Learning to Localize Using a LiDAR Intensity Map

Uber ATG

Andrei Bârsan*[1,2], Shenlong Wang*[1,2], Andrei Pokrovsky[1], Raquel Urtasun[1,2]

[1]Uber ATG, [2]University of Toronto

UNIVERSITY OF TORONTO

1 Motivation

- Robust and accurate localization is one of the cornerstones of an autonomous driving stack.
- Goal: Perform real-time online localization w.r.t. an HD LiDAR intensity map with centimeter-level accuracy.
- Challenges:

Dynamic Objects

Lack of Geometric Cues

Different LiDAR Types

- Past approaches:
  - suffer in geometrically degenerate environments, e.g., bridges
  - cannot generalize to different LiDARs without calibration.

2 Probabilistic Localization

- We learn to match between online sensory observations and a map.
- We incorporate this learned component into a histogram filter together with GPS information:

$$\mathrm{Bel}_t(\mathbf{x}) = \eta \cdot \boxed{P_{\mathrm{LiDAR}}(\mathcal{I}_t|\mathbf{x};\mathbf{w})}\ \boxed{P_{\mathrm{GPS}}(\mathcal{G}_t|\mathbf{x})}\ \boxed{\mathrm{Bel}_{t|t-1}(\mathbf{x}|\mathcal{X}_t)}$$

This gives us a probability distribution over the vehicle pose in world coords. The discretization is centered around the dead reckoning pose.

online and map embedding networks

$$\boxed{P_{\mathrm{LiDAR}}} \propto s\left(\pi\left(f(\mathcal{I};\mathbf{w}_{\mathrm{O}}),\mathbf{x}\right), g(\mathcal{M};\mathbf{w}_{\mathrm{M}})\right)$$

cross-correlation

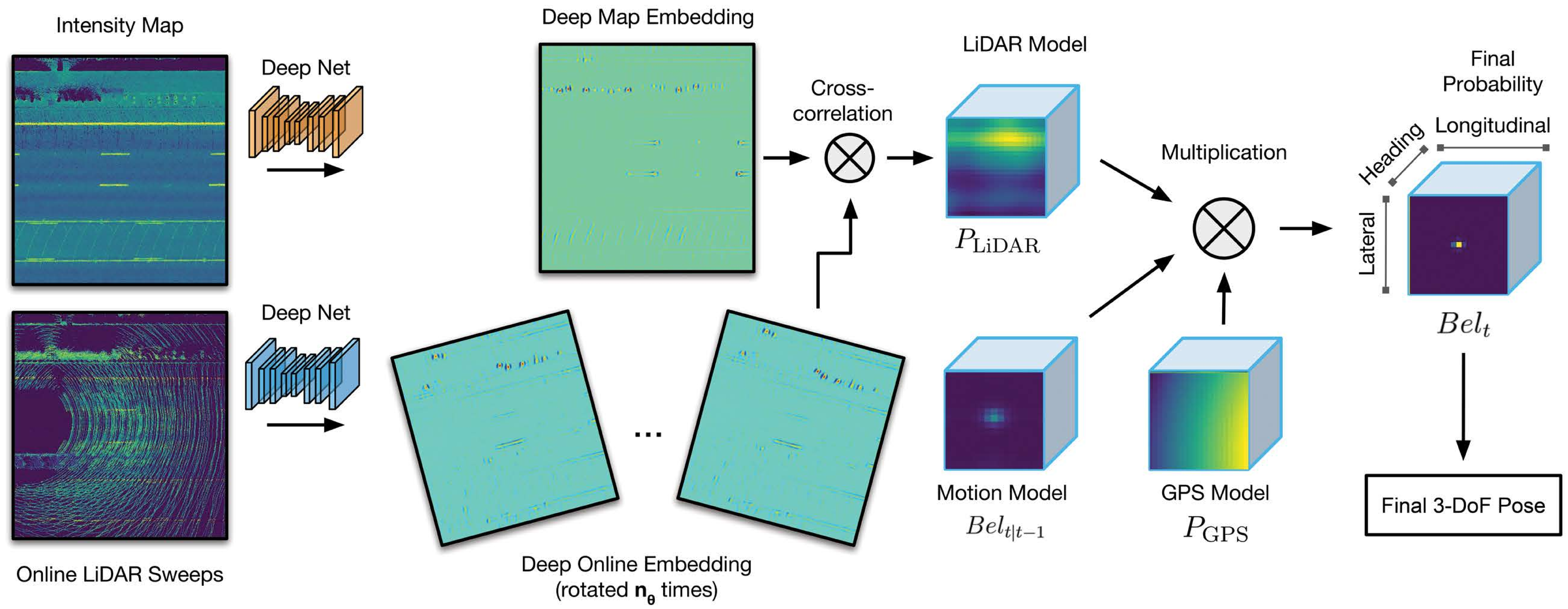
2D rigid transform

Incorporate GPS using a simple Gaussian model.

$$\boxed{P_{\mathrm{GPS}}} \propto \exp\left(-\frac{(g_x - x)^2 + (g_y - y)^2}{\sigma^2_{\mathrm{GPS}}}\right)$$

$$\boxed{\mathrm{Bel}_{t|t-1}(\mathbf{x}|\mathcal{X}_t)} = \sum_{\mathbf{x}_{t-1}} P(\mathbf{x}|\mathcal{X}_t, \mathbf{x}_{t-1})\mathrm{Bel}_{t-1}(\mathbf{x}_{t-1})$$

Motion model uses a Gaussian to model dynamics uncertainty.



