Uncertainty-Aware Vehicle Orientation Estimation for Joint Detection-Prediction Models

Henggang Cui, Fang-Chieh Chou, Jake Charland,
Carlos Vallespi-Gonzalez, Nemanja Djuric

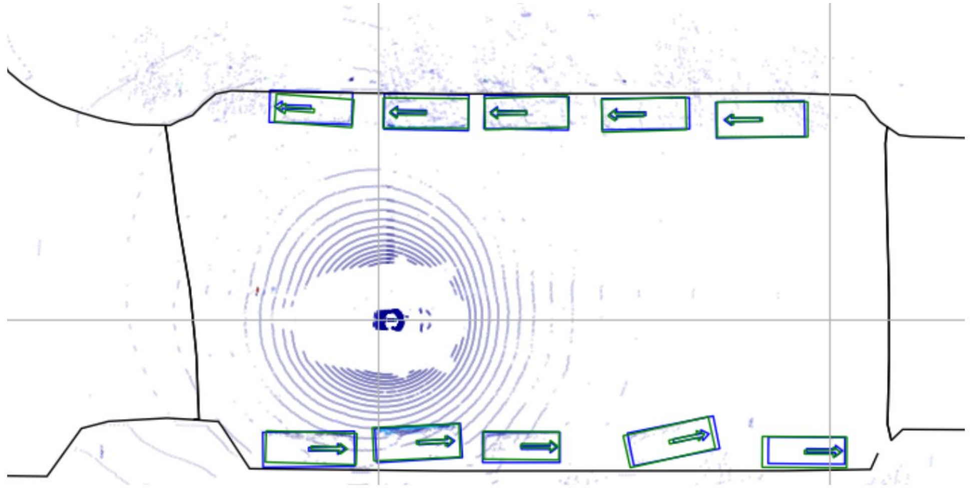
Uber Advanced Technologies Group

IEEE ITSC 2021
INDIANAPOLIS, USA

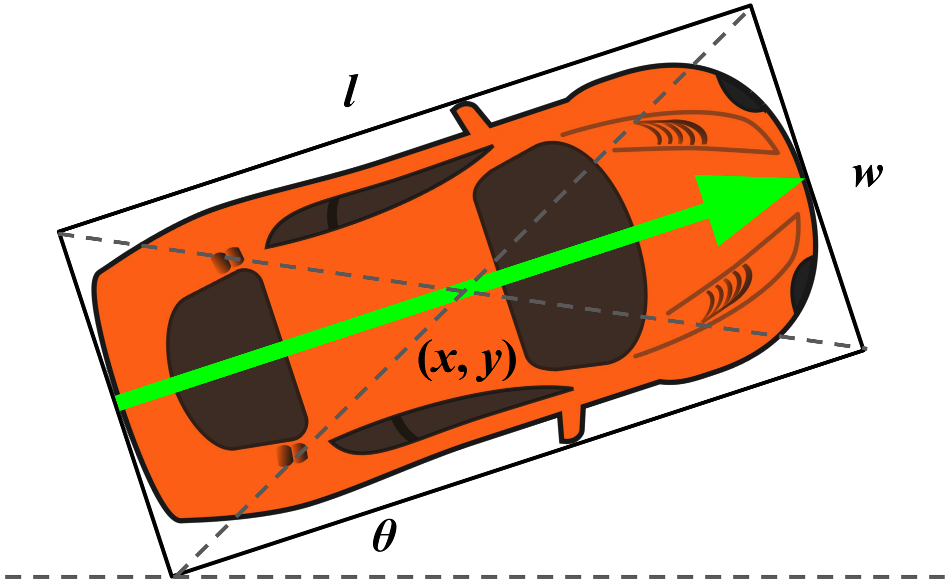
UBER ATG

Detection and orientation estimation for autonomous driving

- Important property for downstream modules of an autonomous system
- Inputs are LiDAR point clouds; outputs are detection boxes, parameterized as (x, y, l, w, θ)
- State-of-the-art approach is to use a deep neural network (e.g., SSD, YOLO, CenterPoint, etc.)
- The network outputs the box parameters for every prior and trains on ground-truth labels



