2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

A Benchmark for RGB-D Visual Odometry, 3D Reconstruction and SLAM

Ankur Handa[1], Thomas Whelan[2], John McDonald[2] and Andrew J. Davison[1]

***Abstract*— We introduce the Imperial College London and National University of Ireland Maynooth (ICL-NUIM) dataset for the evaluation of visual odometry, 3D reconstruction and SLAM algorithms that typically use RGB-D data. We present a collection of handheld RGB-D camera sequences within synthetically generated environments. RGB-D sequences with perfect ground truth poses are provided as well as a ground truth surface model that enables a method of quantitatively evaluating the final map or surface reconstruction accuracy. Care has been taken to simulate typically observed real-world artefacts in the synthetic imagery by modelling sensor noise in both RGB and depth data. While this dataset is useful for the evaluation of visual odometry and SLAM trajectory estimation, our main focus is on providing a method to benchmark the surface reconstruction accuracy which to date has been missing in the RGB-D community despite the plethora of ground truth RGB-D datasets available.**

I. INTRODUCTION

A key attribute of any reconstruction algorithm is the level of detail it can recover from a set of data. Multi-view stereo fusion of images has long been a research focus of the vision community, and has recently become feasible in real-time for reconstructing surfaces from monocular video [1], [2]. The depths of points in the scene are optimised in an energy minimisation framework fusing data from multiple images using a lambertian surface assumption. However, the arrival of commodity depth sensors has led to joint RGB-D fusion approaches that offer increased robustness and accuracy in a range of conditions. The use of active illumination means that these RGB-D approaches are able to recover the 3D shape of surfaces whose appearance properties do not necessarily obey the assumptions commonly used in multi-view stereo methods.

The increased computational processing power in commodity graphics processing units (GPUs) available today has greatly proliferated the interest in real-time surface reconstruction. Systems like KinectFusion [3] and Kintinuous [4] have demonstrated the capability to create dense surface reconstructions by delegating the computational burden to these high performance GPUs present in typical desktop or laptop machines. Reconstructions obtained from these systems qualitatively appear accurate, but there are currently no published quantitative assessments of the accuracy of the 3D surface reconstruction. This is largely due to the inavailability of datasets with 3D surface ground truth.

[1]A. Handa and A. J. Davison are with the Department of Computing, Imperial College London. ahanda,ajd at doc.ic.ac.uk

[2] T. Whelan and J. McDonald are with the Department of Computer Science, National University of Ireland Maynooth, Co. Kildare, Ireland. thomas.j.whelan at nuim.ie

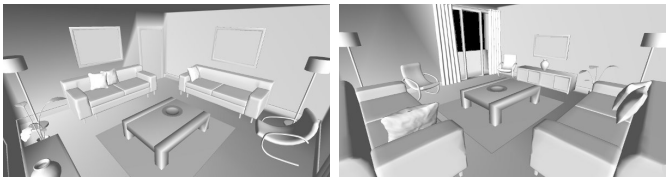


Fig. 1. The interior of our synthetic living room scene (color removed to highlight geometry). The scene is composed of various common objects *e.g.* vase, table, lamp, chairs, sofa *etc.*

In this paper we address this problem and provide a synthetic dataset with both surface and trajectory ground truth to fully quantify the accuracy of a given SLAM system or 3D reconstruction system. We obtain realistic trajectories for use in synthesized sequences by running the Kintinuous system in a standard real world environment and taking the estimated camera paths as ground truth trajectories. This synthetic approach is inspired by the recent work of Handa *et al.* [5] who used a similar synthetic approach to evaluate the performance of different frame rates for photometric camera tracking.

Ground truth dataset generation for the assessment of camera pose tracking and object segmentation has been previously presented in many recent works including Tsukuba [6], TUM RGB-D [7], [8], Sintel [9], KITTI [10] and the NYU Dataset [11]. However, all of these datasets are mainly aimed at trajectory estimation or pure two-view disparity estimation. None has the combination of realistic RGB plus depth data together with full 3D scene and trajectory ground truth over room-sized trajectories which we present here. Scene geometry ground truth for such trajectories suitable for SLAM has not been available for real sensor data due to the difficulty involved in its capture. In a laser scanner approach for example, each point measurement has an error tolerance and is not completely accurate, while occlusions encountered when scanning a scene make full watertight 3D surface models difficult to acquire. In addition to this, perfect time synchronisation with motion capture setups is impossible to achieve. We will show here that a carefully undertaken and high quality synthetic approach instead can fill this gap and serve a major role in enabling full comparison of RGB-D SLAM and visual odometry algorithms. We provide not only a number of realistic pre-rendered sequences, but also open source access to the full pipeline for researchers to generate their own novel test data as required.

We demonstrate the usefulness of our dataset and proposed testing methodology for both trajectory and scene reconstruction accuracy with a comparison of several camera motion

estimation alternatives within the Kintinuous RGB-D dense SLAM system.

