Lane:

A Deep Learning Treatment

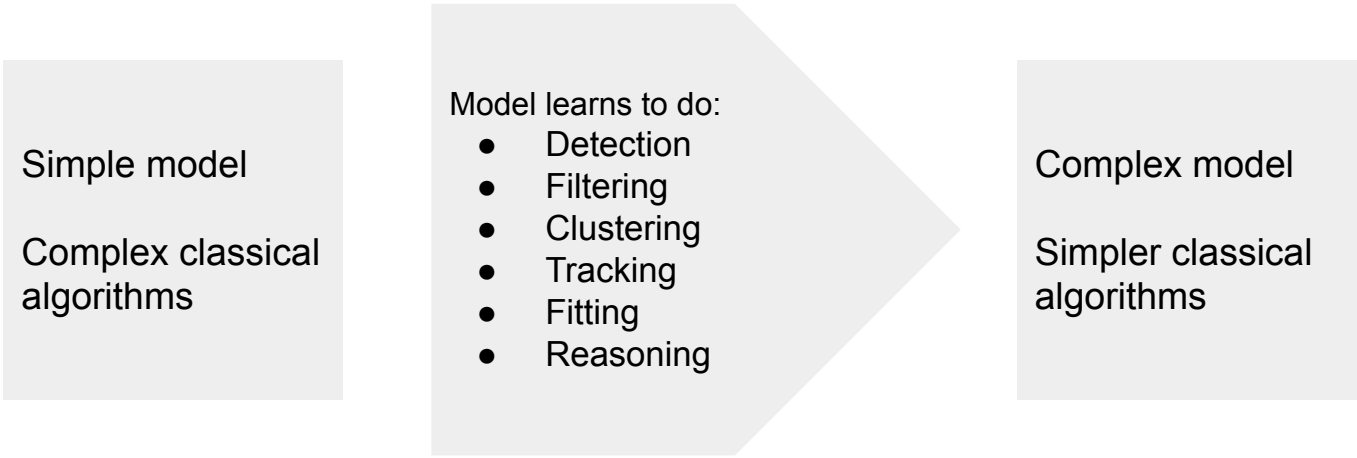
Ankur Agarwal

Acknowledgement

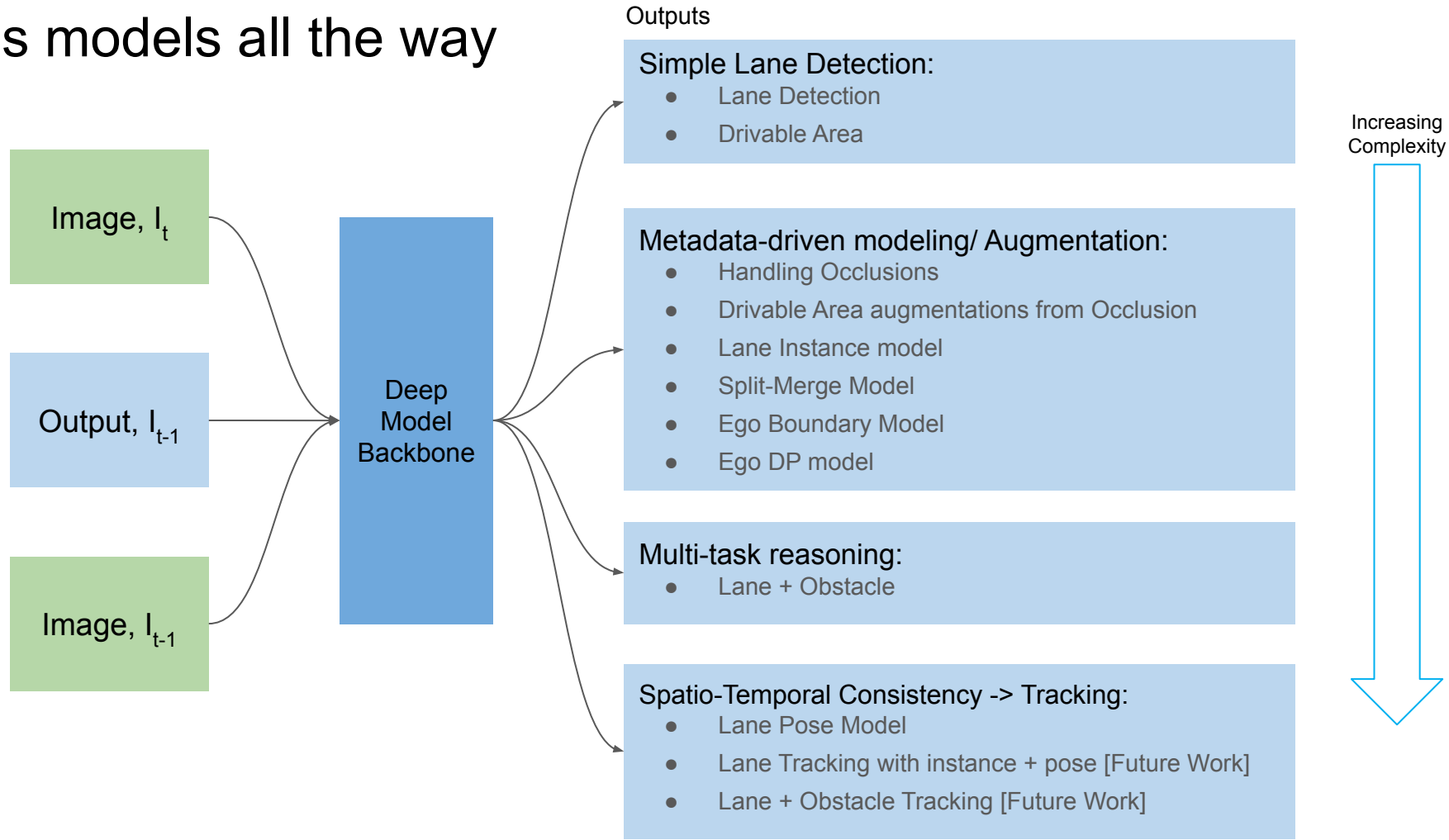
The work shown here is a contribution of the **‘Perception: Lane’ team** in **China** and **US.**

The goal

- Hand-crafted logic and heuristics → data driven models
- Redundancy when generating critical lane outputs
- Make classical and deep learning approaches self-consistent