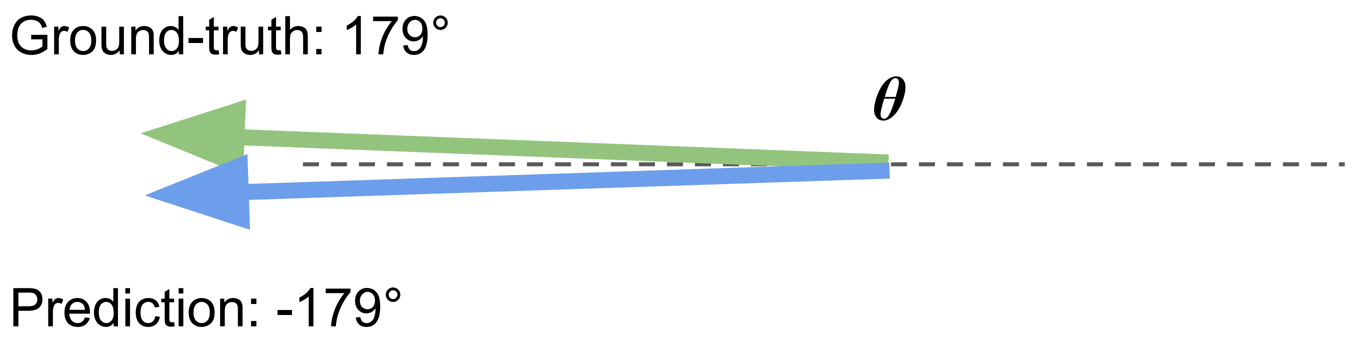
Input LiDAR point clouds and output detection boxes



Detection boxes parameterization

Detection and orientation estimation for autonomous driving

- The (x, y, l, w) parameters are usually trained with L1 loss $\mathcal{L} = \text{L1}(gt, pred)$
- However, we cannot train the orientation θ directly with L1 loss because the space is circular
  - E.g., L1(179°, -179°) is large even though the actual angle difference is only 2°



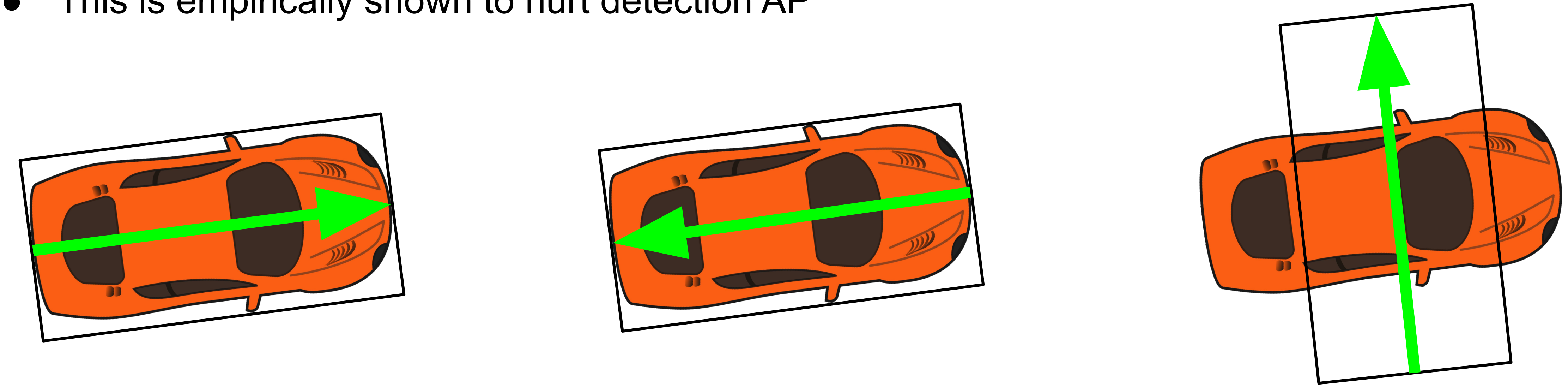
UBER ATG

Traditional orientation estimation methods

- In Luo et al., their model outputs $(\sin(\theta), \cos(\theta))$ and trains them with

$$\mathcal{L}_{full} = \ell_1\left(\sin(\hat{\theta}) - \sin(\theta)\right) + \ell_1\left(\cos(\hat{\theta}) - \cos(\theta)\right)$$

