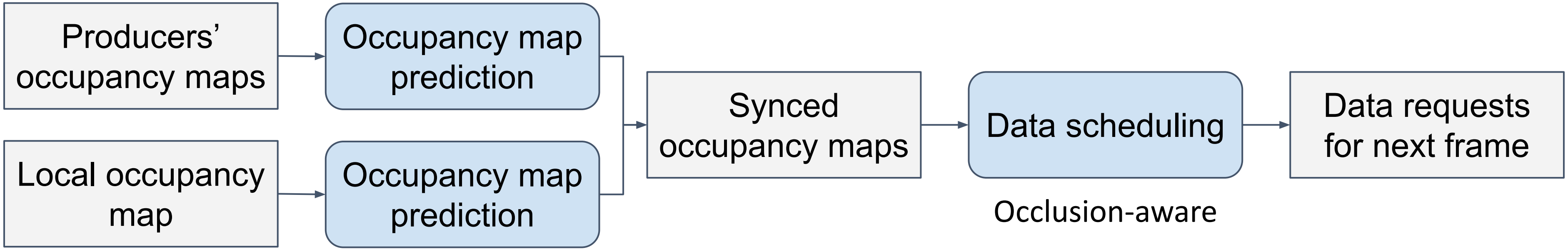
For all CAVs, share occupancy maps

- The map labels occupied, free, and occluded areas



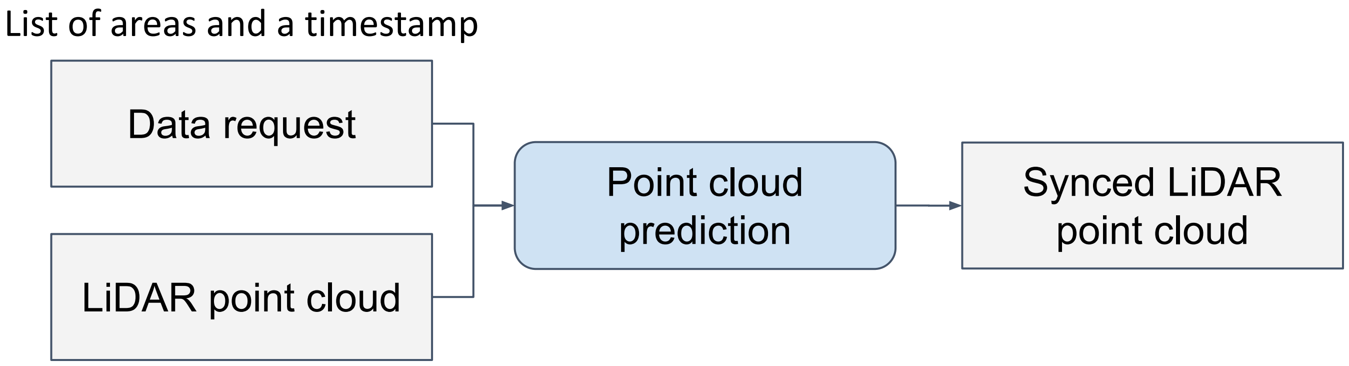
For consumers, prepare data requests

- Make a plan of data sharing for **the next LiDAR cycle**
  - *i.e., which producer share which area*