All data, videos, documentation and scripts are available online under the Creative Commons Attribution license (CC-BY 3.0) at:

`http://www.doc.ic.ac.uk/~ahanda/VaFRIC/iclnuim.html`

II. PRELIMINARIES

In this section we focus on the mathematical minutiae involved in the rest of the paper and other technical assumptions used in the framework. Our synthetic framework is largely inspired by Handa *et al.* [5] who used POVRay [1] and synthetic trajectories to obtain ground truth depth maps and color images. Their dataset used only a single office scene [2] for their experiments. In addition to presenting trajectory only sequences in the same scene, we also make use of a new dataset with 3D surface ground truth. Our perspective camera model is the same as the one used by Handa *et al.* with similar field of view (90 degrees) and image resolution (640×480). The camera intrinsic calibration matrix $\mathbf{K}$ is given by

$$\mathbf{K} = \begin{bmatrix} 481.20 & 0 & 319.50 \\ 0 & -480.0 & 239.50 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

where the negative sign indicates that the Y-axis of the POVRay coordinate system is flipped to the image Y-axis due to POVRay using a left-handed coordinate system. All raycast frames are free from lens distortion and have not had any anti-aliasing applied. The provided ground truth camera poses are in standard $\mathbb{SE}_3$ representation which we denote by $\mathbf{T}_{a\_b}$, read as the pose of b with respect to a.

III. DATASET

Our synthetic dataset consists of images obtained from camera trajectories in raytraced 3D models in POVRay for two different scenes, the *living room* and the *office room*. While the office room has previously appeared in the work of Handa *et al.* [5], we introduce the new living room scene which unlike the office room scene, also has an associated 3D polygonal model which allows evaluation of the accuracy of the final reconstruction. The office room scene is a procedurally generated raytraced scene and thus there is no available polygonal model for surface reconstruction evaluation.

Figure 1 shows snapshots of the interior of the living room from two different views. The living room model is represented in the standard *.obj* file format with *.exr* textures. Images corresponding to different camera poses on a given trajectory are stored in typical RGB and D frame pairs while the camera poses are represented in $\mathbb{SE}_3$ format. In the following section, we describe the procedure to obtain physically realistic trajectories and render images from different poses in a trajectory.

[1] `http://www.povray.org/`

[2] `http://www.ignorancia.org/en/index.php?page=The_office`

Sequence	Frames	Length	Avg. Trans Vel.
kt0 (lr)	1510	6.54m	0.126ms^{-1}
kt1 (lr)	967	2.05m	0.063ms^{-1}
kt2 (lr)	882	8.43m	0.282ms^{-1}
kt3 (lr)	1242	11.32m	0.263ms^{-1}
kt0 (or)	1510	6.54m	0.126ms^{-1}
kt1 (or)	967	6.73m	0.206ms^{-1}
kt2 (or)	882	9.02m	0.302ms^{-1}
kt3 (or)	1242	7.83m	0.182ms^{-1}

TABLE I

STATISTICS ON THE FOUR PRESENTED SEQUENCES FOR BOTH SCENES, LIVING ROOM (LR) AND OFFICE ROOM (OR).

IV. TRAJECTORIES

We obtained source handheld trajectories by running Kintinuous on data collected in a living room and subsequently used these estimated trajectories in the synthetic scenes as ground truth to obtain images and depth maps for each sequence. The trajectories are named kt0, kt1, kt2 and kt3. The trajectory kt3 is particularly interesting due to the presence of a small "loop closure" in the path taken. Table I lists statistics on all four trajectories according to their scaling for the two synthetic scenes. All data is recorded at 30Hz.

Transforming trajectories from the source Kintinuous trajectory coordinate frame to the POVRay coordinate frame is done by applying the following rigid transformation:

$$\mathbf{T}_{pov\_cam} = \mathbf{T}_{pov\_kintinuous} \cdot \mathbf{T}_{kintinuous\_cam} \quad (2)$$

$\mathbf{T}_{kintinuous\_cam}$ denotes the pose obtained in the Kintinuous coordinate frame which is mapped to POVRay coordinate frame by applying the $\mathbf{T}_{pov\_kintinuous}$ transformation. $\mathbf{T}_{pov\_kintinuous}$ is obtained by simply mapping the initial pose to the POVRay pose from where the trajectory begins in the synthetic scene. In particular, the raw source trajectories output by Kintinuous were rotated and uniformly scaled to achieve suitable trajectories within the virtual scenes.

A. Living Room

Figure 2 shows images taken at different camera poses for different trajectories in the living room scene. Special care was taken to simulate real-world lighting phenomena commonly observed in real images *e.g.* specular reflections, shadows and colour bleeding. It is also worth mentioning that when doing pure photometric image registration these artefacts are considered outliers but to ensure that images look as photo-realistic as possible, it is important that they are present in the images.

B. Office Scene

Images obtained at different camera poses for the office room are shown in Figure 3. As previously mentioned the office room is rendered procedurally and does not have an explicit 3D model. Therefore, it is only suitable for benchmarking camera trajectory estimation performance.

In the following section, we detail the process used to alter the perfect raytraced images to simulate realistic sensor noise.

Fig. 2. Images of the living room scene taken at different camera poses. Different real-world lighting phenomena can be clearly observed in the images *e.g.* reflections, shadows, light scattering, sunlight (from the window) and colour bleeding.

