Supplementary Material

1 Running Parameters

As mentioned in Section-VII, the configuration parameters for ORB-SLAM2 is adjusted to reduce the tracking failures as much as possible for each dataset. We mainly lower the threshold for the extraction of ORB features in the provided yaml files (the number of ORB features extracted per image is set as 1000 for all sequences), and through the experiments the specific parameters are set as follows:

1) For all sequences on ICL-NUIM datasets, we set the initial threshold as 15 (default: 20), and the minimal threshold as 5 (default: 7).

2) For the *fr3_s_nt_near* and *fr3_s_nt_far* sequences from TUM RGB-D datasets, we set the initial threshold as 12 and the minimal threshold as 3. In addition, we reduce the minimal number of features required for creating an initial sparse feature map from 500 to 200, otherwise the tracking would fail due to the delayed map initialization.

3) For the sub-selected ETH SLAM sequences, we use the value 12 for the initial threshold and 3 for the minimal threshold.

2 Runtime Statistics

Table 1: Average of the median and mean time in [ms] for tracking and mapping on benchmarks datasets. Since our approach is built on top of ORB-SLAM2, we thus take its results (in VO setting) as a baseline. Only the sequences that both methods successfully track are used to computing the average (e.g., *ceiling_2* on ETH_SLAM dataset is excluded). The relative increase of our method compared to ORB SLAM is presented at the bottom.

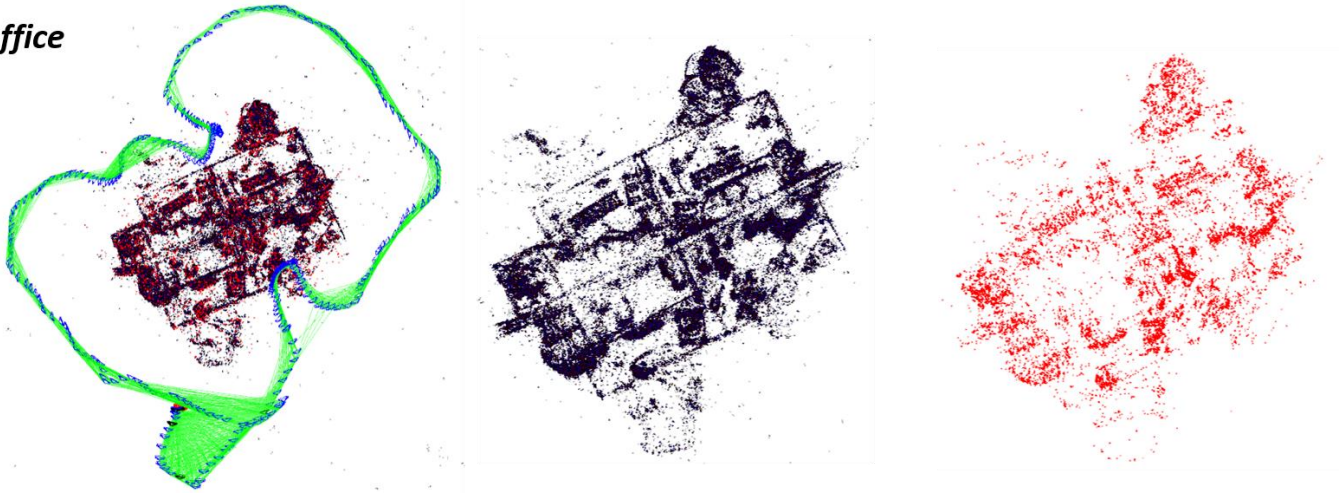
	ICL-NUIM				TUM RGB-D				ETH SLAM			
	Tracking		Mapping		Tracking		Mapping		Tracking		Mapping	
	med.	mean	med.	mean	med.	mean	med.	mean	med.	mean	med.	mean
ORB-VO	20.6	21.2	151	153	25	25.2	222	200	27.6	28.1	172	175
Ours	31.2	32.2	418	402	35.8	36.6	398	396	37.1	38.3	188	220
real_inc.	51.4%	51.8%	176%	162%	43.2%	45.2%	79.2%	98%	34.4%	36.2%	9.3%	25.7%

Although an obvious increase on tracking time can be observed from our method, we think that the absolute tracking time (approx. 35ms in average) can still meet the requirements of some near real-time tasks. On the other hand, our back-end mapping module no surprisingly cost much more time than the original implementation in ORB-SLAM2, except on the ETH SLAM dataset.