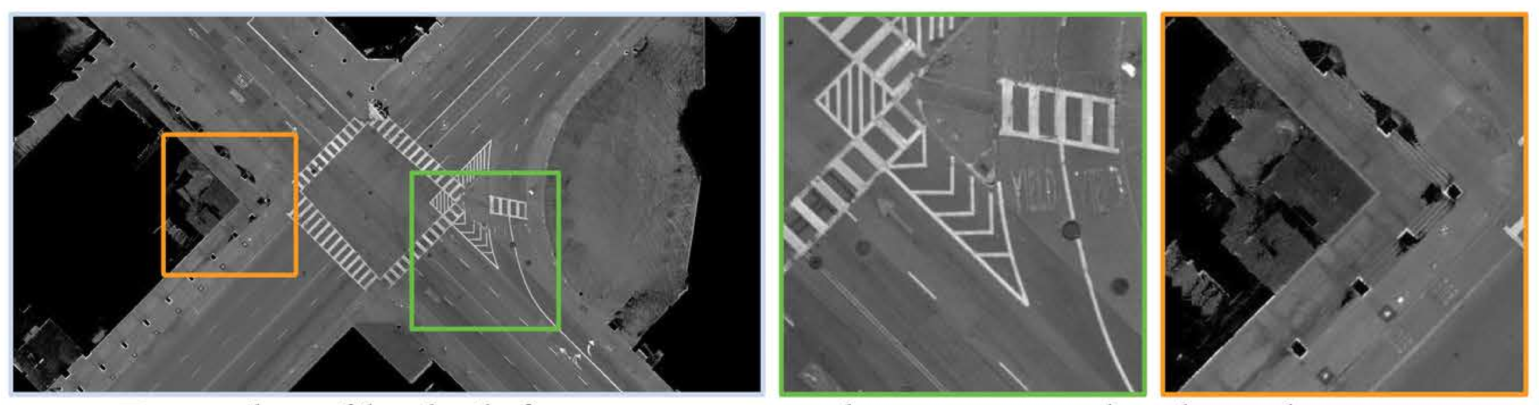
- At each time step we exhaustively search the space $\mathbf{x} = (x, y, \theta)$ around the dead reckoning pose for the best match.
- Obtain current pose from Bel_t:

$$\mathbf{x}_t^* = \frac{\sum_{\mathbf{x}} \mathrm{Bel}_t(\mathbf{x})^\alpha \cdot \mathbf{x}}{\sum_{\mathbf{x}} \mathrm{Bel}_t(\mathbf{x})^\alpha}$$

- Matching in (x, y) is equivalent to a 2D correlation, which we perform in the Fourier domain for performance reasons.

> Matching in spatial domain: 26.7ms
> Matching in Fourier domain: **1.4ms**
> ➡ Real-time system performance: **15Hz on a GPU**

- The learned component of our system is the LiDAR matching, i.e., P_{LiDAR} in the above diagram.
- The embedding nets use the LinkNet architecture.
- Embeddings are learned by backpropagating through the cross-correlation matching. We do not include the temporal filtering or GPS components at train time.
- We use a cross-entropy loss whereby the score volume corresponding to the ground truth is a one-hot encoding of the true offset between the online and the map data in a sample.



Examples of high-definition maps with centimeter-level resolution.

3 Results

- Tested on 280km of highway.
- 99th percentile error **<20cm** (lane marker = 15cm wide).

Table 1: Localization Performance on **Highway-LidarA** Dataset (Per Sequence)

Method	Motion	Prob	Median Error (cm) Lat	Lon	Total	Failure Rate (%) ≤ 100m	≤ 500m	≤ End
Dynamics	Yes	No	439.21	863.68	1216.01	0.46	98.14	100.00
Raw LiDAR	Yes	No	1245.13	590.43	1514.42	1.84	81.02	92.49
ICP	Yes	No	**1.52**	5.04	**5.44**	3.50	5.03	7.14
Ours (LinkNet)	No	No	3.87	4.99	7.76	0.35	0.35	0.72
Ours (LinkNet)	Yes	No	3.81	4.53	7.18	1.06	1.06	1.44
Ours (LinkNet)	Yes	Yes	3.00	**4.33**	6.47	**0.00**	**0.00**	**0.00**

Table 2: Localization Performance on **Misc-LidarB** trained on **Highway-LidarA** (Per Sequence)

Method	Motion	Prob	Median Error (cm) Lat	Lon	Total	Failure Rate (%) ≤ 100m	≤ 500m	≤ End
Dynamics Only	Yes	No	195.73	322.31	468.53	6.13	68.66	84.26
ICP	Yes	No	**2.57**	15.29	16.42	0.46	28.43	37.53
Ours (Transfer)	Yes	No	6.95	**6.38**	**11.73**	**0.00**	**0.71**	**1.95**