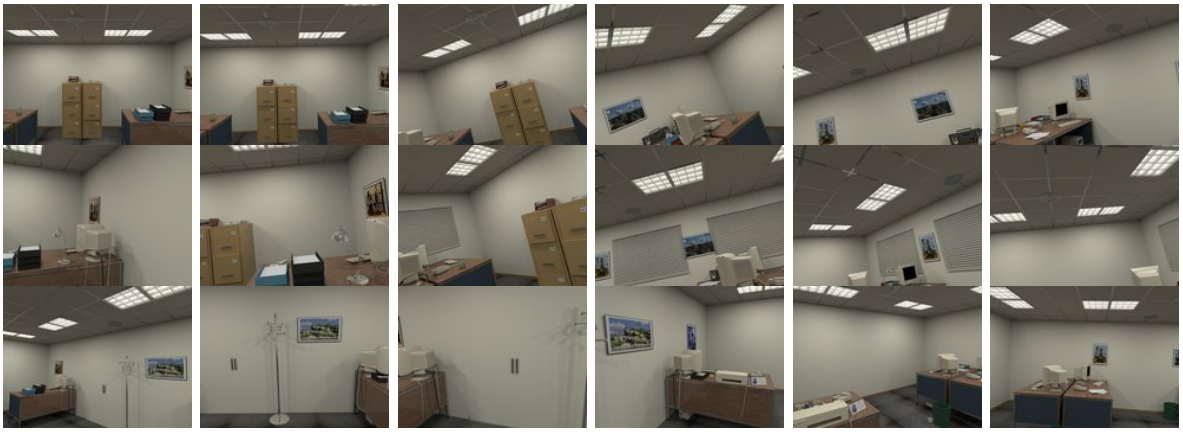


Fig. 3. Images of the office room scene taken at different camera poses. Various objects that are observed in a regular desktop room can be found in the images.

V. NOISE MODELLING

We consider two different sources of noise in our experiments: the noise occurring in the RGB image and the noise present in the depth image.

A. Depth noise

Our depth noise model is inspired by [12] who use random offsets to shift the pixel locations in the image and bilinearly interpolate the ground truth depth values as a first step towards simulating Microsoft Kinect sensor noise. These random shifts cause the depth map to become noisy while the bilinear interpolation ensures that the depth values among neighbouring pixels are correlated. Depth values are then converted to disparity and IID Gaussian noise is added to each disparity value. Finally the disparity values are quantised by rounding to the nearest integer and converted back to depth measurements. The procedure [3] is summarised by the following equation:

$$\hat{Z}(x,y) = \frac{35130}{\lfloor 35130/Z(x+n_x, y+n_y) + \mathcal{N}(0,\sigma_d^2) + 0.5 \rfloor} \quad (3)$$

where variables Z denotes the ground truth depth of a pixel, n_x and n_y denote the random shifts in x and y drawn from a Gaussian distribution $(n_x, n_y) \sim \mathcal{N}(0, \sigma_s^2 \cdot \mathbf{I})$ and σ_d denotes the standard deviation of noise in depth[4]. This model ensures that pixels with small disparity values (*i.e.* far away pixels) have low a SNR in depth and hence are noisier, while pixels with large disparity values are only slightly affected by the added noise.

Additionally, we perturb the vertices corresponding to the depth values along their normals with Gaussian noise $(\mathcal{N}(0,\sigma_\theta^2))$ and project these vertices to obtain laterally corrupted depth. The σ_θ is a bilinear function of the angle that the normal at a point makes with the camera ray and the depth. This has the effect of missing depth values on pixels with very oblique normal angles to the camera viewpoint. Figure 4 shows our synthetic ground truth depth maps with simulated sensor noise. The quantisation in the depth values is manifested in the form of depth bands that appear in the images.

[3]The number 35130 is obtained from the baseline of Kinect sensor by [12] and the associated supplementary material.

[4]We use $\sigma_s = \frac{1}{2}$ and $\sigma_d = \frac{1}{6}$ in our experiments.

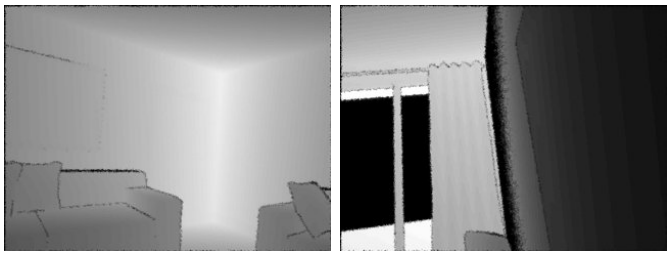


Fig. 4. Realistic depth maps according to the noise model we chose for our experiments. A movie of the associated trajectory can be found at `http://www.youtube.com/watch?v=JOKKYRoXnvg` where the quantisation and noise along the normals is more evident.

B. RGB noise

We follow a similar RGB noise model to [5] to modify the ground truth images with noise statistics one would obtain from a real camera. A Camera Response Function (CRF) is obtained by taking sample images of a static scene for different exposure times. The CRF essentially encodes the function that maps the irradiance on the camera to the digital n-bit brightness value observed in the image. This allows analytical modelling of the noise levels for all pixels in order to compute some noise parameters. These noise parameters are then used to add synthetic noise to the ground truth images. We assume a linear CRF, which is most common as found by Handa *et al.* [10].

VI. EVALUATION

A. Characterisation of experiments

We categorise our experiments in two different settings: experiments assuming perfect data with no noise and experiments assuming real world data with noise both in depth as well as RGB. To evaluate the accuracy of 3D reconstructions, we use the open source software `CloudCompare` [5] to align the ground truth *.obj* model with the reconstructed model produced by Kintinuous to compute reconstruction statistics.