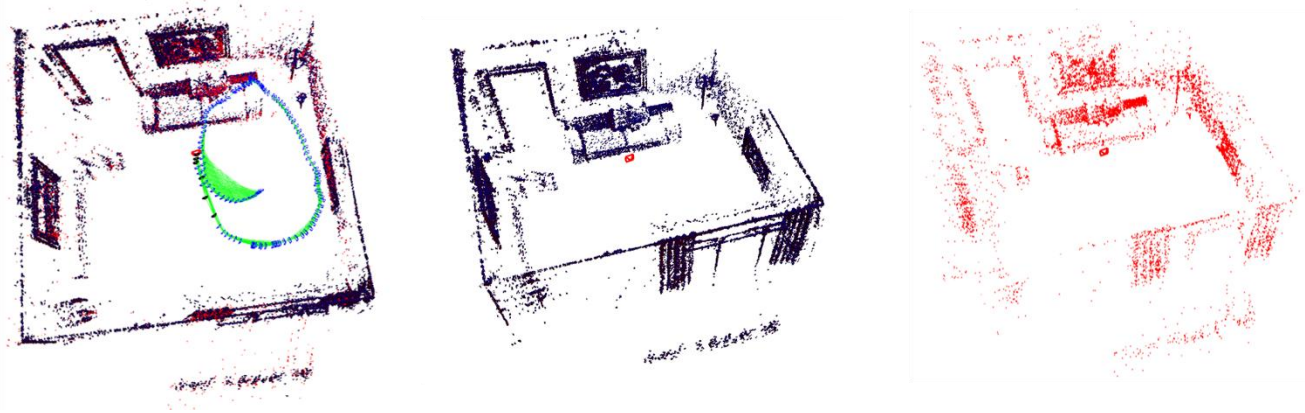
3 Qualitative Examples

The following gives several exampled scenes reconstructed using our VO approach, and from left to right are hybrid maps along with the estimated camera trajectory, the semi-dense point cloud, and the feature-based point cloud. Note that the generated semi-dense geometry matches well with the sparse feature map across synthetic and real sequences (see the zoom-in view).

