DA4AD: End-to-End Deep Attention-based Visual Localization for Autonomous Driving ECCV 2020 Supplementary Material

Yao Zhou Guowei Wan Shenhua Hou Li Yu Gang Wang
Xiaofei Rui Shiyu Song*

Baidu Autonomous Driving Technology Department (ADT)
{zhouyao, wanguowei, houshenhua, yuli01, wanggang29,
ruixiaofei, songshiyu}@baidu.com

1 Implementation Details

Training We find that the joint learning of the descriptors and the attention model poses some challenges to the training process. Similar to [2], we adopt a two-stage training strategy. First, we train our feature embedding networks with all weights in the heatmap as 1.0. Then the heatmap head networks are learned given all other fixed networks. To make the training process more efficient, random noise is added to the predicted pose which is then fed to the WFM module as a better input. As we adopt a coarse-to-fine scheme in the WFM module, it is interesting to note that well-trained coarser levels can impede the training process of finer levels in the network. Therefore, we choose to add random noise to the initial poses for all different levels in the WFM module. To be more specific, we add uniformly distributed random noise of $[0 \sim 2.0]m$, $[0 \sim 0.4]m$ and $[0 \sim 0.2]m$ in $x - y$ dimension from coarse to fine, respectively, and $[0 \sim 2.0]^\circ$, $[0 \sim 0.4]^\circ$ and $[0 \sim 0.2]^\circ$ in yaw dimension, respectively. The batch size and the learning rate are set to be 1 and 0.001.

Hyperparameters We randomly pre-select $K = 4096$ points as the candidate pool in the AKS module. From coarse-to-fine, we select (128, 256, 512) and (32, 64, 128) keypoints in different resolutions of the image pyramid in the AKS module during training and mapping, respectively. From coarse-to-fine, the solution space of the cost volume in WFM module is set as $7 \times 7 \times 7$, $7 \times 7 \times 7$ and $5 \times 5 \times 5$, respectively. And the steps in (x, y, yaw) dimensions are $(0.5m, 0.5m, 0.5^\circ)$, $(0.25m, 0.25m, 0.25^\circ)$ and $(0.125m, 0.125m, 0.125^\circ)$, respectively. Therefore, the maximum affordable offset of the predicted pose is about $(0.5 \times \frac{7-1}{2} = 1.5m, 1.5m, 1.5^\circ)$, which is sufficient for our application. As we mentioned, we monitor the variance of estimated probability vectors $P(\Delta z_i), z \in \{x, y, \psi\}$ for "unavailable" localization results. For single or multi-cameras, we use $(0.4m^2, 0.4m^2, 0.4(^\circ)^2)$ or $(0.625m^2, 0.625m^2, 0.625(^\circ)^2)$. as the threshold, respectively.

Feature-based Method We adopt the original HF-Net implementation from https://github.com/ethz-asl/hfnet. With regard to the threshold of matched

* Author to whom correspondence should be addressed

inliers for an "available" status, we use 30 and 40 for single and multi-camera versions, respectively. When we build the prior map for the feature-based method, ground truth poses were used in the SfM triangulation stage as the initial values, and a BA was used to further refine the rotation parameters with the translation ones fixed. We verified the shift was no more than 1-2 cm even if we don't fix them.

2 More about the Dataset

Our vehicle platform is equipped with a Velodyne HDL-64E LiDAR, a NovAtel PwrPak7D-E1 GNSS receiver integrated with dual antennas and an Epson EG320N IMU, and three Leopard AR0231 rolling shutter cameras facing forward, rear-left, and rear-right. More importantly, the cameras are hardware synchronized with the LiDAR and compensated for the rolling shutter and vehicle motion effect, yielding precise alignment between 3D point clouds and images as shown in Figure 1. For the sake of accurate ground truth poses in challenging scenarios, for example, urban canyons, a near navigation grade IMU sensor, IMU-ISA-100C, containing fiber optic gyros and MEMS accelerometers, was also installed in the vehicle together with a NovAtel ProPak6 GNSS receiver. The ground truth poses in our experiments are provided using the above mentioned advanced sensors through a post-processing solution plus necessary LiDAR SLAM methods [4, 6, 1].