B. Error metrics

We quantify the accuracy of surface reconstruction by using the "cloud/mesh" distance metric provided by `CloudCompare`. The process involves firstly coarsely aligning the reconstruction with the source model by manually selecting point correspondences. From here, the mesh model is densely sampled to create a point cloud model which the reconstruction is finely aligned to using ICP. Finally, for each vertex in the reconstruction, the closest triangle in the model is located and the perpendicular distance between the vertex and closest triangle is recorded. Five standard statistics are computed over the distances for all vertices in the reconstruction: Mean, Median, Std., Min and Max. We provide a tutorial on executing this process at `http://www.youtube.com/watch?v=X9gDAElt8HQ`.

[5] `http://www.danielgm.net/cc/`

Fig. 5. Reconstructions of the living room produced by Kintinuous with ICP for kt1 and kt2.

Error (m)	kt0 (DVO)	kt0	kt1	kt2	kt3
Mean	0.0662	0.0612	0.0034	0.0037	0.0085
Median	0.0593	0.0368	0.0026	0.0030	0.0063
Std.	0.0504	0.0821	0.0033	0.0034	0.0072
Min	0.0000	0.0000	0.0000	0.0000	0.0000
Max	0.3655	0.5456	0.0461	0.0508	0.1562

TABLE II
RECONSTRUCTION STATISTICS AS OBTAINED FROM CLOUDCOMPARE FOR THE LIVING ROOM SEQUENCES WITH NO NOISE. USING ICP ODOMETRY EXCEPT WHERE NOTED FOR KT0.

Our trajectory estimation statistics are inspired by [7], [8] who use absolute trajectory error (ATE) to quantify the accuracy of an entire trajectory. We evaluate 5 different odometers; (i) DVO [13] which performs full frame alignment between pairs of RGB-D frames making use of a robust cost function to improve robustness to outliers; (ii) FOVIS [14] which uses FAST feature correspondences between pairs of RGB-D frames to resolve camera motion, relying on traditional sparse corner type features in the image; (iii) RGB-D [15] which again performs full frame alignment between pairs of RGB-D images, with a less robust cost function than that of [13]; (iv) ICP as used in KinectFusion and Kintinuous [3], [4] which uses only geometric information in camera pose estimation with the notable feature of matching full new depth frames to the existing dense volumetric model of the scene contained within a truncated signed distance function (TSDF) and (v) ICP+RGB-D [16] which is a combination of the aforementioned ICP and RGB-D odometers in a weighted optimisation. Henceforth when discussing trajectory estimation error we refer to the root-mean-square-error metric (RMSE) introducted by Sturm *et al.* [7], [8].

C. Noiseless Scenario

1) Living Room: We run the Kintinuous pipeline on perfectly synthesized images for all four trajectories. A volume size of 4.5m was used in all experiments with a TSDF truncation distance of 0.045m. Sample reconstructions of the living room scene are shown in Figure 5. Table II shows all statistics for the reconstructions using the odometer with the best ATE score, with the exception of kt0 where the best (DVO) and second best (ICP) are listed.

Trajectory kt2 obtains the best mean trajectory error while the worst error statistics are obtained for kt0 where there are parts of the trajectory when the camera is looking at a planar lightly textured region of the scene. Interestingly although the

Odometer	Error (m)	kt0	kt1	kt2	kt3
DVO	RMSE	**0.1138**	0.1055	0.1073	0.1879
	Mean	0.0929	0.0875	0.1034	0.1622
	Median	0.0750	0.0678	0.1031	0.1485
	Std.	0.0657	0.0589	0.0286	0.0947
	Min	0.0040	0.0039	0.0463	0.0328
	Max	0.3091	0.2035	0.1669	0.4217
FOVIS	RMSE	1.9312	2.6679	2.5613	2.1551
	Mean	1.8224	2.5653	2.5051	1.7472
	Median	1.6400	2.4828	2.5938	1.4537
	Std.	0.6389	0.7328	0.5334	1.2615
	Min	0.7482	0.4178	1.0820	0.2167
	Max	3.2882	4.2334	3.2587	5.6442
RGB-D	RMSE	0.4558	0.6288	0.1609	1.0294
	Mean	0.3771	0.5372	0.1339	0.9653
	Median	0.2986	0.4165	0.1018	1.0120
	Std.	0.2559	0.3269	0.0891	0.3575
	Min	0.0635	0.0513	0.0441	0.2758
	Max	1.1423	1.2194	0.4090	1.6445
ICP	RMSE	0.1188	**0.0023**	**0.0015**	**0.0200**
	Mean	0.1045	0.0019	0.0014	0.0165
	Median	0.0753	0.0016	0.0013	0.0124
	Std.	0.0566	0.0012	0.0006	0.0004
	Min	0.0534	0.0005	0.0004	0.0055
	Max	0.2299	0.0048	0.0039	0.0172
ICP+RGB-D	RMSE	0.4365	0.0096	0.2151	0.6975
	Mean	0.3961	0.0086	0.2111	0.6687
	Median	0.3149	0.0078	0.2029	0.6307
	Std.	0.1833	0.0042	0.0415	0.1982
	Min	0.0423	0.0020	0.0868	0.3375
	Max	1.0187	0.0237	0.2973	1.0540

TABLE III

ABSOLUTE TRAJECTORY ERROR (ATE) FOR TRAJECTORIES ON THE LIVING ROOM SEQUENCES ASSUMING NO NOISE.