- Decode orientation $\theta = \arctan(\sin(\theta), \cos(\theta))$
- However, a 180°-flipped orientation gets a higher loss than a 90°-error of the orientation
- This is empirically shown to hurt detection AP



Ground-truth

180°-flipped orientation
OK detection but high loss

90°-error orientation
Terrible detection but lower loss

Traditional orientation estimation methods

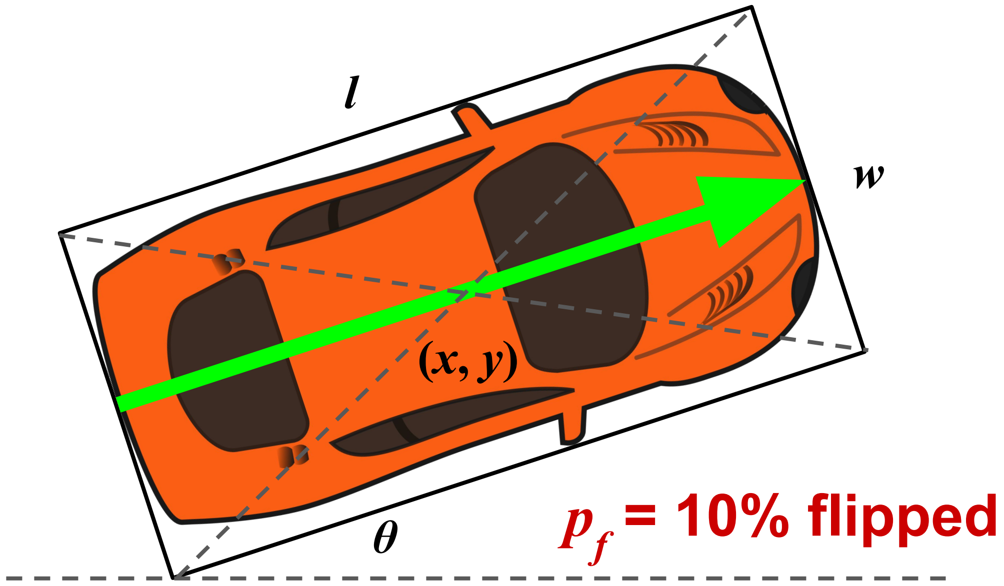
- To mitigate this issue, some works do not penalize the flipped orientations at all in the loss
- Yang et al. predict the orientations as $(\sin(2\theta), \cos(2\theta))$ and train them with

$$\mathcal{L}_{half} = \ell_1\big(\sin(2\hat{\theta}) - \sin(2\theta)\big) + \ell_1\big(\cos(2\hat{\theta}) - \cos(2\theta)\big)$$

- Decode orientation $\theta = \arctan(\sin(2\theta), \cos(2\theta)) / 2$
  - Decoded orientations are in (-90, 90] “half-range”
- A flipped orientation will have zero loss
  - The resulting model is not able to distinguish the front and back of a vehicle
  - Not desired for autonomous driving

Flip-aware orientation estimation

- We propose a *flip-aware orientation estimation* method
  - It will not over-penalize the model when it predicts a flipped orientation
  - But it will still encourage the model to predict the correct orientation
- The model outputs $(\sin(\theta), \cos(\theta), p_f)$
  - p_f is the estimated probability that $\theta + 180°$ is closer to ground-truth than θ



IEEE ITSC 2021 INDIANAPOLIS, USA

UBER ATG

Flip-aware orientation estimation

- $(\sin(\theta), \cos(\theta), p_f)$ are trained with

$$\boxed{\mathcal{L}_{final} = \mathcal{L}_{half} + \min(\mathcal{L}_{full}, \mathcal{L}_{flipped}) + \mathrm{CrossEntropy}(\hat{p}_f, \mathbb{1}_{\mathcal{L}_{full} > \mathcal{L}_{flipped}})}$$

;

  - $\mathcal{L}_{half}$ is half-range loss but computed from $(\sin(\theta), \cos(\theta))$

$$\mathcal{L}_{half} = \ell_1\big(\sin(2\hat{\theta}) - \sin(2\theta)\big) + \ell_1\big(\cos(2\hat{\theta}) - \cos(2\theta)\big)$$