Our data was collected approximately every once a week in the northwest region of Beijing, China, close to the Daoxiang Lake park as shown in Figure 2. It is split into two non-overlapping regions A and B. Data from area A was used for training and area B was used for testing to verify the generalization capability of the network. The training data composes of 10,385 frames evenly sampled from 104,000 frames (189.7 km) to accelerate the training procedure by avoiding visually repetitive data. The testing data includes 30,495 frames over 59.4 km. The data collection time, date and usage are detailed in Table 1. More sample images are shown in Figure 3 demonstrating that the diversity of our dataset in terms of both the urban driving scenarios captured and the time spent on data collection. The frame rates of our cameras are 10hz (front) and 15hz (rear left and right) with a high image resolution of 1920×1080.

3 More Ablations on Loss and Keypoints

Loss Function In Section 4.6, we propose to use multiple loss functions together. We then take a deeper look at the contribution of each of these functions and display the results in Table 2. It is seen that only using the absolute loss already achieves acceptable results. If we incorporate the concentration or similarity loss function, both can improve the overall performance in terms of accuracy, however, the N/A ratio with the similarity loss drops a lot. Overall, we achieve the best performance when we use all the three loss functions together. Furthermore, Figure 4 shows the generated heatmaps when we apply different

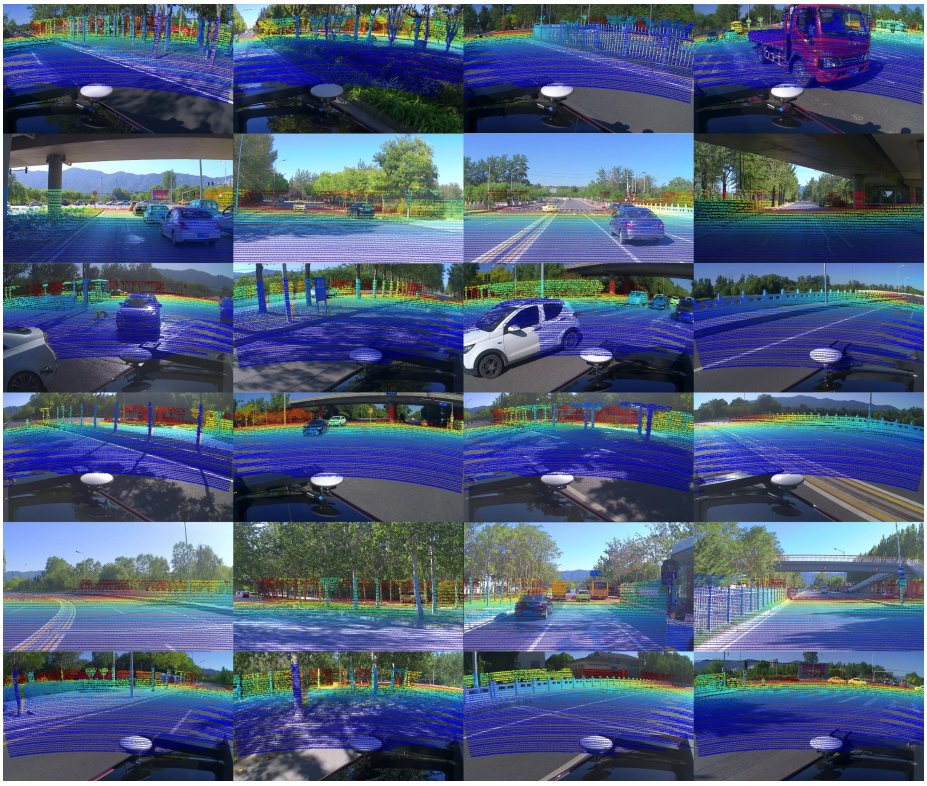

Fig. 1: The illustration of the hardware synchronized camera images and LiDAR scans. By leveraging accurate timestamp synchronization, and rolling shutter and vehicle motion compensation, we achieve precise alignment between LiDAR point cloud projections and image pixels.

