Probabilistic Object Reconstruction with Online Loop Closure

Anonymous 3DV submission

Paper ID ****

Abstract

In recent years, major advances have been made in 3D scene reconstruction, with a number of approaches now able to yield dense, globally-consistent models at scale. However, much less progress has been made for objects, which can exhibit far fewer unambiguous geometric/texture cues than a full scene, and thus be much harder to track against. Recently, robust, globally-consistent scene reconstruction has been achieved by combining a multi-segment representation with loop closure detection and an online model correction algorithm. The multi-segment approach naturally reduces drift, and the correction algorithm further refines the resulting model. In this paper, we show how to extend this approach to reconstruct accurate, globally-consistent object models. At the heart of our approach is a novel, probabilistic fusion framework that we use to separate the object of interest from the surrounding scene. We perform both qualitative and quantitative experiments to compare our approach to the current state-of-the-art, and demonstrate compelling improvements in both pose estimation and model quality.

1. Introduction

Dense Simultaneous Localisation and Mapping (SLAM) has proven very effective for reconstructing moderately-sized scenes, with much recent research driven by the availability of inexpensive, consumer-grade depth sensing equipment [12, 13, 17]. However, accurate pose estimation in the presence of erroneous measurements and visual aliasing in the scene remains difficult to fully solve. Common approaches [1, 20] exploit geometric/texture cues to estimate the pose of each frame, but tracking can drift – or even fail completely – without reliable features against which to track. Loop closure events are another common source of tracking problems, with explicit handling often required. Moreover, a failure to handle loop closures correctly can lead to errors in the model, exacerbating any existing tracking issues. These problems are only magnified when reconstructing objects: an object's surface will generally contain far fewer points than the full scene, and a lack of unambiguous points on the surface can lead to an increase in data association errors when attempting to recover the pose.

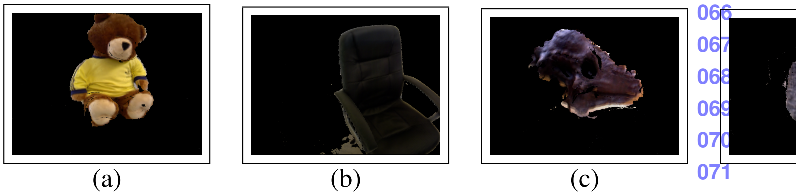
(a) (b) (c)

Figure 1: Reconstruction examples: **(a)** teddy, **(b)** chair, **(c)** dinosaur head, **(d)** rock.

Recently, robust, globally-consistent scene reconstruction has been achieved by combining a multi-segment representation with loop closure detection and an online model correction algorithm [7]. In this paper, we show how to extend this approach to reconstruct accurate, globally-consistent object models, introducing a probabilistic rigid-object reconstruction framework based on depth and texture features. The framework facilitates the correction of tracking drift by representing the object as a collection of overlapping subsegments between which transformations may be inferred to keep them aligned, resulting in a consistent overall model. By combining these transformed surfaces, we extract an implicitly deformed surface, optimised for via the probabilistic formulation that follows. We utilise a volumetric representation for each of these object subsegments, as with larger-scale reconstruction systems [7]. Each voxel in the subsegments has additional appearance posterior information pertaining to the voxel's membership of the object. Over time, multiple volumes containing both surface and probabilistic appearance information are maintained and manipulated to yield a robust and temporally consistent model. Finally, we optimise for the optimum object shape within a CRF (Conditional Random Field) framework.

We perform both quantitative and qualitative experiments to compare our approach to the state-of-the-art object reconstruction approach of Ren et al. [29], demonstrating compelling improvements in both pose estimation and model quality.