;

$$\sin(2\hat{\theta}) = 2\sin(\hat{\theta})\cos(\hat{\theta}) \;;\; \cos(2\hat{\theta}) = \cos^2(\hat{\theta}) - \sin^2(\hat{\theta})$$

  - $\mathcal{L}_{full}$ is full-range loss $\mathcal{L}_{full} = \ell_1\big(\sin(\hat{\theta}) - \sin(\theta)\big) + \ell_1\big(\cos(\hat{\theta}) - \cos(\theta)\big)$

  - $\mathcal{L}_{flipped}$ is full-range loss of the flipped orientation

$$\mathcal{L}_{flipped} = \ell_1\big(-\sin(\hat{\theta}) - \sin(\theta)\big) + \ell_1\big(-\cos(\hat{\theta}) - \cos(\theta)\big)$$

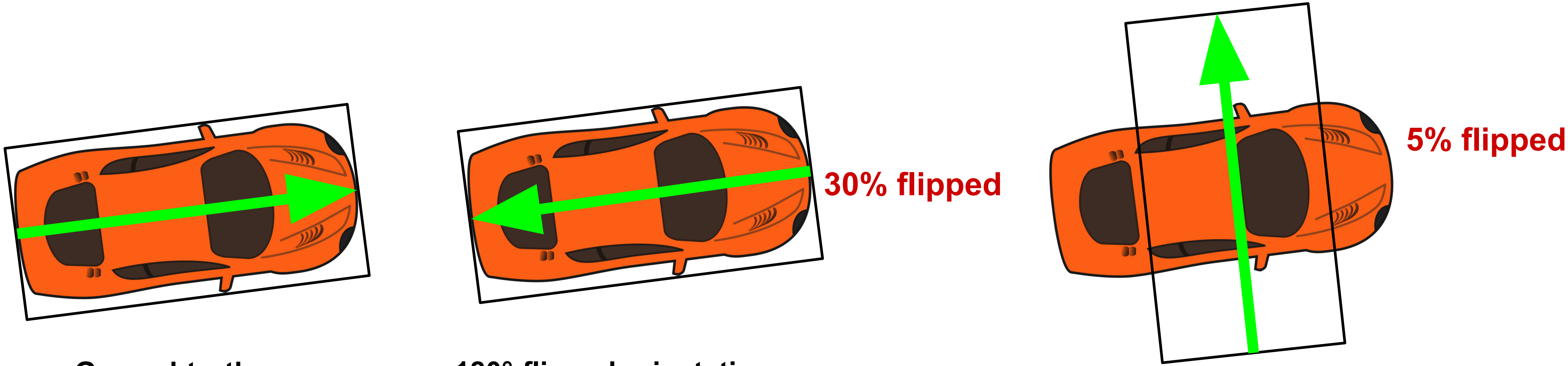
  - Cross-entropy loss to train p_f, where the GT is decided by which of $\mathcal{L}_{full}$ or $\mathcal{L}_{flipped}$ is smaller

Flip-aware orientation estimation

- $(\sin(\theta), \cos(\theta), p_f)$ are trained with

$$\boxed{\mathcal{L}_{final} = \mathcal{L}_{half} + \min(\mathcal{L}_{full}, \mathcal{L}_{flipped}) + \text{CrossEntropy}(\hat{p}_f, \mathbb{1}_{\mathcal{L}_{full} > \mathcal{L}_{flipped}})},$$

- A 180°-flipped orientation only gets penalized by the cross-entropy loss term
  - p_f will be pushed to increase by the cross-entropy loss
  - $\sin(\theta)$ and $\cos(\theta)$ will stay the same