fr3_office



living_t3_noise



office_t2_noise

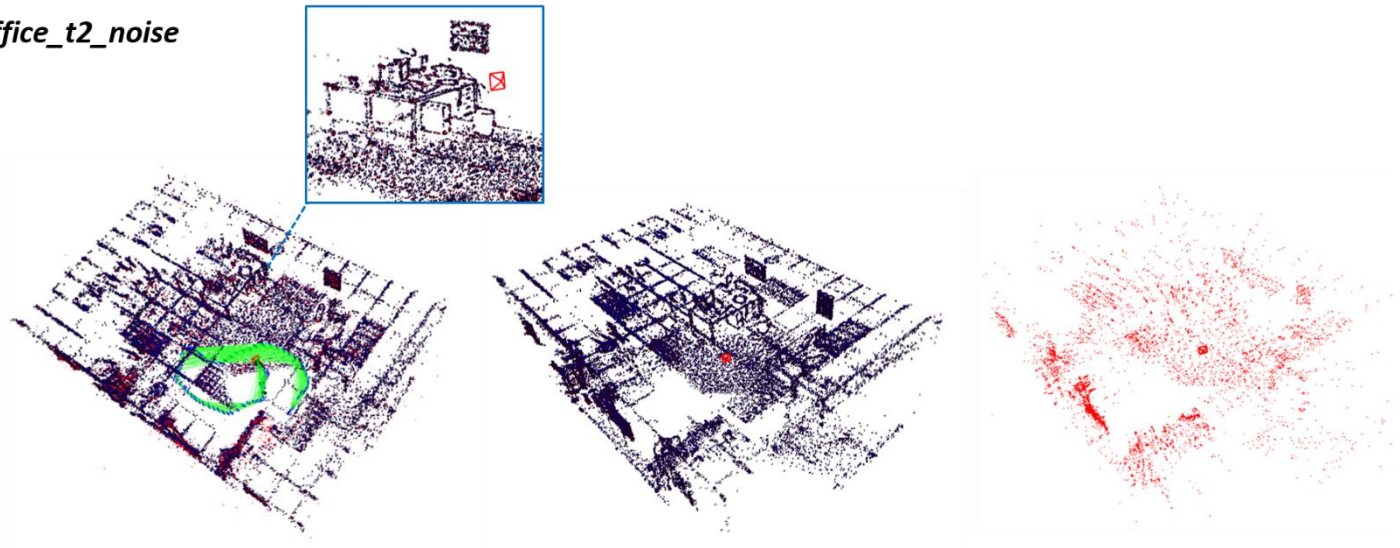
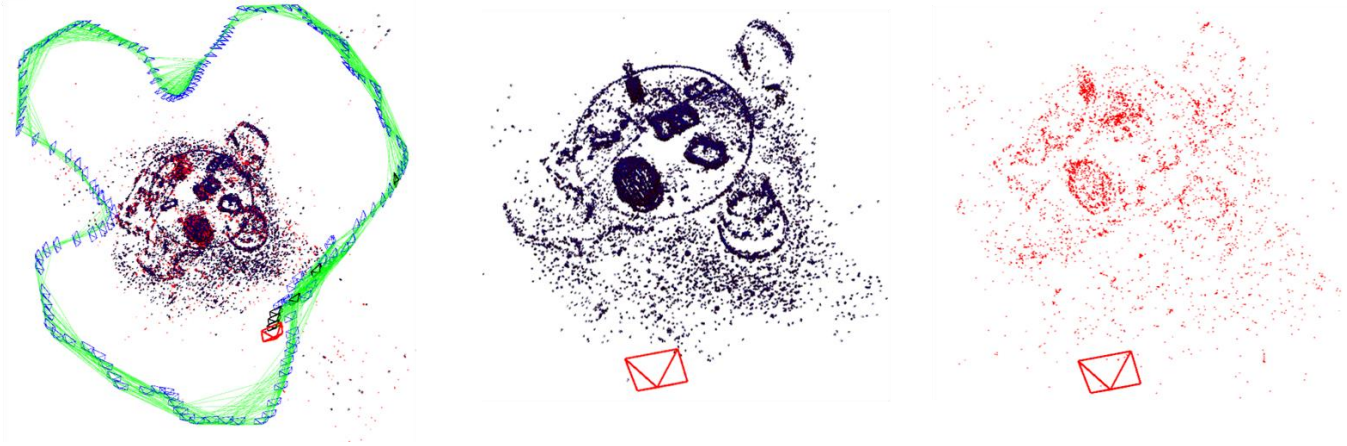
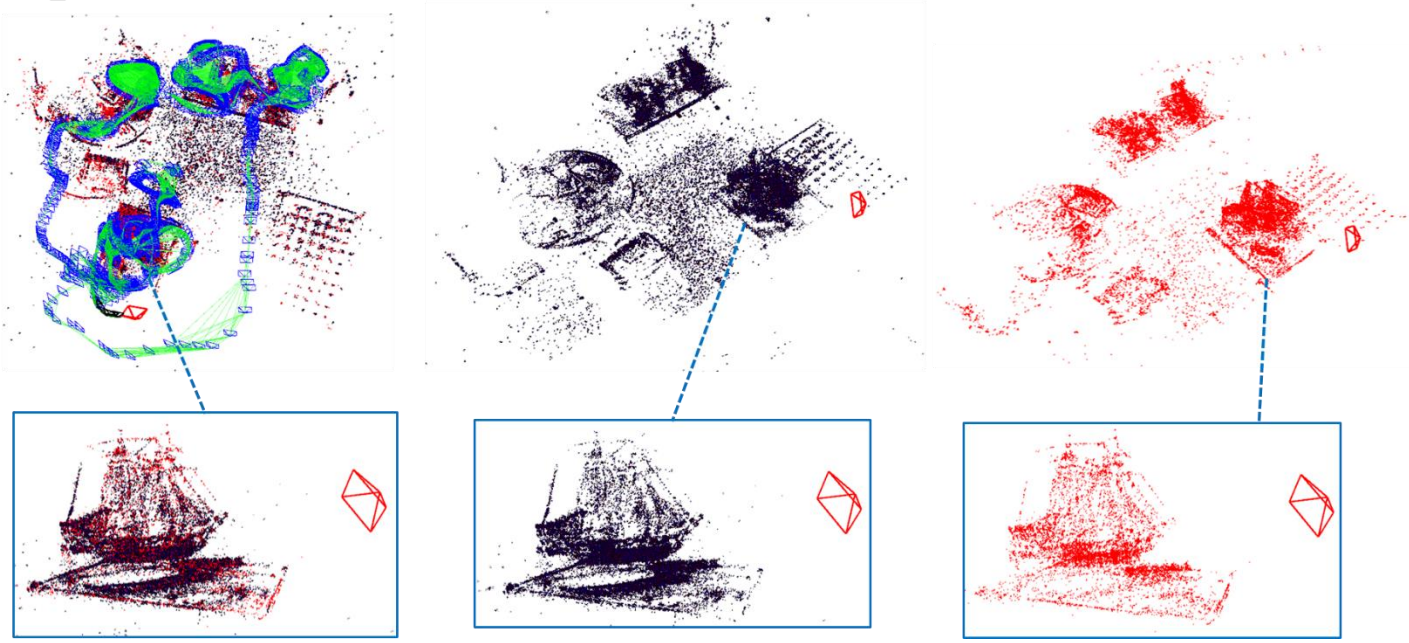