Seq.	Date	Time	Training		Testing		Mapping		Description
			Dist.	Fram.	Dist.	Fram.	Dist.	Fram.	
1	Sep. 18, 2019	14:34:21-16:10:21	-	-	-	-	48.96	13607	Only for Mapping
2	Sep. 24, 2019	12:49:36-15:08:37	37.99	2229	10.95	6862	-	-	Early Autumn
3	Oct. 14, 2019	14:26:19-15:58:19	25.30	1364	5.447	2269	-	-	Early Autumn
4	Oct. 21, 2019	16:22:20-18:02:20	25.32	1521	10.95	6464	-	-	Late Autumn, Dusk
5	Oct. 25, 2019	10:48:21-12:33:21	25.17	1369	5.449	3199	-	-	Late Autumn
6	Nov. 30, 2019	11:29:08-13:22:08	25.40	1480	4.723	2178	-	-	Winter
7	Dec. 16, 2019	12:34:35-14:06:35	25.26	1168	10.91	4847	-	-	Foggy lens, Snowy
8	Dec. 25, 2019	15:36:58-17:12:58	25.25	1254	10.95	4676	-	-	Winter, Dusk

Table 1: Data collection time, date and their usage for training, mapping and testing purposes.

loss functions. As we can see, if we use the absolute loss only, the heatmaps highlight the areas that are suitable for the localization task, but a large portion of the background and dynamic objects also gain high response which is not

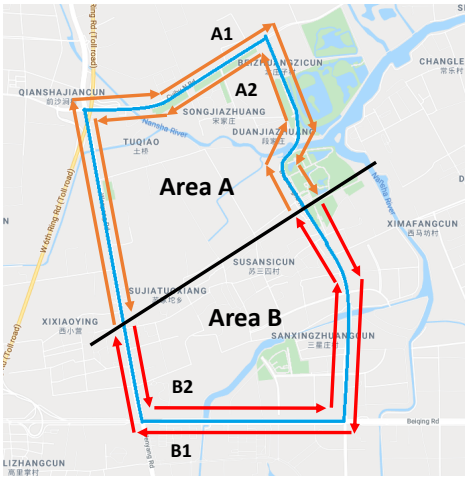

Fig. 2: The data collection route in northwest Beijing. The dataset is divided into two non-overlapping regions A and B for training and testing, respectively.

our desire. By using all the three loss functions, we achieve the heatmaps that are clean and effectively suppress the influence of the background and dynamic objects.

Method	N/A (%)	Horizontal RMS/Max(m)	Horizontal 0.1/0.2/0.3(%)	Longitudinal RMS/Max(m)	Lateral RMS/Max(m)	Yaw RMS/Max(°)	Yaw 0.1/0.3/0.6(%)
A.L.	98.8	0.066/**1.599**	82.4/95.0/99.2	0.055/**1.598**	0.024/**1.110**	0.058/2.428	87.4/99.4/99.9
A.L.+C.L.	99.9	0.064/2.022	85.2/96.0/99.5	0.050/1.967	0.028/1.186	0.054/**1.088**	**90.4**/**99.8**/99.9
A.L.+C.L.+S.L.	**100.0**	**0.058**/2.617	**86.3**/**96.8**/99.5	**0.048**/2.512	**0.023**/1.541	0.054/3.208	89.4/99.6/99.9

Table 2: Comparison using different loss functions. We denote the absolute, concentration and similarity loss as "A.L.", "C.L." and "S.L.", respectively. Overall, we achieve the best performance when we use all the three loss functions together.

Keypoints During the mapping stage, a set of keypoints are selected for different resolutions in the pyramid in the AKS module. In this section, we analyze the impact of the number of keypoints on performance. Specifically, we present the performance when using (128, 256, 512), (64, 128, 256), (32, 64, 128), (16, 32, 64), and (8, 16, 32) keypoints for different resolutions in Table 3. It is seen that the performance improves steadily as we increase the number of keypoints. After comprehensive consideration, we choose (32, 64, 128) as the default setting.