Odometer	Error (m)	kt0	kt1	kt2	kt3
DVO	RMSE	0.3977	0.4461	0.3271	0.2066
	Mean	0.3507	0.3993	0.2986	0.1880
	Median	0.3744	0.3795	0.2607	0.1632
	Std.	0.1875	0.1989	0.1335	0.0855
	Min	0.0231	0.1156	0.0936	0.0350
	Max	0.6883	0.9707	0.5475	0.3840
FOVIS	RMSE	3.3962	1.0309	1.7696	1.9822
	Mean	3.2087	0.9455	1.5408	1.8482
	Median	2.7582	0.9366	1.2945	1.7277
	Std.	1.1130	0.4109	0.8704	0.7164
	Min	1.3133	0.1856	0.5911	0.5636
	Max	6.0739	2.2207	5.9525	3.5404
RGB-D	RMSE	0.2701	0.6173	0.2664	0.4750
	Mean	0.2534	0.5271	0.2275	0.3593
	Median	0.2717	0.4744	0.1804	0.2390
	Std.	0.0935	0.3214	0.1387	0.3108
	Min	0.0634	0.0616	0.1235	0.0506
	Max	0.4620	1.5493	0.7902	1.5557
ICP	RMSE	**0.0029**	**0.0385**	**0.0016**	**0.0021**
	Mean	0.0026	0.0341	0.0015	0.0019
	Median	0.0026	0.0360	0.0015	0.0017
	Std.	0.0012	0.0179	0.0006	0.0010
	Min	0.0001	0.0021	0.0001	0.0002
	Max	0.0100	0.0862	0.0055	0.0043
ICP+RGB-D	RMSE	0.2248	0.4558	0.6367	0.0535
	Mean	0.2120	0.4238	0.5691	0.0399
	Median	0.1812	0.4150	0.4963	0.0257
	Std.	0.0748	0.1677	0.2854	0.0356
	Min	0.0508	0.0650	0.2465	0.0023
	Max	0.3529	0.8325	1.6939	0.1301

TABLE IV

ABSOLUTE TRAJECTORY ERROR (ATE) FOR TRAJECTORIES ON THE OFFICE ROOM SEQUENCES ASSUMING NO NOISE.

DVO method achieves a lower ATE on kt0, the reconstruction accuracy is lower than that achieved by ICP, which scored second best on the ATE. This is explained in how the ICP method performs frame-to-model tracking, where DVO does not. This also highlights how a better trajectory score does not imply a better reconstruction, even when both make use of a TSDF for volumetric fusion. If explicit frame-to-model tracking was also being employed by the DVO algorithm we would expect the surface reconstruction accuracy to improve as it would no longer be simply forward-feeding data into the TSDF. The heat maps shown in Figure 6 highlight the areas where the reconstructions are least accurate.

The Absolute Trajectory Error (ATE) statistics are listed in Table III where DVO narrowly beats ICP for trajectory kt0 where otherwise ICP maintains the best performance for all the other trajectories. FOVIS performs the worst for all different trajectories suggesting that the four other methods which all use dense full frame alignment are superior on this kind of data. In general, photometric methods appear to perform quite poorly, as observable in the ATE plots in Figure 8, though their necessity and merit is shown by the performance of DVO on kt0 where even on perfect data the geometry only reliant ICP breaks down.

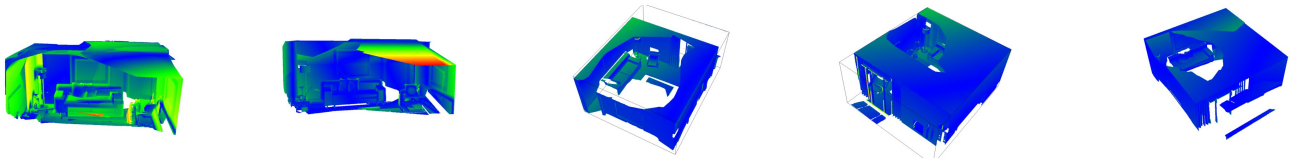
2) Office Room: As previously discussed, the office room scene is only suitable for trajectory estimation experiments. Although there is no surface ground truth to judge the accuracy of reconstruction, Figure 7 shows the reconstructions

Fig. 7. Reconstructions produced by Kintinuous with ICP for office scenes kt1 and kt2.

of two of the trajectories. Our trajectory estimation statistics for all five odometers are presented in Table IV. We again observe that ICP performs the best among all evaluated odometers, though with a notably higher error on kt1 than the other sequences, again due to the camera facing an area of the scene with low geometric texture. Once again the superiority of dense full frame alignment methods over sparse features is shown in the poor performance of FOVIS versus all other methods. However like on the living room dataset, the performance of the photometric-based odometers is quite poor, perhaps suggesting that some of the assumptions made by these methods are not holding on this kind of data. ATE plots are shown in Figure 8.