table_3



desk_3



4 Usage of Slanted Support Plane Model

As stated in the ablation study, the improvement of using slanted model is not very notable, in terms of mapping accuracy (Fig. 6) and tracking performance (Fig. 7a), especially on noisy sequences.

From our simulations, we think that the slight difference results from the slow convergence of plane parameters (i.e., theta in Eq 11), due to the limited number of iterations (only three iterations are used in the paper, where two for depth estimates and one for derivatives). The original implementation in [22] performs an iterative optimization with a maximal number of 20 iterations, and uses more robust NCC scores to reject outliers, which is however not affordable in a VO application.

Initially, we indeed carried out five iterations for the optimization of Eq. 11, where the odd iterations (first, third and fifth) are to optimize depth value, and the even (second and fourth) are to adjust derivatives. The depth deviations for the ICL-NUIM noisy dataset, as well as the cumulative tracking error plots are showed below.

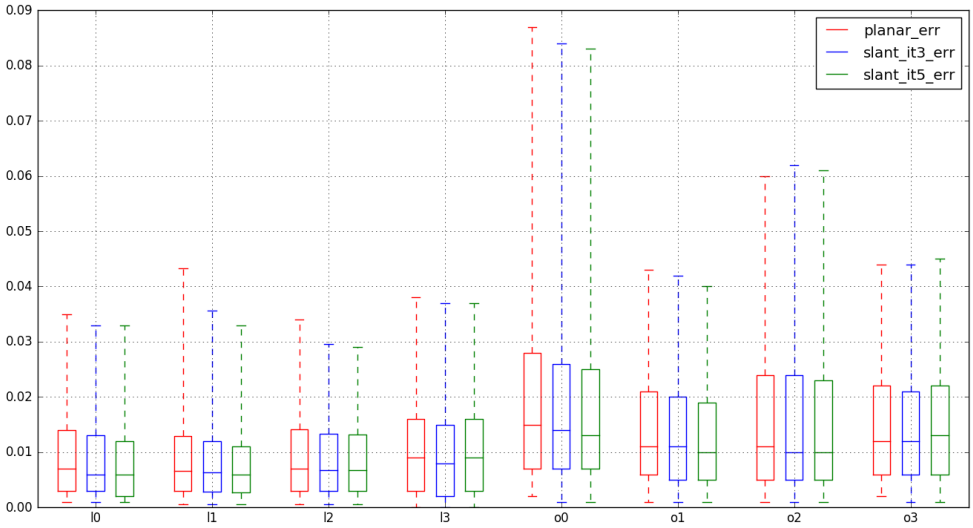


Figure 1. Depth deviations to the ground truth depth on ICL-NUIM noisy dataset

From the figure, we can see that the accuracy of depth estimates can be further improved by applying more iterations, except for the sequences *living_t3* and *office_t3*. On these two sequences, the feature-based part in hybrid front-end tracking got lost during a small segment due to insufficient matches, since executing more iterations further increases the time required for mapping thread, thus slowing the expansion of the sparse feature map and leading to track lost in some cases, e.g., weakly-textured walls. Although the feature associations can recover later, the absolute tracking performance still degrades, thus affecting the estimation of depth values.