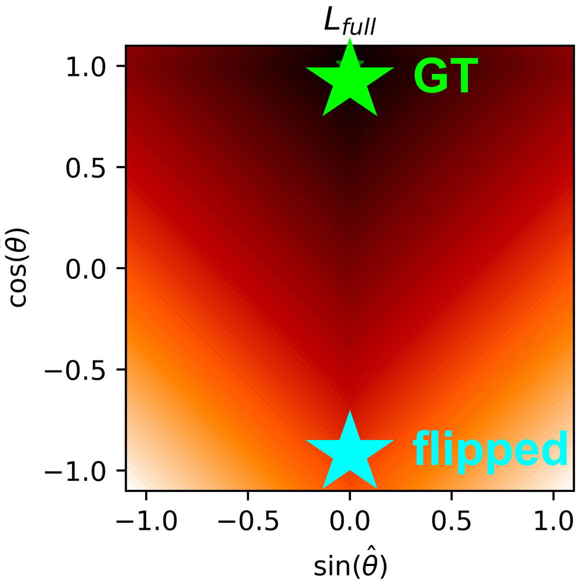


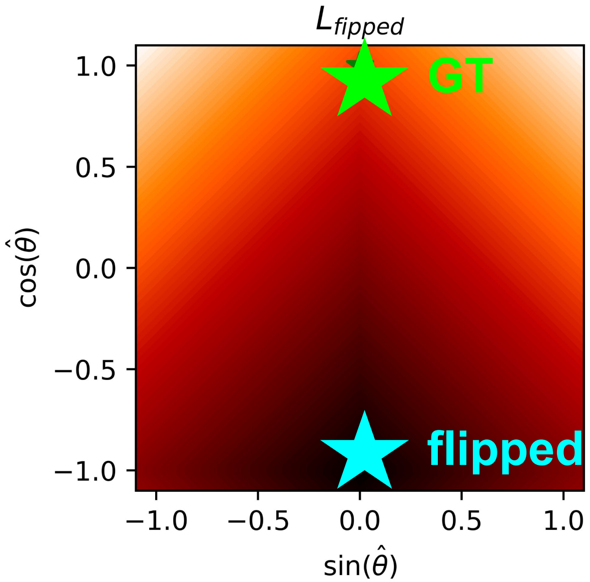
Ground-truth

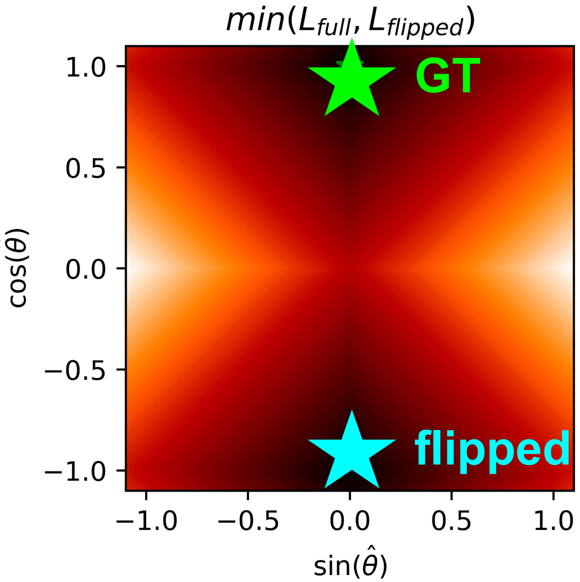
180°-flipped orientation
Penalized by cross-entropy loss

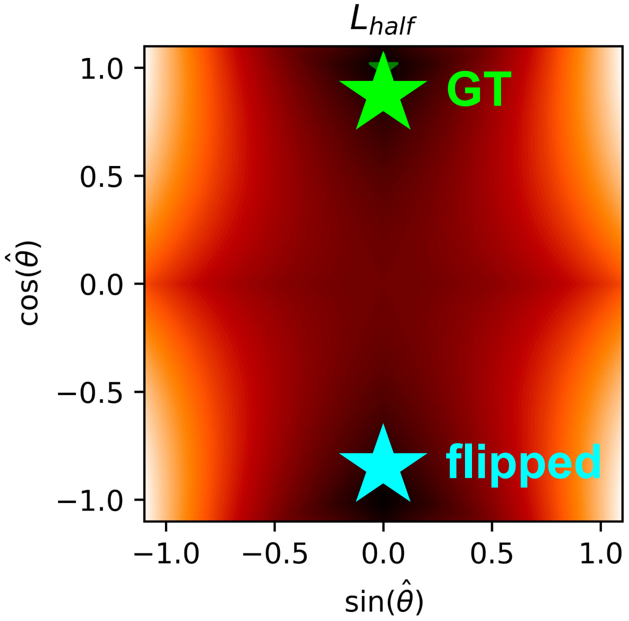
90°-error orientation
High regression loss