4 More Results

To help us further understand the performance comparison with other methods, we split the testing trials into groups under different circumstances as shown in

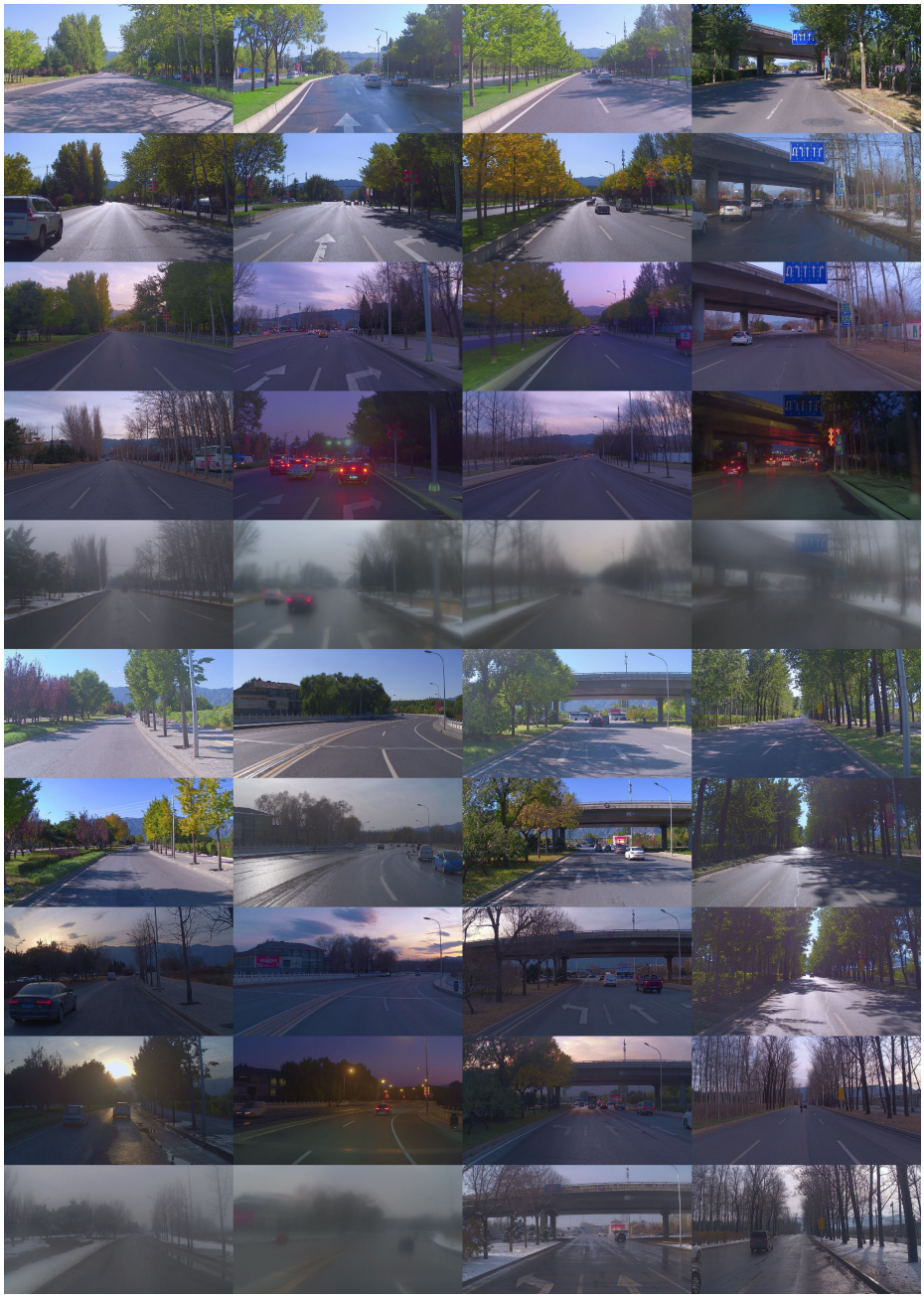

Fig. 3: Sample images from our newly collected dataset. These illustrate the different urban scenarios with varying lighting conditions, seasonal changes, and difficult circumstances, that make it a challenging dataset for the evaluation of a vision-based localization system.

