Uber ATG We are hiring! Meet us at booth **#1155**.

Motivation

- Centimeter-level localization is a key task for self-driving.
- Learning to match observations to maps shown to be highly effective.
- Detailed maps can have very demanding storage requirements.
- Goals:



Low Storage & Transfer Costs

Fast Deployment & Update Times

- Address his by **learning a compression scheme** optimal for localization by jointly learning localization and compression.



Related Work

- Learning-based Online Localization
 - Learning to Localize Using a LiDAR Intensity Map (I. A. Bârsan et al., CoRL '18, our previous work) showed it is viable to cast localization as a learnable matching task.
- ▶ L³-Net by Lu et al., 2019 presents a system which learns to match point clouds for localization in an end-to-end pipeline.
- Learning-based Image Compression
- ▶ **RNN-based** (Toderici et al, '15, '16', '17, etc.)
- **GAN-based** (Rippel & Bourdev '17, Augustsson '18)
- Task-specific compression (videos, faces, medical imagery)



Probabilistic Localization

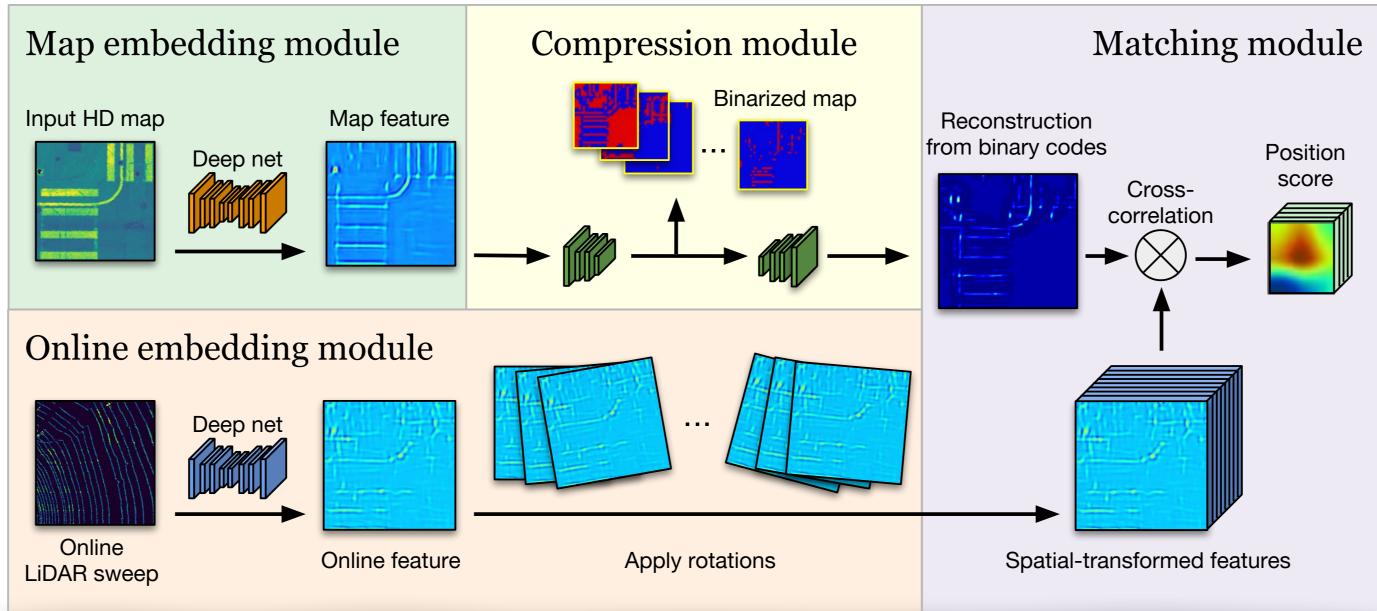
- Our goal is to perform **online localization**, and compute a centimeter-level accurate map-relative pose of the AV at every time step.
- The poses are parameterized with three degrees of freedom (x, y, yaw).
- Localization follows a standard **histogram filtering** formulation.
- We train the matching module leveraged in the **update step** of the filter.
- This 3D search space is discretized, and searched exhaustively around the predicted pose at each time.
- Predicted pose = past pose + integrated IMU & wheel encoders.