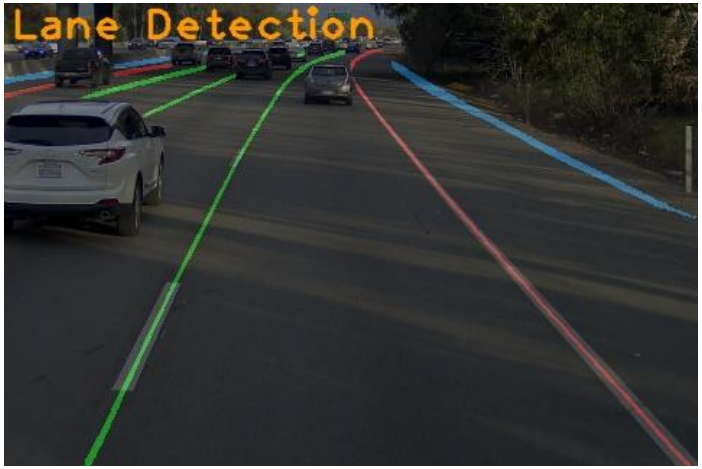
It’s models all the way




Lane Detection
Drivable Area
Occluded Lanes
Ego Boundaries
Ego Driving Path

Lane + Drivable Area Model

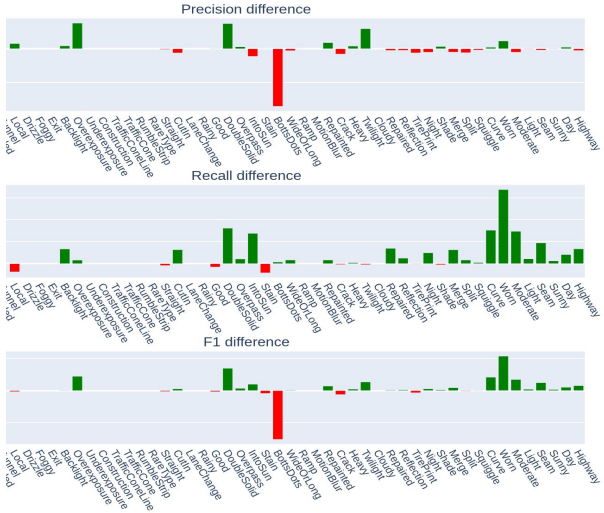
- Different lane types: solid, dashed, curb
- Drivable Area
- “Almost” camera-pose agnostic; we can use front, rear or side cameras with minimal retraining/ fine-tuning effort.



		0811v4	Unified lane + drivable 20210803	Lane + Occluded lane model 20210814	Lane + drivable area + occluded lane (+ artificial occlusions)
solid	**f1**	0.909	0.914	0.912	0.914
	recall	0.906	0.921	0.911	0.907
	precision	0.912	0.908	0.913	0.921
dash	**f1**	0.847	0.860	0.855	0.860
	recall	0.809	0.835	0.827	0.838
	precision	0.887	0.887	0.885	0.887
curb	**f1**	0.723	0.731	0.726	0.726
	recall	0.761	0.777	0.768	0.768
	precision	0.689	0.691	0.688	0.689

	Drivable area region metrics				Drivable area boundary metrics		
	precision	**recall**	**f1**	**IoU**	**precision**	**recall**	**f1**
20210124	0.997	0.988	0.993	0.986	0.822	0.793	0.808
20210722	0.997	0.990	0.994	0.988	0.851	0.823	0.837

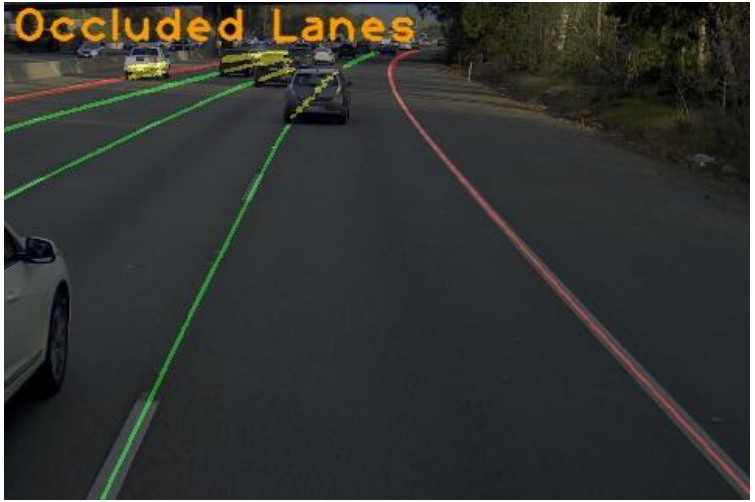
Solid



Dash

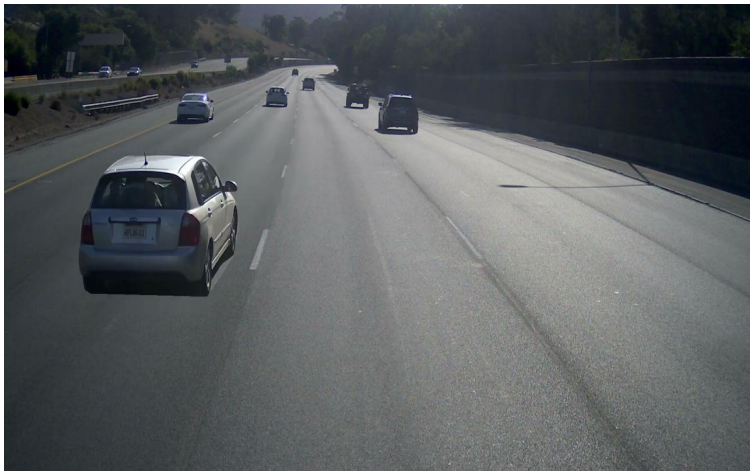
Handling Occlusions:

- Model infers lane marks occluded by obstacles.
- Classical logic gets some more information to work with.
- Helps with:
  - Vehicles cutting-in; especially during curves on a non-planar road surface.