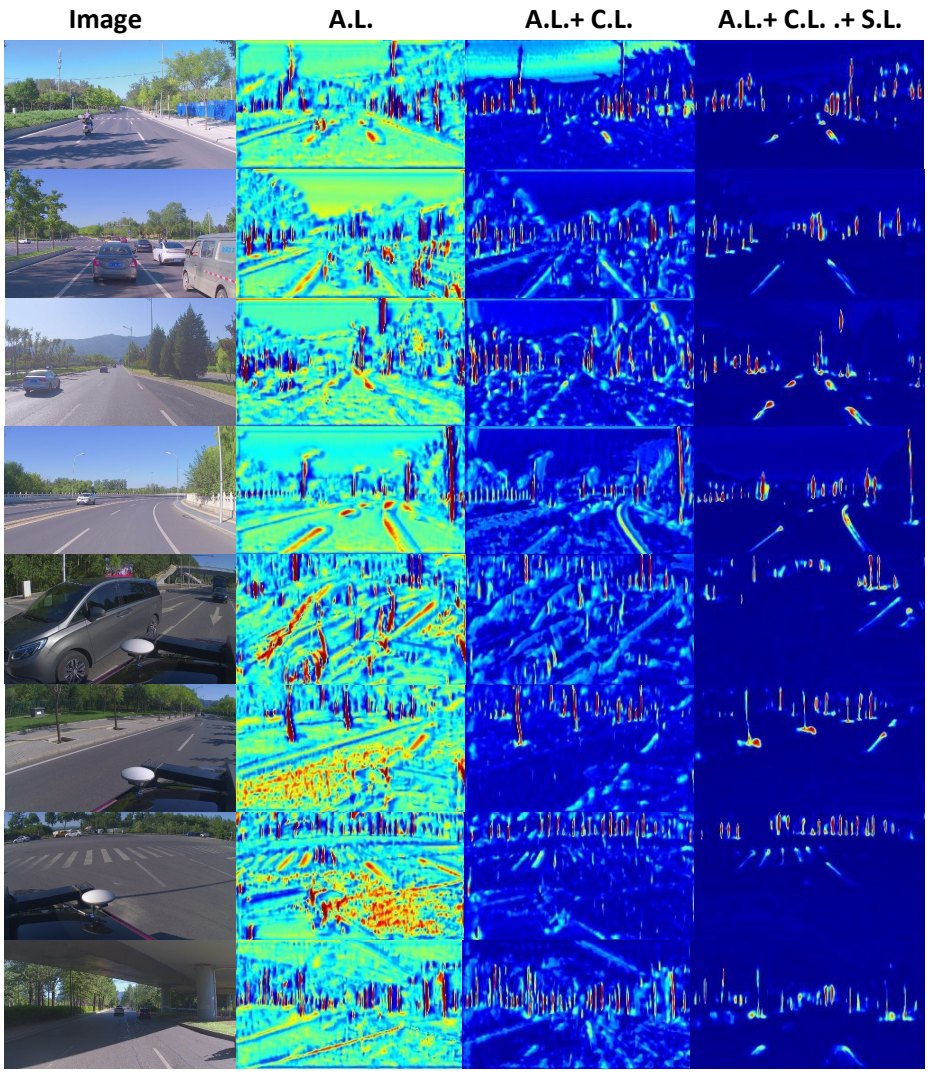

Fig. 4: Comparison of the generated heatmaps when applying different loss functions. We achieve the cleanest heatmaps that suppress the noise caused by the background and dynamic objects when we use all the three proposed loss functions.