	$p(\mathbf{x}_{t+1}' \in \mathbb{R}^3)$	Predicte	ed pose at tim
 Predict with Dead Reckoning (IMU+encoders) 			ate with learn t compressec
$p(\mathbf{x}_t \in \mathbb{R}^3)$ True N Distribution of pose at time	over $p(\mathbf{x}_{t+1})$	$\in \mathbb{R}^3$)	Distribution pose at time

Learning to Localize through Compressed Binary Maps

Xinkai Wei^{1,2}*, Ioan Andrei Bârsan^{1,3}*, Shenlong Wang^{1,3}*, Julieta Martinez¹, Raquel Urtasun^{1,3} ¹⁾ Uber ATG, ²⁾ University of Waterloo, ³⁾ University of Toronto

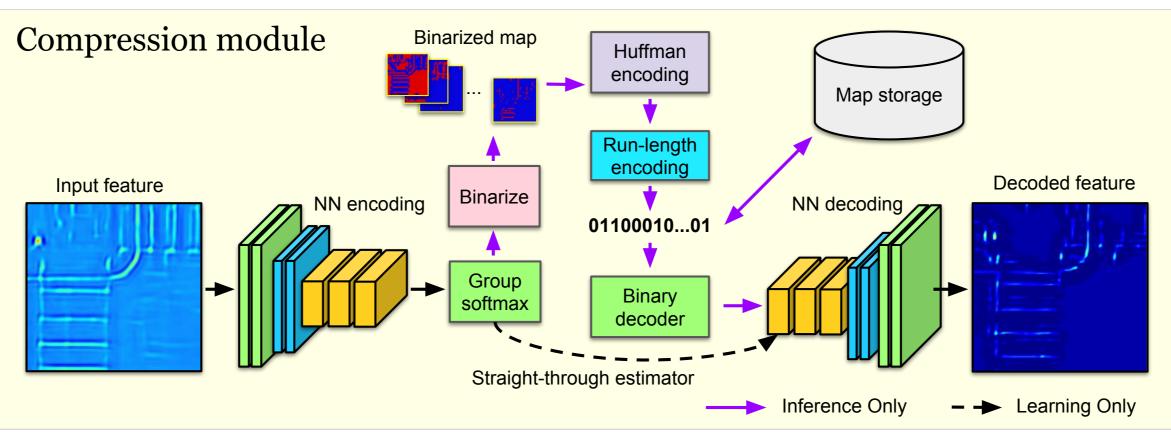
The LiDAR matching depicted below is trained to match observations to **compressed maps**, using a learned matching method.



- Note: No need to compress online observations (never stored).
- Input LiDAR and maps are all in **bird's-eye view** (2D).



- We compute feature embeddings for online data and for map data such that **matching** accuracy is maximized.
- Train with compression in the loop to reduce the map's bitrate.
- Build good sparse binary representations such that Huffman and Run-**Length Encoding** can do a very good job.



• Training to (1) maximize matching performance while (2) minimizing code length and (3) ensuring the **binarization-induced** error is minimal.

 $\ell = \ell_{\text{LOC}}(\mathbf{y}, \mathbf{y}_{\text{GT}}) + \lambda_1 \ell_{\text{CODELEN}}(\mathbf{p}) + \lambda_2 \ell_{\text{HARDBIN}}(\mathbf{p})$

(1) Localization term: Cross-entropy between predicted 3D (x, y, yaw) score map and ground truth one-hot offset.

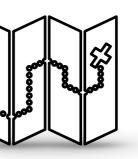
$$\ell_{\text{Loc}}(\mathbf{y}, \mathbf{y}_{\text{GT}}) = \sum_{i} y_{\text{GT}, i} \log \theta$$

(2) Entropy in the mini batch is a differentiable surrogate of code length.

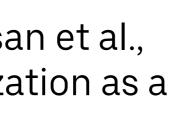
$$\ell_{\text{CODELEN}}(\mathbf{p}) = \bar{\mathbf{p}} \log \bar{\mathbf{p}}$$

(3) Minimize **per-pixel** entropy to reduce **hard binarization-induced** error.

Output of $\ell_{\text{HARDBIN}}(\mathbf{p}) = \sum_{i} p_i \log p_i$ Group Softmax



High-Accuracy Localization





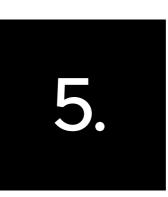
ne (t+1)

ned matching **d** map

over (t+1)