Using Drivable Area for augmenting data:

- Reproduce not-very-frequent, but hard, cases.
- Bypass the need for human labeling

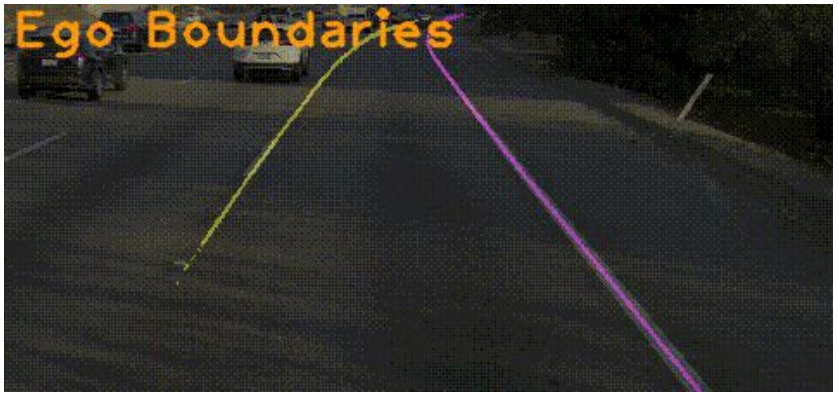


Ego Driving Path Model

- Directly outputs driving path in image space.
- Helps with:
  - A redundant driving path estimate.
- Future work: generate all valid lane paths.
- Combined with the obstacle model, use other vehicle trajectories to create a valid path

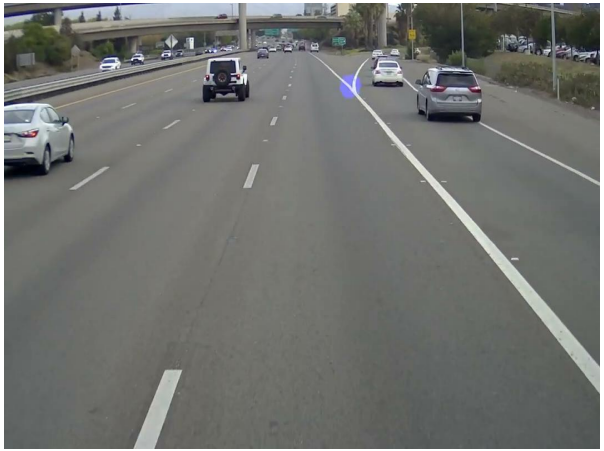
Ego Boundary Model

- Detecs lane marks that belong to ego left and right boundaries.
- Helps with:
  - Multiple faint/ old lane marks
  - Large motion/ bumps that affect tracking



Split/ Merge Models

- Detects different types of split and merge geometries.



Instance Segmentation

- Clusters points belonging to a lane boundary, in image space
- Helps with:
  - Reducing filtering+clustering code complexity
  - Noisy lane marks
  - Non-planar road surface

Lane + Obstacle

- Model learns to reason about lane AND obstacles.

Model	Parameters	Training Data
Lane	1.2M	84K (lane)
Lane + Drivable Area	1.2M	100K (lane), 34K (drivable area)
Lane + Drivable Area + Occluded lane	1.2M	96K (lane instances) + 10K (artificial GT)
Lane + Drivable Area + Occluded lane + Ego DP	1.2M	40K (ego DP)
Lane + Drivable Area + Split/ Merge	1.3M	27K (various split/ merge scenarios)
Lane + Drivable Area + Instance	1.37M	96K (lane instances)
Lane + Obstacle	20M	140K (2D obstacle), 220K (lane)