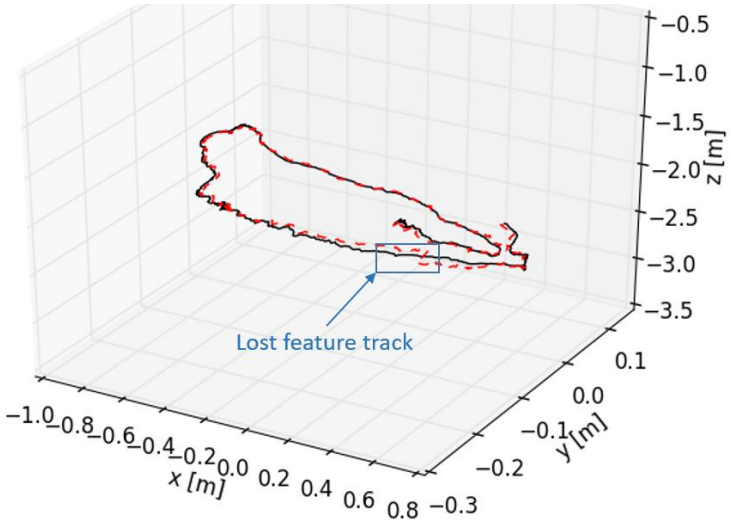


Figure 2. Estimated 3D camera trajectory on the sequence *office_t3*

The cumulative error plots for ICL-NUIM noisy, TUM RGB-D and ETH SLAM datasets, as well as the corresponding median values of absolute trajectory errors are also given.

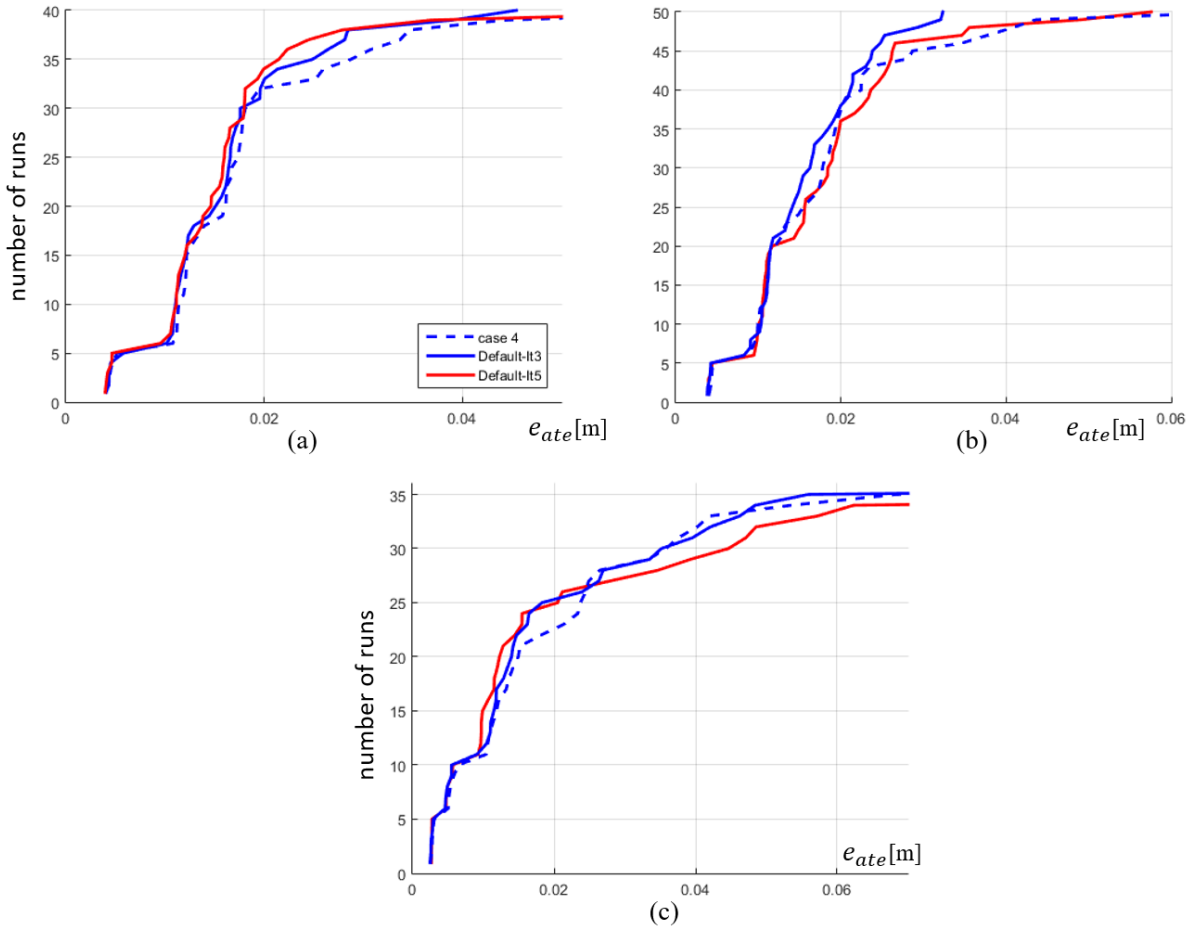


Figure 3. Results for (a) ICL-NUIM noisy, (b) TUM RGB-D, and (c) ETH SLAM.