Output of Matching Pipeline

 $\bar{\mathbf{p}} = \frac{1}{W \times H \times B} \sum_{i} \mathbf{p}_{i}$



Results

- ▶ 200x better than lossless
- ▶ 20 los
- ▶ 40 ba
- ► Re Wİ stc
- ► Red

Ox better than	lossle	SS.			168 T	iR	
x better than lo	у	167.5					
sy WebP codeo	C .	-	-				
better than		ic lear	nina	age 2.5			
seline.	gener	le leal	mg	TiB of Storage		6 TiB	
	ana of	oound	- 12)	ю <u>8</u> 5			
sults enable m	-	F			0.92 TiB		
de road networ	KS LU I		Udiu	2.5			
orage.							0.28 TiE
gional maps ca	gional maps can fit in RAM.					WebP	Ours Ours
					PNG	(1%)	(Recon.) (Task)
					Storac	ge for the Full	US Road Network in TiB
Method	hod Median error (cm) Failure rate (%						Bit per pixel
	Lat	Lon	Total	$\leq 100 \text{m}$	≤ 500 m	End	
Lossless (PNG)	1.55	2.05	3.09	0.00	1.09	2.44	4.94
JPG-5	4.32	5.48	8.41	0.00	1.09	1.25	0.18
JPG-50	3.29	5.60	7.59	0.00	1.09	5.26	1.03
WebP-5		5.75	6.53	2.04	5.43		0.30
WebP-50		2.75	3.76	0.00	3.26		1.05
Ours	1.61	2.26	3.47	0.00	1.09	1.22	0.0083
Compar	ison to	non-le	earning	g baselin	es on our	urban da	taset.
Method	Method Median error (cm)					(%)	Bit per pixel
	Lat	Lon	Total	≤ 100 r			
Lossless (PNG)	1.55	2.05	3.09	0.0	0 1.0	09 2.44	4.93580
Ours (recon, $8\times$)	<u>1.59</u>	<u>2.16</u>	3.24	0.0		<u></u>	0.02689
Ours (recon, $16 \times$)	1.76	2.48	3.62	0.0	0 0.0	00 2.56	0.01155
Ours (match, $8 \times$)	1.61	2.26	3.47	0.0	0 <u>1.(</u>	<u>)9</u> 1.22	<u>0.00830</u>
Ours (match, $16 \times$)	1.62	2.77	3.84	<u>1.0</u>	$\underline{0}$ 2.1	17 4.26	0.00733
Comparis	son to I	earnin	g-base	ed baseli	nes on ou	ır urban d	ataset.
Method	N	ledian F	Err (cm		Failure l	Rate (%)	b/m ²
	X					500m Ei	
PNG, 5cm/px	1.5	5 2.0	5 3.	.09 0	.00 1	.09 2.4	44 1948.55
PNG, 10cm/px	4.3				_	3.26 4.0	
JPG@50, 10cm/p	x 4.5	1 5.7	8 8	.95 0	.00 <u>1</u>	.09 10.	.64 63.42
PNG, 15cm/px	15.7	23.	66 31	.73 10).31 2	0.65 22.	.03 173.97
IDC@50 15 cm/m	. 11 6	7 10	20 25	x 1 / Ο	70 1	2 0/ 16	28 20.00

Ox better than	lossle	SS.			г	.68 TiB		
x better than lo	Y	1	67.5					
sy WebP codeo	5	-						
% better than		ic loar	nina	age	7.5			
	yener	IIIIY	TiB of Storage			6 TiB	_	
seline.	B of	5 —						
sults enable m	Ϊ							
de road networ	road networks to fit onboard							0.92 TiB
orage.					<u> </u>			
gional maps can fit in RAM.					0			
9.0			•			PNG	WebP (1%)	Ours Ours (Recon.) (Task)
						Storage f	• •	S Road Network in TiB
	7.7.11						、 、	
Method	Media					e rate (%		Bit per pixel
	Lat	Lon	Total	≤ 100)m <	≤ 500m	End	
Lossless (PNG)	1.55	2.05	3.09	0.	00	1.09	2.44	4.94
JPG-5	4.32	5.48	8.41	0.	00	1.09	1.25	0.18
JPG-50	3.29	5.60	7.59	0.	00	1.09	5.26	1.03
WebP-5		5.75	6.53		04	5.43	13.95	0.30
WebP-50		2.75	3.76		00	3.26	3.30	1.05
Ours	1.61	2.26	3.47	0.	00	1.09	1.22	0.0083
Compar	ison to	non-le	earning	g base	lines	on our u i	r ban dat	aset.
Method	Method Median error (cm)					re rate (%	<i>(o</i>)	Bit per pixel
	Lat	Lon	Total	≤ 1	00m	$\leq 500 \mathrm{m}$	End	
Lossless (PNG)	1.55	2.05	3.09		0.00	1.09	2.44	4.93580
Ours (recon, $8 \times$)	1.59	2.16	3.24		0.00	$\frac{1.09}{1.09}$	$\overline{1.22}$	0.02689
Ours (recon, $16 \times$)	1.76	$\overline{2.48}$	3.62		0.00	$\overline{0.00}$	2.56	0.01155
Ours (match, $8 \times$)	1.61	2.26	3.47		0.00	<u>1.09</u>	1.22	0.00830
Ours (match, $16 \times$)	1.62	2.77	3.84		1.00	2.17	4.26	0.00733
Comparis	son to l	earnin	g-base	ed bas	elines	on our l	urban da	ataset.
Method	Μ	edian I	Err (cm	1)	Fa	ailure Ra	te (%)	b/m ²
	Lat Lon		n To	5 otal ≤ 100		$0m \leq 500m$		d
PNG, 5cm/px	1.5	5 2.0	5 3	.09	0.00	1.0	9 2.4	4 1948.55
PNG, 10cm/px	4.3			.50	3.19	$\frac{1.0}{3.2}$		
JPG@50, 10cm/pz				.95	0.00	1.0		
PNG, 15cm/px	15.7			.73	10.31			
$\mathbf{DC} \otimes 50 + 15 \operatorname{cm/m}$			20 25	51/	0.20	12 0	16	20.00

