Supplementary Material for DELTAS: Depth Estimation by Learning Triangulation And densification of Sparse points

Ayan Sinha[1], Zak Murez[1], James Bartolozzi*, Vijay Badrinarayanan[2*], and Andrew Rabinovich[3*]

[1] Magic Leap Inc., CA, USA {asinha,zmurez}@magicleap.com
*Work done at Magic Leap bartolozzij@gmail.com
[2] Wayve.ai, London, UK vijay@wayve.ai
[3] InsideIQ Inc., CA, USA andrew@insideiq.team

1 Ablation Studies

We first analyze the sparse triangulated points output from our network and perform ablation studies on our critical hyper-parameters and its influence on the depth estimation performance of our trained model, i.e., we only investigate the parameter values at inference time.

Sparse Depth Analysis: We first validate the need to triangulate points in a differentiable manner as opposed to directly using sparse points output by a standard SLAM systems for the task of dense depth estimation. Table 1 lists the performance of sparse and dense depth estimations using COLMAP. We see that the sparse depth is of very poor quality, and the dense depth calculated by COLMAP is able to reduce the performance gap due to additional post processing steps like block matching etc. However, the best performance is that of using sparse map predicted by COLMAP as input to a sparse-to-dense depth estimation network, illustrating the power of deep networks. The sparse-to-dense network is identical to the network structure described in the main manuscript minus the triangulation module, and the descriptor and detector heads. Note that the performance of sparse-to-dense network is significantly worse than that of our approach end-to-end approach described in the main manuscript. Table 2 shows the performance of the sparse depth output by the triangulation module. We see that the performance is significantly better than that of sparse points output by COLMAP, and robust accross different ratios of interest points and random points. This indicates that the network learns context around a point to circumvent the hardness of triangulating non-interest points.

Number of Points: We first study the influence of the number of sampled points in the target image on the final depth estimation. In Table 3 we see that the performance of our approach is fairly robust in the range of 256 to 512 points. Performance slightly degrades for more than 512 points. Unsurprisingly, the performance significantly degrades when no triangulated points are considered

for depth estimation which would be equivalent to monocular depth estimation. However,even as few as 32 points greatly improves the performance of depth estimation. If we were to swap the depth of triangulated points with ground truth depth, we see that the performance is significantly better. Consequently, our method can be used in conjunction with an active sensor when available without requiring any retraining of the networks. A hybrid system consisting of an active sensor and our passive sensing approach is useful towards reducing the frame rate of the active sensor, and hence, reducing the power consumption.

Ratio of points: We investigated the influence of the ratio of the number of interest points from the interest point detector to the total number of points which are a combination of those detected by the detector and points sampled randomly from the image. For e.g., 0.75 indicates $3/4^{th}$ points sampled from the detector and the rest chosen randomly. We see in Table 4 that the performance of our approach is robust across all ratios. This indicates that the network is not biased towards corner points, but can robustly match points across the image.

NMS Radius: Next we investigate the influence of the non-maximum suppression (NMS) radius value for the interest point detector on the performance. Note that small values of NMS result in interest points being sampled predominantly from high texture regions and being clustered together, whereas high values of NMS encourage the points to be well distributed. In Table 5 we see that small values of NMS hurt performance, with the performance improving till NMS value of 9 and then again degrading for value of 11. This indicates that the network prefers well separated, uniformly sampled points across the image.

Threshold: We also investigated the performance of depth estimation for different thresholds on the interest point detector. In Table 6 we see that threshold values of 0.0001 and 0.0005 result in similar performance. The performance degrades for higher values of 0.001 and 0.005. This suggests that the network does not particularly favour high quality interest points, but a large number of them, which are made available when the threshold is low.

Epipolar Length: In Table 7 we investigate the influence of the length of the sampled descriptors along the epipolar line on depth estimation. We see that the performance is robust across all values of length ranging from 25 pixels to 150 pixels. This observation can further reduce the training time and inference time for depth estimation.