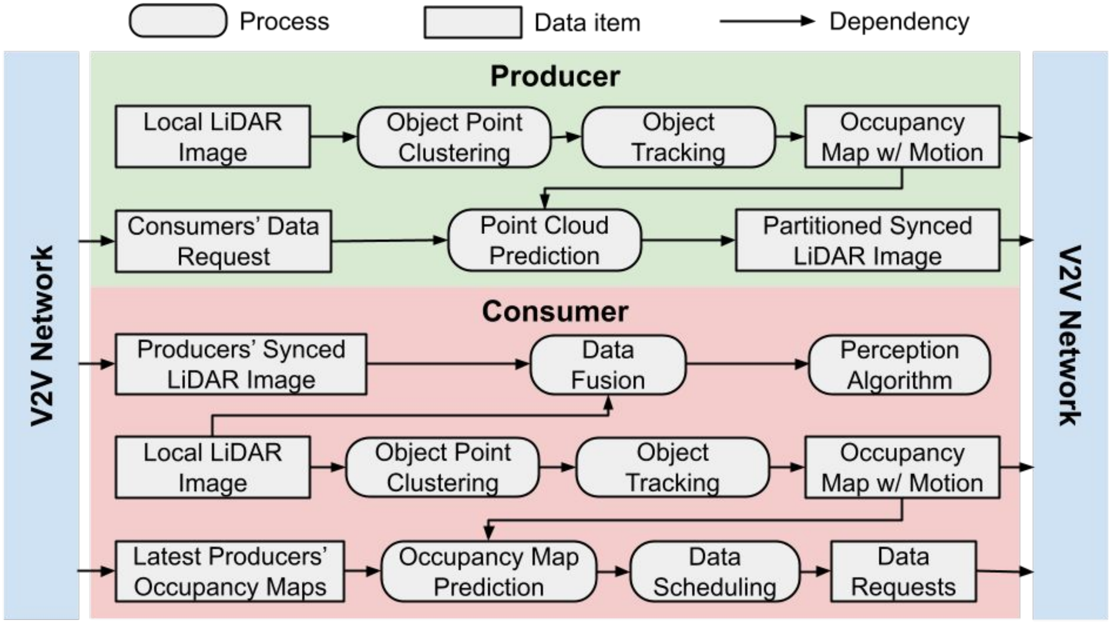
For producers, share requested data

- Share the latest point cloud on the requested areas, and synchronize the point clouds to the requested timestamp.



Execute all processes in parallel

- Compared with single-CAV perception, the only delay is from data fusion.



RAO Perception Benefits and performance

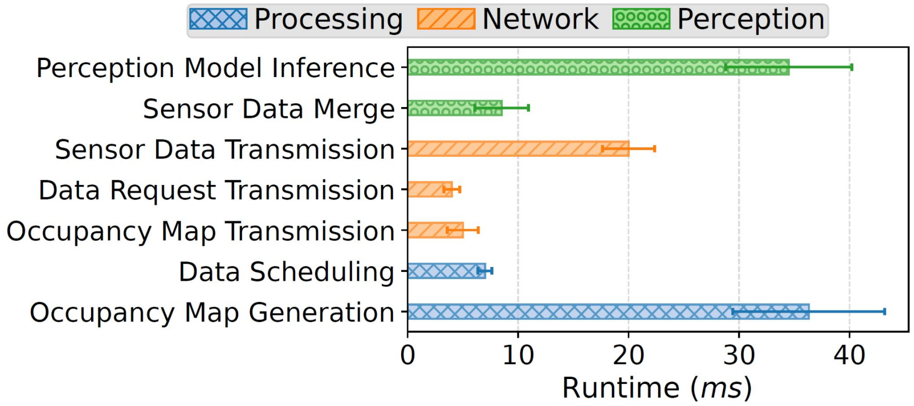
- RAO achieves the best perception accuracy compared with EMP[1] and AutoCast[2].
  - *We used various simulated and real-world datasets,*
  - *We used PointPillars as the perception model.*

Traffic Scene	**Perception AP@0.5/AP@0.7**			
	Local-only	EMP	AutoCast	***RAO***
DAIR-V2X-C	48.99/40.78%	48.82/40.68%	50.36/41.18%	**53.11/43.49%**
CARLA-SUMO	48.63/37.17%	64.08/54.26%	64.91/51.50%	**74.79/62.01%**
- Town05	40.68/30.18%	48.63/38.25%	63.61/39.88%	**69.81/58.72%**
- Town06	65.46/48.30%	73.22/53.22%	67.55/58.47%	**81.72/65.19%**
- Town10HD	40.12/32.58%	64.34/57.18%	69.90/52.50%	**78.53/65.25%**
Mcity	51.51/41.13%	64.88/50.50%	65.76/48.32%	**69.13/51.25%**

[1] Zhang, Xumiao, et al. "Emp: Edge-assisted multi-vehicle perception." *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*. 2021.
[2] Qiu, Hang, et al. "AutoCast: scalable infrastructure-less cooperative perception for distributed collaborative driving." *Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services*. 2022.

System Overhead - Latency & Data Volume

- The total avg latency of all the modules is 80.82 ms (14.40 ms variance)
- RAO can process LiDAR at regular full frame rate of 10 FPS
- RAO incurs similar data overhead compared to the STOA approach



Metrics	EMP	AutoCast	***RAO***
LiDAR Points	8320±3228	3140±2171	3110±2501
Control data (KB)	<0.1	<0.1	1.77±0.50
Total Volume (KB)	24.37±9.46	9.17±6.36	10.90±7.32

Summary

- RAO is a **real-time occlusion-aware** cooperative perception system running on **asynchronous** sensors.

- RAO tackles two problems in existing cooperative perception.
  - *Use prediction methods to mitigate sensor asynchronization.*
  - *Use on-demand data sharing to optimize data scheduling.*

Thank You!