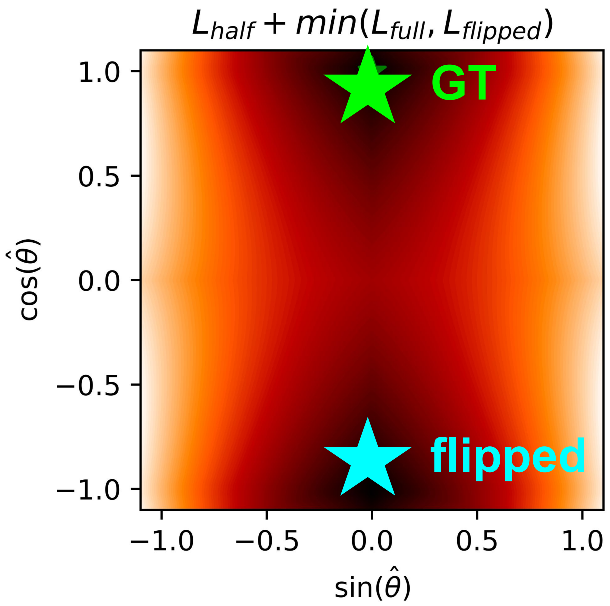
Loss analysis











- Darker color means smaller loss
- Ground-truth $(\sin(\theta), \cos(\theta)) = (0, 1)$
- A flipped orientation (0, -1) has zero loss from the regression terms and only gets penalized by the cross-entropy loss

Experimental setup

- Implemented flipped-aware method in MultiXNet [1]
  - A joint detection-prediction model for autonomous driving
- Baselines
  - Full-range loss $\mathcal{L}_{\text{full}}$
  - Half-range loss $\mathcal{L}_{\text{half}}$
  - MultiBin [3] with 2 bins and 4 bins
- Dataset
  - nuScenes [2]: 1,000 scenes collected from public roads in Boston and Singapore

IEEE ITSC 2021
INDIANAPOLIS, USA

UBER ATG

[1] Djuric et al. MultiXNet: Multiclass Multistage Multimodal Motion Prediction.

[2] Caesar et al. nuScenes: A Multimodal Dataset for Autonomous Driving.

[3] Mousavian et al. 3D Bounding Box Estimation Using Deep Learning and Geometry.