Offset value: We investigated the performance of our trained network for 1 pixel and 2 pixel offsets to compensate for pose error. We see in Table 8 that 2 pixel offset does not improve performance, suggesting that the pose in ScanNet is sufficiently reliable. This parameter however might be of greater influence in cases wherein pose estimation is unreliable.

Table 1. Performance of depth estimation on ScanNet using COLMAP. Sparse refers to the sparse map predicted by COLMAP, Dense refers to the dense depth map predicted by COLMAP, and Sparse +DNN refers to densification of the sparse map predicted by COLMAP using a deep neural network.

Approach	Abs Rel	Abs	Sq Rel	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Sparse	0.2629	0.4618	0.3882	0.5713	0.7498	0.8322
Dense	0.1371	0.2643	0.1379	0.8344	0.9080	0.9383
Sparse + DNN	**0.1242**	**0.1990**	**0.0658**	**0.8756**	**0.9649**	**0.9878**

Table 2. Performance of sparse depth estimation on ScanNet for different ratios of interest points and random points. We use sequences of length 3 and sample every 20 frames.

Ratio	Abs Rel	Abs	Sq Rel	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
0.0	0.0993	0.1899	0.0503	0.8856	0.9701	**0.9906**
0.25	**0.0986**	**0.1891**	**0.0502**	**0.8869**	**0.9703**	**0.9906**
0.5	0.0988	0.1893	0.0503	0.8866	0.9702	0.9905
0.75	0.0988	0.1893	0.0503	0.8866	0.9702	0.9905
1.0	0.0988	0.1893	0.0503	0.8866	0.9702	0.9905

Model Architecture: Finally, we explore the performance of our approach on model architecture. We swap our ResNet-50 backbone with a VGG-9 backbone similar to that of SuperPoint. We use the same training procedure as that of the ResNet-50 architecture mentioned in the main manuscript. In Table 9 we that the extremely light-weight VGG-9 architecture performs much better than MVDepthNet and some values are comparable or even better than those of DPSNet. Furthermore, the total number of GMACs is only 16.9, which is $\approx$ 18x more efficient that DPSNet and 8x more efficient that real-time MVDepthNet. In Table 10 we see that we observe only a slight degradation in pose performance (rotation and translation) compared to SuperPoint. This reinforces our conclusion in the main manuscript that our supervision can complement that of SuperPoint. Overall, the robust performance of our network with extremely low compute is a promising first step to derive scaling laws as done in EfficientNet.

Qualitative results: In Figure 1 we see that our depth maps are more consistent with respect to ground truth, and respect the geometry of the scene better. For e.g., the lamp in the second row, the chair at the back in the fourth row, the phone in the seventh row and the cabinet in the eight row are qualitatively better than all other methods. Furthermore, we are also able to coherently reconstruct depth where the active depth sensor fails, for e.g. the windows in the second and seventh row and the transparent glass side-table in the sixth row.

Fig. 1. Qualitative Performance of our networks on sampled images from ScanNet.

Table 3. Performance of depth estimation on ScanNet for different number of sparse points. We use sequences of length 3 and sample every 20 frames.

Num Points	Abs Rel	Abs	Sq Rel	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
0	0.2203	0.3049	0.1375	0.7198	0.9022	0.9650
32	0.1105	0.1793	0.0582	0.9002	0.9722	0.9886
128	0.0960	0.1591	0.0514	0.9232	0.9760	**0.9895**
256	0.0934	0.1550	**0.0505**	0.9276	0.9766	**0.9895**
384	**0.0931**	0.1541	**0.0505**	0.9285	**0.9767**	0.9894
512	0.0932	**0.1540**	0.0506	**0.9287**	**0.9767**	0.9893
640	0.0936	0.1543	0.0509	0.9285	0.9766	0.9892
768	0.0942	0.1549	0.0512	0.9282	0.9766	0.9891
512 (GT)	**0.0680**	**0.1111**	**0.0406**	**0.9562**	**0.9800**	**0.9903**

Table 4. Performance of depth estimation on ScanNet for different ratios of interest points and random points. We use sequences of length 3 and sample every 20 frames.