 Dx better than lossless. Dx better than lowest-quality say WebP codec. D% better than generic learning seline. esults enable maps of country-de road networks to fit onboard brage. egional maps can fit in RAM. 				167.5 7.5 5 2.5 2.5	F	B TiB	WebP (1%)	0.92 TiB 0.28 Ti 0.28 Ti 0urs Ours (Recon.) (Task)	iB
	N <i>K</i> 1		<u> </u>			2		S Road Network in TiB	
Method	Median Lat I		<i>,</i>	Failt < 100m		a te (%) 00m) End	Bit per pixel	
Lossless (PNG)			.09	0.00		1.09	2.44	4.94	
			.41	0.00		1.09	1.25	0.18	
			.59	0.00		1.09	5.26	1.03	
			.53	2.04		5.43	13.95	0.30	
			.76	0.00		3.26	3.30	1.05	
Ours			.47	0.00		1.09	1.22	0.0083	
Compari	ison to ı	non-lea	rning	baseline	s on	our ur	ban dat	aset.	
Method	Media	an error	(cm)	Failure rate (%)			<i>b</i>)	Bit per pixel	
	Lat	Lon	Total	$\leq 100 \mathrm{m}$	\leq	500m	End		
Lossless (PNG)	1.55	2.05	3.09	0.00		<u>1.09</u>	2.44	4.93580	
Ours (recon, $8 \times$)	<u>1.59</u>	2.16	3.24	0.00		<u>1.09</u>	1.22	0.02689	
Ours (recon, $16 \times$)	1.76	2.48	3.62	0.00		0.00	2.56	0.01155	
Ours (match, $8 \times$)	1.61	2.26	3.47	0.00		1.09	1.22	0.00830	
Ours (match, $16 \times$)	1.62	2.77	3.84	<u>1.00</u>		2.17	4.26	0.00733	
Comparison to learning-based baselines on our urban dataset.									
Method Median Err (cn				Failure Rate (%)			te (%)	b/m ²	
	Lat	Lon	To	tal ≤ 10)0m	\leq 500	m En	d	
PNG, 5cm/px	1.55	2.05	3.0)9 0.0)0	1.09) 2.4	4 1948.55	
PNG, 10cm/px	4.37	6.68			19	3.26	_		
JPG@50, 10cm/px	4.51	5.78	8.9	95 0.0)0	1.09	<u>)</u> 10.6	64 63.42	
PNG, 15cm/px	15.73	23.66	5 31.	73 10.	31	20.6	5 22.0)3 173.97	
JPG@50, 15cm/px	x 11.67	18.20	25.	14 9.2	28	13.0	4 16.2	28 <u>29.00</u>	
Ours $(16 \times)$	<u>1.76</u>	<u>2.48</u>	<u>3.6</u>	<u>62</u> 0.0)0	0.00	<u>2.5</u>	<u>6</u> 2.87	

Ablation: Error, Failure Rate and bits/m² as a function of **map resolution** (cm/px).



- definition maps: storage.
- Several avenues for future work remain, including:
- **six-degrees-of-freedom** localization.
- Learning with **mapping-in-the-loop**.



170

Conclusions & Outlook

This work addresses one of the main challenges associated with high-

We've shown that task-specific compression can improve over generalpurpose compression, allowing giant maps to be kept in-memory.

Investigating methods for compressing 3D point clouds and doing full

• End-to-end learning with the pose filter in the loop, similar to L³-Net.