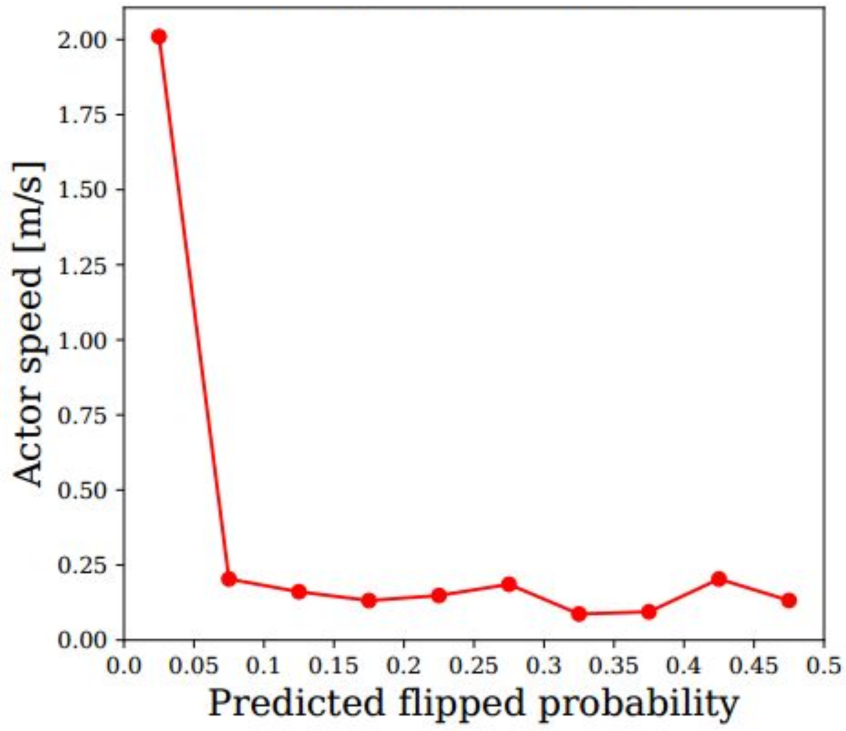
Quantitative results

Method	$AP_{0.7}$ ↑	Orientation error [deg] ↓	$AOS_{0.7}$ ↑
$\mathcal{L}_{\text{full}}$	57.1 ± 0.5	**8.2 ± 0.2**	55.1 ± 0.5
$\mathcal{L}_{\text{half}}$	**60.8 ± 1.0**	59.9 ± 1.4	40.7 ± 0.7
`MultiBin-2`	57.3 ± 0.6	9.4 ± 0.7	55.0 ± 0.3
`MultiBin-4`	58.0 ± 0.5	9.8 ± 0.5	55.5 ± 0.3
`Flip-aware`	**60.7 ± 0.2**	9.6 ± 0.8	**57.9 ± 0.4**

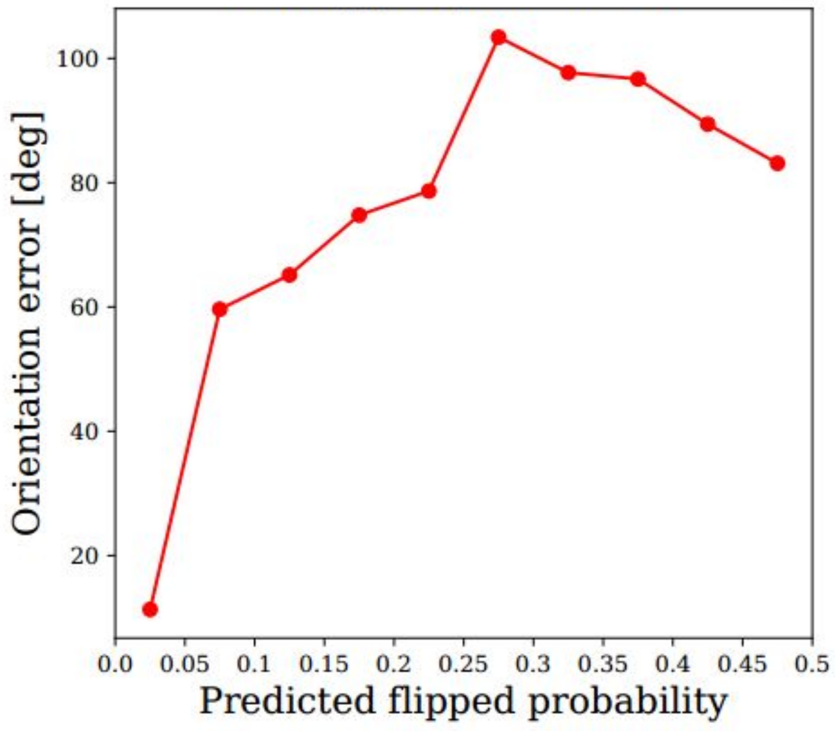
- Metrics:
  - Average precision (AP) for detection performance
  - Orientation error
  - Average Orientation Similarity (AOS) for both detection and orientation performance
- $\mathcal{L}_{\text{half}}$ has better detection AP but is not able to distinguish front and back of a car
- Flip-aware has similar AP as $\mathcal{L}_{\text{half}}$ and has the best AOS among all methods

Analysis for the flipped probability p_f output

Flipped prob vs. speed



Flipped prob vs. orientation error



- Most of the actors that are predicted to have high flipped probabilities are static actors with 0 speed
- The actors that are predicted to have higher flipped probabilities tend to have larger orientation errors
    - The flipped probability is a good measurement of the orientation uncertainty

UBER ATG

Ablation study

Method	$AP_{0.7}$ ↑	Orientation error [deg] ↓	$AOS_{0.7}$ ↑
`Flip-aware`	**60.7**	9.6	**57.9**
`*-no-half`	59.6	10.1	55.2
`*-no-flip`	58.9	**7.8**	56.8

- Flip-aware-no-half
    - Without the half-range loss term $\mathcal{L}_{\text{half}}$
- Flip-aware-no-flip
    - Without the flipped probability output p_f and use $\mathcal{L}_{\text{full}} + \mathcal{L}_{\text{half}}$ as loss
- Both variations underperform the proposed flip-aware loss

UBER ATG

Conclusions

- We proposed a *flip-aware orientation estimation* method
  - Estimate box orientations with a probabilistic output that estimated orientation is flipped
  - Improves both the orientation estimation and detection accuracies