Ratio	Abs Rel	Abs	Sq Rel	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
0.0	0.0935	0.1544	0.0508	0.9283	0.9766	**0.9893**
0.25	0.0933	**0.1540**	0.0507	0.9286	0.9766	**0.9893**
0.5	**0.0932**	**0.1540**	**0.0506**	**0.9287**	**0.9767**	**0.9893**
0.75	0.0933	**0.1540**	0.0507	0.9286	0.9766	**0.9893**
1.0	0.0933	**0.1540**	0.0507	0.9286	0.9766	**0.9893**

Table 5. Performance of depth estimation on ScanNet for different radius for non-maximum suppression (NMS Rad). We use sequences of length 3 and sample every 20 frames.

NMS Rad	Abs Rel	Abs	Sq Rel	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
3	0.0942	0.1554	0.0512	0.9278	0.9765	0.9892
5	0.0937	0.1545	0.0508	0.9283	0.9766	0.9892
7	0.0937	0.1545	0.0510	0.9284	0.9766	0.9892
9	**0.0932**	**0.1540**	**0.0506**	**0.9287**	**0.9767**	**0.9893**
11	0.0938	0.1546	0.0511	0.9285	0.9766	0.9891

Table 6. Performance of depth estimation on ScanNet for different thresholds for the detector. We use sequences of length 3 and sample every 20 frames.

Thresh	Abs Rel	Abs	Sq Rel	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
0.0001	**0.0932**	**0.1539**	**0.0506**	**0.9287**	**0.9767**	**0.9893**
0.0005	**0.0932**	0.1540	**0.0506**	**0.9287**	**0.9767**	**0.9893**
0.001	0.0933	0.1540	0.0507	0.9286	**0.9767**	**0.9893**
0.005	0.0934	0.1544	0.0507	0.9283	0.9766	**0.9893**

Table 7. Performance of depth estimation on ScanNet for different lengths of the sampled descriptors. We use sequences of length 3 and sample every 20 frames.

Length	Abs Rel	Abs	Sq Rel	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
25	0.0934	0.1542	0.0508	**0.9287**	**0.9767**	**0.9893**
50	0.0933	**0.1540**	0.0507	**0.9287**	**0.9767**	**0.9893**
100	**0.0932**	**0.1540**	**0.0506**	**0.9287**	**0.9767**	**0.9893**
150	**0.0932**	**0.1540**	**0.0506**	0.9286	**0.9767**	**0.9893**

Table 8. Performance of depth estimation on ScanNet for different sampling offsets. We use sequences of length 3 and sample every 20 frames.

Offsets	Abs Rel	Abs	Sq Rel	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
1 pix	**0.0932**	**0.1540**	**0.0506**	**0.9287**	**0.9767**	**0.9893**
2 pix	0.0933	0.1541	0.0507	0.9285	0.9766	**0.9893**

Table 9. Performance of depth estimation on ScanNet for different architectures. We use sequences of length 3 and sample every 20 frames.

Arch	Abs Rel	Abs	Sq Rel	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$	GMACs
MVDepth	0.1054	0.1911	0.0970	0.8952	0.9707	0.9895	134.8
DPS	0.1025	0.1675	0.0574	0.9102	0.9708	0.9872	295.6
VGG-9	0.1073	0.1815	0.0581	0.9023	0.9719	0.9890	**16.9**
ResNet-50	**0.0932**	**0.1540**	**0.0506**	**0.9287**	**0.9767**	**0.9893**	84.4

Table 10. Performance of different descriptors on ScanNet.

	MLE	**MScore**	**Num**	**Rep**	**rot@5°**	**trans@5cm**
SuperPoint	**2.545**	**0.375**	129	0.519	0.489	0.244
VGG-9	3.057	0.325	1619	**0.751**	0.472	0.228
ResNet-50	3.101	0.329	1511	0.738	**0.518**	**0.254**