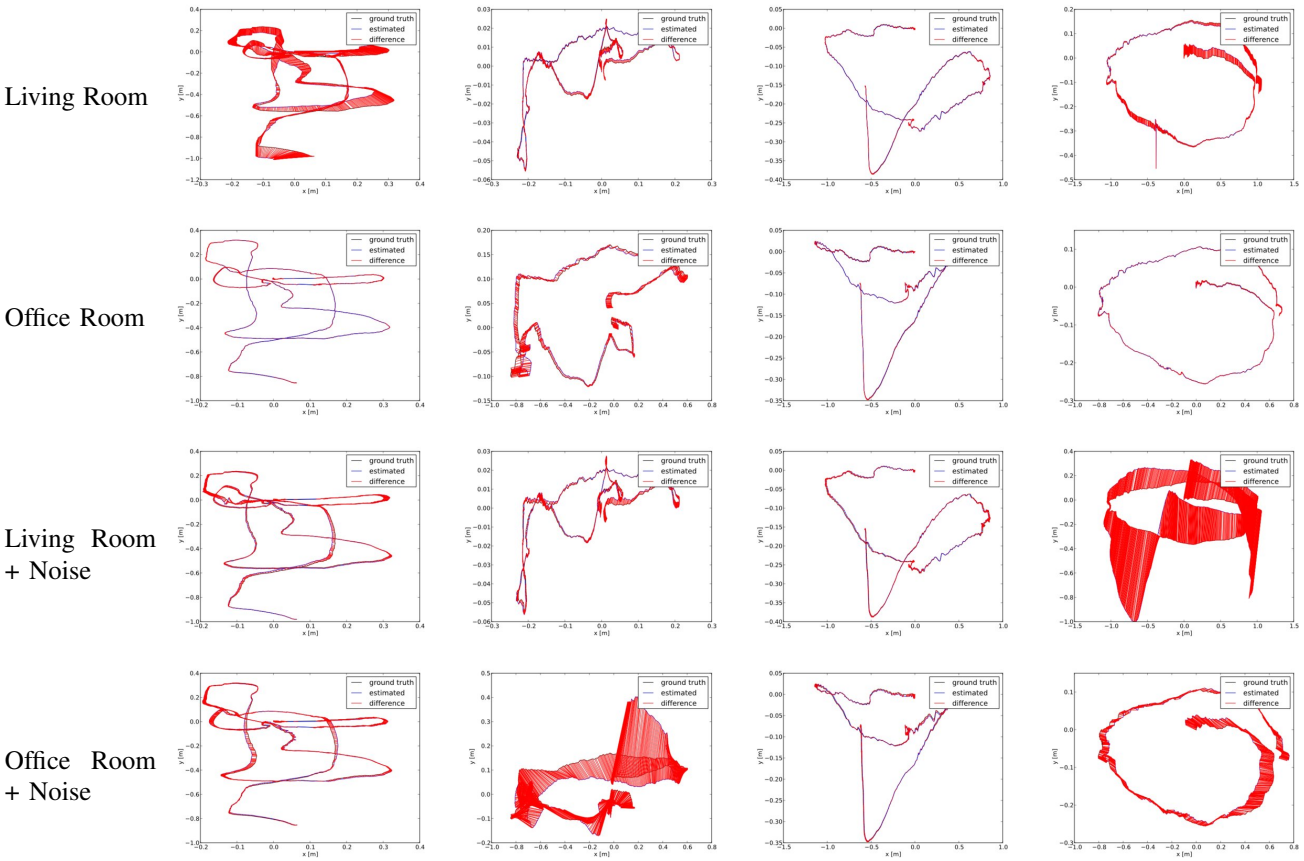
D. Real-World Simulation

Our real world data simulation sees a degradation in the quality of image as well as depth with both being affected by specifically modelled noise. We perform the same set of experiments on this noisy data and describe in detail below



(a) kt0 (DVO) (b) kt0 (c) kt1 (d) kt2 (e) kt3

Fig. 6. Heat maps for the reconstructions obtained for all four trajectories on the living room dataset. Colour coding is relative to the error obtained. All using ICP odometry except where noted for kt0. Note the roughness of the DVO reconstruction on kt0 compared to the ICP reconstruction, even though DVO scored a better ATE.



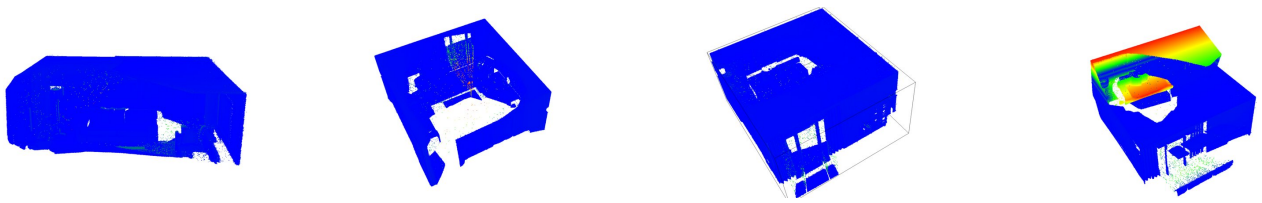
(a) kt0 (b) kt1 (c) kt2 (d) kt3

Fig. 8. Estimated trajectories from the best of each five evaluated odometers compared against ground truth trajectories (as highlighted in bold in Tables III, IV, V and VI).

our interpretation of the results.