Keypoints	N/A	Horizontal		Longitudinal	Lateral	Yaw	
	(%)	RMS/Max(m)	0.1/0.2/0.3(%)	RMS/Max(m)	RMS/Max(m)	RMS/Max(°)	0.1/0.3/0.6(%)
(128, 256, 512)	100.0	0.053/2.014	87.6/97.1/99.5	0.044/1.655	0.020/1.292	0.051/1.558	89.1/99.7/99.9
(64, 128, 256)	100.0	0.055/1.748	87.1/97.0/99.6	0.045/1.736	0.021/1.465	0.052/1.917	89.9/99.7/99.9
(32, 64, 128)	100.0	0.058/2.617	86.3/96.8/99.5	0.048/2.512	0.023/1.541	0.054/3.208	89.4/99.6/99.9
(16, 32, 64)	99.9	0.063/2.751	84.2/96.3/99.2	0.052/2.635	0.024/1.966	0.057/2.393	87.4/99.5/99.8
(8, 16, 32)	99.7	0.092/9.847	81.4/94.2/97.7	0.074/7.674	0.039/7.855	0.076/3.385	82.6/98.0/99.0

Table 3: Comparison using different keypoints. Note that increasing the number of keypoints gives better performance. (32, 64, 128) is chosen as the default setting for our system.

Table 4. As our map is built in Sep., the performance of our system is comparable in all cases, for example, autumn, winter or foggy lens, except for difficult lighting conditions (dusk) which are well-known difficult problems for image based methods. But we can still achieve reasonably good performance at dusk.

We made some modifications to the original implementation of the HF-Net [3] method. In this section, we further analyze the contribution of each of them as shown in Table 5. "HFNet[3] (S)" is the original HF-Net implementation using PnP + RANSAC with single camera. For the "HFNet" method, we only extend the original implementation to leverage multi-camera input through a local bundle adjustment. Next, for "HFNet+", we furnish it with the predicted poses instead of using the embedded global features in it. Finally, "HFNet++" is the method that estimates 3 DoF poses with all the three modifications.

References

1. Droeschel, D., Behnke, S.: Efficient continuous-time SLAM for 3D LiDAR-based online mapping. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). pp. 1–9. IEEE (2018)
2. Noh, H., Araujo, A., Sim, J., Weyand, T., Han, B.: Large-scale image retrieval with attentive deep local features. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 3456–3465 (2017)
3. Sarlin, P.E., Cadena, C., Siegwart, R., Dymczyk, M.: From coarse to fine: Robust hierarchical localization at large scale. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
4. Shiratori, T., Berclaz, J., Harville, M., Shah, C., Li, T., Matsushita, Y., Shiller, S.: Efficient large-scale point cloud registration using loop closures. In: Proceedings of the International Conference on 3D Vision (3DV). pp. 232–240. IEEE (2015)
5. Wan, G., Yang, X., Cai, R., Li, H., Zhou, Y., Wang, H., Song, S.: Robust and precise vehicle localization based on multi-sensor fusion in diverse city scenes. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). pp. 4670–4677 (2018)
6. Yang, S., Zhu, X., Nian, X., Feng, L., Qu, X., Mal, T.: A robust pose graph approach for city scale LiDAR mapping. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 1175–1182. IEEE (2018)