Table 1. Median value of the absolute trajectory errors [cm] on ICL-NUIM noisy, TUM RGBD, and ETH SLAM datasets. The best and worst results are marked in green and red respectively. For the Default cases (it_3 and it_5), the left column list the median error value

and the right column the relative accuracy compared to the method using planar model (Case 4). The relative accuracy is computed as $(ME_S - ME_P) / ME_P$, where ME_S represents the median error of Default cases and ME_P the median value of Case 4.

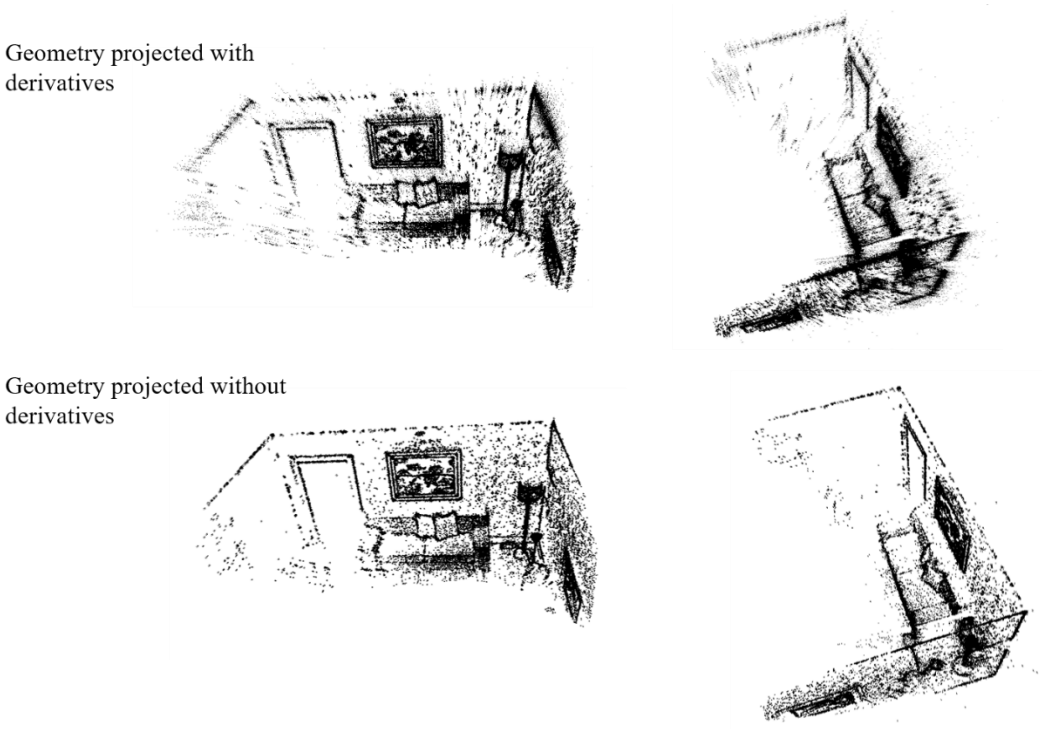
	Case 4	Default (it_3)		Default (it_5)	
lt0_n	0.45	0.44	**-2.21%**	0.42	**-5.86%**
lt1_n	1.21	1.11	**-8.62%**	1.09	**-10.0%**
lt2_n	1.26	1.24	**-2.22%**	1.22	**-3.29%**
lt3_n	1.13	1.12	**-1.23%**	1.20	+6.37%
ot0_n	3.37	2.81	**-16.5%**	2.79	**-17.2%**
ot1_n	2.52	2.00	**-20.4%**	1.81	**-28.2%**
ot2_n	1.68	1.68	+0.41%	1.55	**-7.54%**
ot3_n	1.75	1.63	**-6.89%**	1.80	+2.77%
fr1 xyz	1.10	1.11	+0.69%	1.08	**-1.62%**
fr1 desk	1.78	1.64	**-7.51%**	1.56	**-12.1%**
fr2 xyz	0.44	0.41	**-5.92%**	0.40	**-8.90%**
fr2 desk	0.99	1.04	+4.97%	0.98	**-1.02%**
fr3 office	1.80	1.67	**-7.29%**	1.70	**-5.57%**
fr3 s_t_far	1.02	1.03	+1.20%	1.05	+2.82%
fr3 ns_t_near	1.99	1.95	**-1.67%**	1.96	**-1.42%**
fr3 ns_t_far	3.77	2.38	**-36.9%**	2.66	**-29.6%**
fr3 s_nt_near	1.35	1.38	+2.23%	2.52	+87.4%
fr3 s_nt_far	2.01	1.84	**-8.31%**	2.36	+17.4%
cables_1	1.14	1.11	**-3.02%**	0.99	**-13.2%**
planar_2	0.28	0.26	**-6.00%**	0.27	**-2.68%**
table_3	0.54	0.49	**-8.36%**	0.49	**-8.29%**
mannequin_1	1.52	1.47	**-3.36%**	1.16	**-23.9%**
scene_1	1.49	1.40	**-6.39%**	1.23	**-17.7%**
sofa_1	3.33	3.96	+18.7%	4.46	+34.0%
ceiling_2	28.8	27.6	**-4.09%**	-	-
desk_3	4.19	3.52	**-16.1%**	4.71	+12.3%