1) Living Room: With the addition of sensor noise in all image channels, overall surface reconstruction error increases. ICP remains the best for all four sequences, with mean vertex-to-model error, listed in Table VII, increasing on all but kt0, where the error actually decreased. This can be explained by essentially random chance induced by noise as to whether or not the ICP pose estimate "slips" while the camera is facing a particular part of the scene where only a wall and very thin plant is visible. Reconstruction heat map renderings are shown in Figure 9. In terms of trajectory estimation, ICP remains the strongest on kt1, kt2 and kt3 while overtaking DVO on kt0 for the same reasons listed previously, with the error increasing on all trajectories with ICP except kt0, see Table V. There is a significant increase in error for ICP on the kt3 sequence, again induced by geometrically poor viewpoints made worse by noisy depth measurements. Interestingly the error for a number of photometric methods on various sequences including FOVIS decreases, implying that noise is somewhat useful for photometric-based methods. ATE plots are shown in Figure 8.

2) Office Room: As listed in Table VI, we see a more significant change in trajectory estimation performance. On all sequences the ICP error increases by varying degrees, either due to noise or issues associated with a reliance on geometry alone for pose estimation. And once again we see the performance of photometric-based methods increase, so



(a) kt0 (b) kt1 (c) kt2 (d) kt3

Fig. 9. Heat maps for the reconstructions obtained for all four trajectories with simulated noise on the living room dataset. Colour coding is relative to the error obtained. All using ICP odometry. Significant sparsely distributed surface outliers caused by noisy depth measurements cause the kt0, kt1 and kt2 heatmaps to appear mostly blue.

Odometer	Error (m)	kt0	kt1	kt2	kt3
DVO	RMSE	0.2911	0.1246	0.4733	0.5436
	Mean	0.2588	0.1189	0.4399	0.5127
	Median	0.2461	0.1116	0.4299	0.4516
	Std.	0.1332	0.0373	0.1749	0.1810
	Min	0.0118	0.0514	0.1370	0.2669
	Max	0.6340	0.1999	0.8342	0.9516
FOVIS	RMSE	2.0512	1.8676	1.4945	1.4742
	Mean	1.7323	1.6520	1.4067	1.2589
	Median	1.3307	1.7810	1.3514	1.1069
	Std.	1.0985	0.8712	0.5040	0.7671
	Min	0.2357	0.3332	0.2695	0.2258
	Max	6.4476	3.0389	2.7725	4.1914
RGB-D	RMSE	0.3603	0.5951	0.2931	0.8524
	Mean	0.3226	0.5083	0.2578	0.8159
	Median	0.3158	0.4411	0.1967	0.8174
	Std.	0.1604	0.3094	0.1394	0.2468
	Min	0.0419	0.1011	0.0459	0.1474
	Max	0.7435	1.1708	0.5150	1.3271
ICP	RMSE	**0.0724**	**0.0054**	**0.0104**	**0.3554**
	Mean	0.0690	0.0049	0.0097	0.3279
	Median	0.0587	0.0045	0.0104	0.2927
	Std.	0.0220	0.0024	0.0037	0.1371
	Min	0.0054	0.0006	0.0035	0.1205
	Max	0.1628	0.0113	0.0153	0.7691
ICP+RGB-D	RMSE	0.3936	0.0214	0.1289	0.8640
	Mean	0.3575	0.0189	0.1225	0.8364
	Median	0.2792	0.0161	0.1317	0.8158
	Std.	0.1645	0.0101	0.0405	0.2167
	Min	0.0589	0.0035	0.0395	0.2404
	Max	0.8730	0.0477	0.1862	1.2739

TABLE V

ABSOLUTE TRAJECTORY ERROR (ATE) FOR TRAJECTORIES ON THE LIVING ROOM SEQUENCES WITH SIMULATED NOISE.

Odometer	Error (m)	kt0	kt1	kt2	kt3
DVO	RMSE	0.3350	**0.3778**	0.3593	0.2338
	Mean	0.3018	0.3229	0.3075	0.2247
	Median	0.2735	0.2697	0.2058	0.2250
	Std.	0.1454	0.1962	0.1858	0.0647
	Min	0.0656	0.1045	0.0652	0.0591
	Max	0.6667	0.9203	0.6813	0.3287
FOVIS	RMSE	3.2956	1.3006	0.8661	1.5954
	Mean	3.0079	1.0859	0.8084	1.5403
	Median	2.8995	1.0391	0.7954	1.4895
	Std.	1.3466	0.7157	0.3107	0.4158
	Min	0.8151	0.0645	0.1674	0.5766
	Max	5.1697	3.2913	2.1270	2.2927
RGB-D	RMSE	0.1710	0.5366	0.2289	0.2298
	Mean	0.1562	0.4931	0.2106	0.2055
	Median	0.1376	0.5230	0.1971	0.1740
	Std.	0.0695	0.2117	0.0894	0.1029
	Min	0.0434	0.1015	0.0787	0.0722
	Max	0.3444	0.9700	0.4930	0.6628
ICP	RMSE	**0.0216**	0.9691	**0.0109**	0.9323
	Mean	0.0204	0.9062	0.0105	0.8443
	Median	0.0190	0.8420	0.0099	0.9285
	Std.	0.0073	0.3437	0.0029	0.3953
	Min	0.0022	0.2548	0.0041	0.1532
	Max	0.0491	1.9055	0.0257	1.7119
ICP+RGB-D	RMSE	0.2495	0.4395	0.4750	**0.0838**
	Mean	0.2354	0.4079	0.3466	0.0760
	Median	0.2037	0.4052	0.2817	0.0562
	Std.	0.0827	0.1637	0.3249	0.0353
	Min	0.0430	0.0515	0.0870	0.0306
	Max	0.3987	0.8007	1.5913	0.1400

TABLE VI

ABSOLUTE TRAJECTORY ERROR (ATE) FOR TRAJECTORIES ON THE OFFICE ROOM SEQUENCES WITH SIMULATED NOISE.

much so that DVO outperforms ICP on kt1 while ICP+RGB-D performs best on kt3. These results highlight how much more robust photometric-based methods are to noisy sensor readings. ATE plots are shown in Figure 8.