	Method	N/A (%)	Horizontal		Longitudinal	Lateral	Yaw	
			RMS/Max(m)	0.1/0.2/0.3(%)	RMS/Max(m)	RMS/Max(m)	RMS/Max(°)	0.1/0.3/0.6(%)
Early Autumn	Struct-based (S)	88.0	0.224/1.407	26.6/53.6/74.8	0.203/1.388	0.065/1.312	0.152/2.129	37.8/92.0/99.3
	HFNet[3] (S)	87.5	0.224/6.494	34.6/60.7/76.1	0.192/4.714	0.082/6.407	0.080/15.65	78.2/97.8/99.8
	HFNet++(S)	97.7	0.212/6.049	32.1/59.2/77.0	0.182/2.138	0.078/6.004	0.082/16.03	75.2/98.0/99.8
	HFNet++SIFT(S)	74.4	0.211/8.181	37.8/64.9/80.6	0.168/7.349	0.093/8.154	0.089/17.42	81.0/97.3/99.2
	HFNet++	99.2	0.139/6.049	47.0/80.3/94.2	0.124/2.736	0.039/6.004	0.062/16.03	87.9/99.5/99.8
	HFNet++SIFT	84.3	0.194/8.281	35.2/72.8/85.4	0.156/7.349	0.079/7.046	0.072/13.97	87.0/98.0/99.2
	Ours (S)	93.9	0.104/2.012	68.3/88.9/94.1	0.087/1.431	0.042/1.787	0.063/1.685	83.4/97.6/99.7
	Ours	100.0	0.048/2.617	95.2/98.9/99.4	0.036/2.512	0.024/1.541	0.055/3.208	91.2/99.4/99.8
	LiDAR [5]	100.0	0.049/0.166	95.6/100.0/100.0	0.034/0.161	0.028/0.153	0.061/0.599	83.8/99.9/100.0
Late Autumn	Struct-based (S)	98.3	0.248/1.528	16.5/44.2/70.0	0.222/1.363	0.075/0.971	0.137/1.141	46.7/88.8/99.5
	HFNet[3] (S)	73.8	0.203/2.970	36.2/64.9/78.6	0.180/2.772	0.068/1.097	0.070/0.969	78.6/98.9/99.9
	HFNet++(S)	96.7	0.208/1.832	29.8/59.4/76.5	0.185/1.660	0.068/1.190	0.075/0.880	74.1/99.1/100.0
	HFNet++SIFT(S)	51.4	0.310/5.184	16.0/42.2/59.9	0.258/4.089	0.118/3.347	0.100/5.369	70.0/95.6/99.4
	HFNet++	98.3	0.145/10.18	54.1/77.4/84.6	0.134/9.342	0.033/4.033	0.052/1.222	89.3/99.6/99.9
	HFNet++SIFT	58.4	0.283/6.463	23.9/49.0/65.5	0.230/6.334	0.113/3.347	0.108/14.68	71.3/96.3/99.0
	Ours (S)	96.9	0.138/1.625	59.7/78.9/88.4	0.117/1.623	0.050/1.406	0.078/1.334	77.0/96.1/99.2
	Ours	100.0	0.043/1.648	93.9/99.2/99.7	0.034/1.157	0.020/1.418	0.054/1.441	88.0/99.7/99.9
	LiDAR [5]	100.0	0.050/0.180	96.9/100.0/100.0	0.037/0.172	0.026/0.179	0.059/0.327	83.4/100.0/100.0
Winter	Struct-based (S)	94.8	0.255/2.046	19.5/49.5/68.7	0.232/1.350	0.074/1.800	0.136/1.532	44.8/90.5/99.7
	HFNet[3] (S)	48.7	0.227/4.382	43.4/69.4/81.9	0.166/4.211	0.116/3.411	0.118/7.564	73.4/94.6/97.7
	HFNet++(S)	76.9	0.193/3.289	36.8/67.5/82.5	0.164/2.341	0.077/3.233	0.085/6.944	72.3/97.4/99.6
	HFNet++SIFT(S)	9.5	0.402/3.626	23.3/47.7/62.2	0.258/2.053	0.235/3.536	0.347/10.28	63.4/87.2/93.1
	HFNet++	95.1	0.176/10.74	45.6/75.7/89.0	0.150/10.71	0.064/5.549	0.091/15.08	77.2/97.2/99.6
	HFNet++SIFT	14.8	0.391/6.424	21.4/49.6/65.5	0.243/6.410	0.238/3.796	0.356/10.28	54.7/82.7/90.2
	Ours (S)	96.9	0.083/2.187	79.3/93.4/96.7	0.067/2.184	0.035/1.738	0.061/1.575	84.3/98.2/99.3
	Ours	99.9	0.041/1.158	97.6/99.6/99.8	0.028/1.147	0.024/0.321	0.043/0.643	96.4/99.8/100.0