The results in Figure 3 and Table 1 above show that the tracking accuracy further increases (e.g., ICL-NUIM dataset, and easy-level sequences on ETH SLAM), but the robustness decreases (difficult sequences on TUM RGBD and ETH SLAM), when increasing the number of iterations. We argue that the enhancement in accuracy results from the improved depth estimates (as showed in Figure 1), while the robustness declines for the same reason analyzed above, i.e., the lost of feature track. This is more commonly observed in those chanellenging sequences (e.g., *fr3_s_nt_near/far*, *ceiling_2*, and *desk_3*), where the feature associations are already kind of weak.

We also have tried other iteration configurations, but still found it tricky to strike a balance between convergence and time cost. For a better overall performance across all the datasets, we finally reduce the iteration number to 3 times for producing the results presented on the paper. Although the improvement in absolute metric scale is not

apparent, using a slanted model indeed benefits the tracking performance, from the perspective of relative accuracy listed in Table 1 above.

It is worth noting that the (inverse) depth derivatives $[d_{u,x},\ d_{u,y}]$ (see Eq. 10) are used as the auxiliary variables to enhance the estimation accuracy of depth d_u associated to the central pixel, and not applied to the direct BA stage for computing the cost term defined in Eq. 7. This means that when computing the cost term over the pixel neighborhood (defined in Eq. 7), this center depth is propagated to its neighboring pixels still following a fronto-parallel assumption like in DSO. The reason is that we found radial structures as well as many outliers would be produced if we use the derivatives estimated via a limited number of iterations to model the real 3D geometry, illustrated as follows:



Actually in standard multi-view reconstruction pipelines, the central depth would be propagated to its adjacent pixel that is then optimized over its own neighborhood, and this process would be repeated several times to obtain consistent point clouds. However, such a scheme is not affordable for a real-time VO/SLAM task, and we thereby only take this slanted plane model to better estimate the depth of the central pixel.

5 Standard Deviation of Multiple Trials (Figure 7)

Table 2: Standard deviation [cm] of the absolute trajectory errors over 5 trials on each sequence (corresponding to Figure 7(a)). The Nan symbol indicates that there exist at least one tracking failure among all trials on the sequence.

	Case 1	Case 2	Case 3	Case 4	Default
lt0	0.0382	0.0377	0.0845	0.2433	0.0511
lt1	0.7315	0.8237	0.0534	0.1317	0.0493
lt2	1.9439	0.3383	0.1617	0.2984	0.2132