VII. CONCLUSIONS

In this paper we have presented a new benchmark aimed at RGB-D visual odometry, 3D reconstruction and SLAM systems that not only provides ground truth camera pose information for every frame but also provides a means of quantitatively evaluating the quality of the final map or surface reconstruction produced. Further, we have evaluated a number of existing visual odometry methods within the Kintinuous pipeline and shown through experimentation that a good trajectory estimate, which previous to this paper was the only viable benchmark measure, is not indicative of a good surface reconstruction. We additionally presented a simple synthetic noise model to simulate RGB-D data and provide a more realistic set of synthesized data, which demonstrated how existing photometric methods are more robust to sensor noise. Limitations include a lack of motion blur and rolling shutter simulation, however by providing the clean no noise raytraced data we wish to leave the door open to users of the benchmark to apply their own noise models to the data if they wish to simulate particular sensors more closely, where future models may well model the sensor under study more accurately. In future work we aim to generate new larger scenes and also include object annotations for each dataset, enabling ground truth benchmarking of machine learning driven scene understanding and

Error (m)	kt0	kt1	kt2	kt3
Mean	0.0114	0.0080	0.0085	0.1503
Median	0.0084	0.0048	0.0071	0.0124
Std.	0.0171	0.0286	0.0136	0.2745
Min	0.0000	0.0000	0.0000	0.0000
Max	1.0377	1.0911	1.0798	1.0499

TABLE VII

RECONSTRUCTION RESULTS FOR LIVING ROOM SEQUENCES WITH SIMULATED NOISE. ALL USING ICP ODOMETRY.

segmentation algorithms.

ACKNOWLEDGMENT

Research presented in this paper was funded by a Strategic Research Cluster grant (07/SRC/I1168) by Science Foundation Ireland under the Irish National Development Plan, the Embark Initiative of the Irish Research Council and ERC Starting Grant 210346. We would also like to thank Jaime Vives Piqueres who provided us with the open source living room scene.

REFERENCES

[1] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "DTAM: Dense tracking and mapping in real-time," in *Computer Vision (ICCV), 2011 IEEE Int. Conf. on*, pp. 2320 –2327, November 2011.

[2] J. Stuehmer, S. Gumhold, and D. Cremers, "Real-time dense geometry from a handheld camera," in *Pattern Recognition (Proc. DAGM)*, (Darmstadt, Germany), pp. 11–20, September 2010.

[3] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon, "KinectFusion: Real-Time Dense Surface Mapping and Tracking," in *Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR)*, 2011.

[4] T. Whelan, J. McDonald, M. Kaess, M. Fallon, H. Johannsson, and J. J. Leonard, "Kintinuous: Spatially Extended KinectFusion," in *Workshop on RGB-D: Advanced Reasoning with Depth Cameras, in conjunction with Robotics: Science and Systems*, 2012.

[5] A. Handa, R. A. Newcombe, A. Angeli, and A. J. Davison, "Real-Time Camera Tracking: When is High Frame-Rate Best?," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2012.

[6] M. Peris, S. Martull, A. Maki, Y. Ohkawa, and K. Fukui, "Towards a simulation driven stereo vision system," in *Pattern Recognition (ICPR), 2012 21st International Conference on*, pp. 1038–1042, IEEE, 2012.

[7] J. Sturm, S. Magnenat, N. Engelhard, F. Pomerleau, F. Colas, W. Burgard, D. Cremers, and R. Siegwart, "Towards a benchmark for RGB-D SLAM evaluation," in *Proceedings of the RGB-D Workshop on Advanced Reasoning with Depth Cameras at Robotics: Science and Systems (RSS)*, 2011.

[8] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for RGB-D SLAM evaluation," in *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS)*, 2012.

[9] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black, "A naturalistic open source movie for optical flow evaluation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 611–625, Oct. 2012.

[10] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3354–3361, IEEE, 2012.

[11] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *Computer Vision–ECCV 2012*, 2012.

[12] J. T. Barron and J. Malik, "Intrinsic scene properties from a single RGB-D image," June 2013.

[13] C. Kerl, J. Sturm, and D. Cremers, "Robust odometry estimation for RGB-D cameras," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, May 2013.

[14] A. S. Huang, A. Bachrach, P. Henry, M. Krainin, D. Maturana, D. Fox, and N. Roy, "Visual odometry and mapping for autonomous flight using an RGB-D camera," in *Int. Symposium on Robotics Research (ISRR)*, (Flagstaff, Arizona, USA), August 2011.

[15] F. Steinbrucker, J. Sturm, and D. Cremers, "Real-Time Visual Odometry from Dense RGB-D Images," in *Workshop on Live Dense Reconstruction from Moving Cameras at ICCV*, 2011.

[16] T. Whelan, H. Johannsson, M. Kaess, J. J. Leonard, and J. McDonald, "Robust real-time visual odometry for dense RGB-D mapping," 2013.