	LiDAR [5]	100.0	0.053/0.220	93.6/100.0/100.0	0.037/0.202	0.030/0.145	0.081/0.401	67.0/100.0/100.0
Dusk	Struct-based (S)	93.9	0.226/2.669	21.7/55.1/75.8	0.200/0.170	0.0709/2.388	0.153/4.218	45.7/91.0/98.6
	HFNet[3] (S)	61.0	0.254/13.09	25.1/53.3/71.3	0.231/13.07	0.069/3.150	0.069/7.035	80.3/98.5/99.8
	HFNet++(S)	87.4	0.232/4.080	23.7/52.3/72.6	0.206/2.057	0.071/4.079	0.072/9.317	77.3/98.8/100.0
	HFNet++SIFT(S)	34.6	0.339/5.995	16.0/37.6/57.9	0.268/2.601	0.152/5.646	0.130/7.730	64.8/94.4/98.1
	HFNet++	97.2	0.183/3.669	34.4/62.9/83.8	0.171/3.610	0.039/1.373	0.061/1.878	85.1/99.1/99.9
	HFNet++SIFT	41.9	0.293/5.995	26.9/45.9/63.9	0.237/2.415	0.123/5.646	0.113/7.730	70.1/95.5/98.4
	Ours (S)	95.5	0.175/2.875	40.4/68.3/86.2	0.162/2.871	0.046/1.259	0.072/1.657	79.7/97.6/99.4
	Ours	100.0	0.123/0.696	42.4/84.8/98.8	0.118/0.648	0.021/0.640	0.052/0.987	91.2/99.5/99.9
	LiDAR [5]	100.0	0.056/0.333	90.0/99.7/100.0	0.044/0.252	0.026/0.262	0.080/0.559	68.4/99.9/100.0
Foggy Lens, Snowy	Struct-based (S)	80.7	0.282/2.592	15.4/43.9/65.0	0.241/1.509	0.103/2.533	0.185/2.677	35.9/84.6/97.4
	HFNet[3] (S)	6.4	1.365/322.6	23.7/43.3/55.8	1.301/322.5	0.173/8.844	0.135/3.437	57.7/93.6/98.4
	HFNet++(S)	16.6	0.254/2.103	27.3/54.5/70.2	0.217/1.277	0.099/1.727	0.096/0.787	64.8/96.0/99.8
	HFNet++SIFT(S)	-	-	-	-	-	-	-
	HFNet++	68.4	0.333/13.62	39.2/63.5/74.9	0.245/13.54	0.169/6.380	0.173/25.22	57.5/91.1/97.0
	HFNet++SIFT	7.5	0.293/5.913	36.4/63.6/75.8	0.146/4.149	0.214/5.434	0.343/6.934	35.3/75.2/86.5
	Ours (S)	94.5	0.118/3.119	58.6/87.3/94.4	0.103/3.074	0.041/1.110	0.074/1.421	77.5/97.7/99.6
	Ours	100.0	0.047/0.659	94.2/99.4/99.8	0.035/0.649	0.024/0.606	0.062/0.920	80.0/99.9/100.0
	LiDAR [5]	100.0	0.067/1.831	88.5/97.7/99.1	0.046/1.446	0.039/1.184	0.082/1.608	71.1/98.8/99.7

Table 4: Comparison under different circumstances. The performance of our system is comparable in all cases, for example, autumn, winter or foggy lens, except for difficult lighting conditions (dusk). We can still achieve reasonably good performance at dusk.

Method	N/A (%)	Horizontal		Longitudinal	Lateral	Yaw	
		RMS/Max(m)	0.1/0.2/0.3(%)	RMS/Max(m)	RMS/Max(m)	RMS/Max(°)	0.1/0.3/0.6(%)
HFNet[3] (S)	61.4	0.243/322.6	34.3/61.4/76.3	0.211/322.5	0.081/8.844	0.081/15.65	77.8/97.8/99.6
HFNet	79.2	0.374/541.1	43.5/71.9/85.4	0.320/528.7	0.118/133.1	0.156/179.5	78.7/97.5/99.2
HFNet+	93.3	0.176/14.02	45.8/74.3/87.0	0.153/13.96	0.055/6.556	0.075/25.27	83.3/98.2/99.5
HFNet++	93.2	0.176/13.62	45.4/73.9/87.0	0.152/13.54	0.056/6.380	0.077/25.22	82.6/98.2/99.5

Table 5: Comparison with different HF-Net modifications. Note that our modifications to its original implementation improve the performance significantly.