lt3	0.1053	0.1749	0.1650	0.2002	0.0750
ot0	9.2496	0.5308	Nan	0.5710	0.7722
ot1	1.0345	0.8849	2.0494	0.2013	0.2545
ot2	0.1952	0.1211	0.0960	0.0972	0.1898
ot3	0.1551	0.2413	0.3159	0.0835	0.1799
lt0_n	0.0203	0.0089	0.0536	0.0410	0.0680
lt1_n	6.9010	0.4354	0.0816	0.0830	0.0614
lt2_n	0.7404	0.2441	1.2096	0.2342	0.2324
lt3_n	1.3351	0.5000	0.1892	0.1265	0.1008
ot0_n	Nan	6.3133	0.6568	1.8159	0.8987
ot1_n	1.0521	0.4140	1.6687	0.6627	0.3786
ot2_n	0.2225	0.0956	0.4160	0.0912	0.0797
ot3_n	0.0927	0.1822	0.2081	0.0977	0.0905
fr1 xyz	Nan	0.0114	0.0881	0.0208	0.0161
fr1 desk	3.5449	2.9143	9.2522	2.3464	0.7012
fr2 xyz	Nan	0.0131	0.0524	0.0186	0.0198
fr2 desk	Nan	0.0416	0.0522	0.1615	0.1405
fr3 office	0.9392	0.4158	0.2006	0.2247	0.5449
fr3 s_t_far	0.0656	0.1085	0.0803	0.0740	0.0835
fr3 ns_t_near	0.2120	0.1587	0.0838	0.1698	0.1538
fr3 ns_t_far	0.3891	0.7272	1.2096	0.5782	0.4526
fr3 s_nt_near	0.7971	0.5154	0.4185	0.2567	0.1349
fr3 s_nt_far	1.0085	0.8858	0.3818	0.4511	0.3633
cables_1	0.1846	0.3634	0.1494	0.1166	0.1058
planar_2	0.0134	0.0048	0.0274	0.0190	0.0173
table_3	0.1638	0.2371	0.2455	0.0590	0.0435
mannequin_1	1.5299	1.2783	0.9286	1.0549	0.2879
scene_1	0.3062	0.1502	0.3820	0.4180	0.1579
sofa_1	0.4456	1.3048	0.9158	0.7611	0.8222
ceiling_2	Nan	Nan	21.9205	Nan	13.5339
desk_3	13.0362	20.9591	5.4402	1.9332	1.3505

Table 3: Standard deviation [cm] of the absolute trajectory errors of 5 trials (corresponding to Figure 7(b) and 7(c))

	Indirect	Direct	ORB-VO	ORB-SALM
lt0	0.6383	0.2917	0.0470	0.0311
lt1	0.0643	0.1353	0.6699	2.4259
lt2	0.2787	0.9753	0.2068	0.2599
lt3	0.2320	0.0669	0.6206	0.2385
ot0	0.4286	0.1077	0.3617	0.2748

ot1	0.1073	0.8686	0.6554	2.0079
ot2	0.1811	0.1324	0.1282	0.1367
ot3	0.2401	0.0952	0.3334	2.5526
lt0_n	0.0356	0.0087	0.0340	0.0612
lt1_n	0.0939	0.0997	1.1752	1.4706
lt2_n	1.8928	0.0688	0.1968	0.4361
lt3_n	0.3219	0.2643	0.9488	0.0767
ot0_n	1.0866	0.7507	0.3294	0.2886
ot1_n	12.4393	0.3101	0.8182	1.3173
ot2_n	3.9611	3.9411	0.1517	0.2849
ot3_n	0.2374	0.0277	0.1553	2.0041
fr1 xyz	0.0210	Nan	0.0152	0.0285
fr1 desk	Nan	16.7117	Nan	Nan
fr2 xyz	0.0281	0.0445	0.0180	0.0139
fr2 desk	14.9415	2.8585	0.0706	0.1240
fr3 office	0.3598	0.2909	0.4586	0.0943
fr3 s_t_far	0.0998	0.0570	0.0846	0.0740
fr3 ns_t_near	0.0874	0.0689	0.5671	0.3210
fr3 ns_t_far	1.4955	0.7346	2.8640	2.0654
fr3 s_nt_near	0.3942	0.2407	0.9469	0.3002
fr3 s_nt_far	0.1493	Nan	2.6633	1.5188
cables_1	0.6276	0.2415	0.0467	0.0364
planar_2	0.0225	0.0375	0.0340	0.0252
table_3	0.0879	0.7396	0.0589	0.0933
mannequin_1	0.7886	0.8648	0.3233	0.6087
scene_1	0.2853	0.6177	0.6040	0.3093
sofa_1	0.8267	1.4389	1.7700	1.5622
ceiling_2	Nan	Nan	Nan	Nan
desk_3	10.5212	25.0540	10.1901	0.7398