2. Related Work

Dense 3D Reconstruction. Much recent work stems back to the seminal KinectFusion work of Newcombe et al. [12]. This was used to build an implicit, voxel-based TSDF representation [3] of a small-scale environment, but could only reconstruct static scenes, and struggled to scale due to inefficient use of memory and difficulties in preventing significant tracking drift in larger-scale scenes. Scalability has progressively been addressed by a moving reconstruction window [19, 26], octrees [30], and sparse methods based on voxel hashing and streaming data to and from the GPU [13, 17]. This has made it possible to reconstruct static scenes whose size is only limited by available system memory, although reconstructing a large scene can still occupy significant space. Tracking drift has also been addressed to some extent, generally by detecting loop closures and either rigidly or non-rigidly deforming parts of the scene [31, 27, 28]. Other approaches exist that do not explicitly detect loop closures [5]. Recently, Dai et al. [4] introduced a system that improves pose estimation for large-scale scenes by considering each previously seen frame within a hierarchical framework and performing sparse feature matching to optimise for the camera pose. However, mismatches between keypoints are reported to have an impact on the robustness of their optimisation procedure. Kähler et al. [7] took a different approach, showing how to combine a multi-segment representation of the scene with loop closure detection and an online model correction algorithm to achieve accurate, globally-consistent scene reconstruction. They reduce drift by tracking only against recent segments of the scene and adjusting the poses between segments online, before refining the final model using pose graph optimisation. In this paper, we extend this latter approach to achieve globally-consistent reconstructions of objects.

Object Reconstruction. In addition to scene-scale works, there has been much work on object reconstruction and object-centric SLAM. Kolev et al. [8] presented a probabilistic 3D segmentation and surface extraction algorithm based on a variational evolution of a level set representation, but did not handle loop closures and tested their approach only on images with no background noise. Weise et al. [25] presented an explicit, surfel-based reconstruction system for objects rotating in front of a 3D range scanner. They maintain an object topology graph to handle loop closures online, but they only deform the object model when detecting loop closure events, whereas our approach continuously updates the rigid transformations between object segments, making it easier for the user to see what the final model will look like. Ohno et al. [14] presented a robotic system for reconstructing unknown objects in an environment by pushing them and estimating their motion using 3D flow. Krainin et al. [9] present another robotic system that uses Kalman filtering and articulated ICP to track both the robot's manipulator and the object. They perform loop closure in a similar way to [25] and achieve very good surfel models of the object, but their approach requires specialist hardware. Cui et al. [2] presented an object reconstruction system based on Time of Flight (ToF) sensors. They use a super-resolution representation of chunks of raw depth images to reconstruct detailed models. Mihalyi et al. [10] used augmented reality markers to make it possible for untrained users to achieve robust object reconstructions. Their approach works for a range of objects, but the need to add markers to the scene in advance is quite limiting in practice. Narayan et al. [11] combine KinectFusion with visual hull techniques to reconstruct objects with concavities and translucencies. Panteleris et al. [15, 16] reconstruct objects by tracking hand-object manipulations. Their approach runs in real time, but they do not handle loop closures. Tzionas and Gall [24] also make use of hand-object interactions, presenting an elegant system that can reconstruct featureless and highly symmetric objects by tracking contact points between the hand and the object. Their system produces appealing results, but is not real-time and can fail if the fingers slip over the manipulated object. Gupta et al. [6] performed object reconstruction based on monocular, multi-view cues. They segment the objects using graph cuts and track based on geometric/texture cues, but do not handle loop closures and report fluctuating tracking quality caused by illumination variation and specular surfaces. Recently, Slavcheva et al. [22] presented an object reconstruction system that estimates poses by registering pairs of TSDF volumes. Their system handles loop closures and achieves high-quality results, but at the cost of relying on fiducial markers to improve their tracking and performing loop closure offline as a post-processing step.

The closest approach to ours is that of Ren et al. [29], who presented a probabilistic tracking and reconstruction system that reconstructs objects based on an appearance model, evolving a level set representation for voxels that are on the object. However, they do not detect loop closures and are prone to tracking drift. Their later work [18] extended [29] to track multiple objects for which an initial model is available in real time.

3. Method

An overview of our object reconstruction pipeline is shown in Figure 2. We divide our object model into subvolumes, exactly one of which is active at any one time. Each subvolume consists of a TSDF, colour volume and object probability volume, and contains a rigid body transform that specifies its pose relative to the global coordinate frame. At each new camera frame, we apply our segmentation model to the colour input image to construct an object probability map, and then fuse the probabilities in this map into the object probability volume of the active subvolume (see Sec-

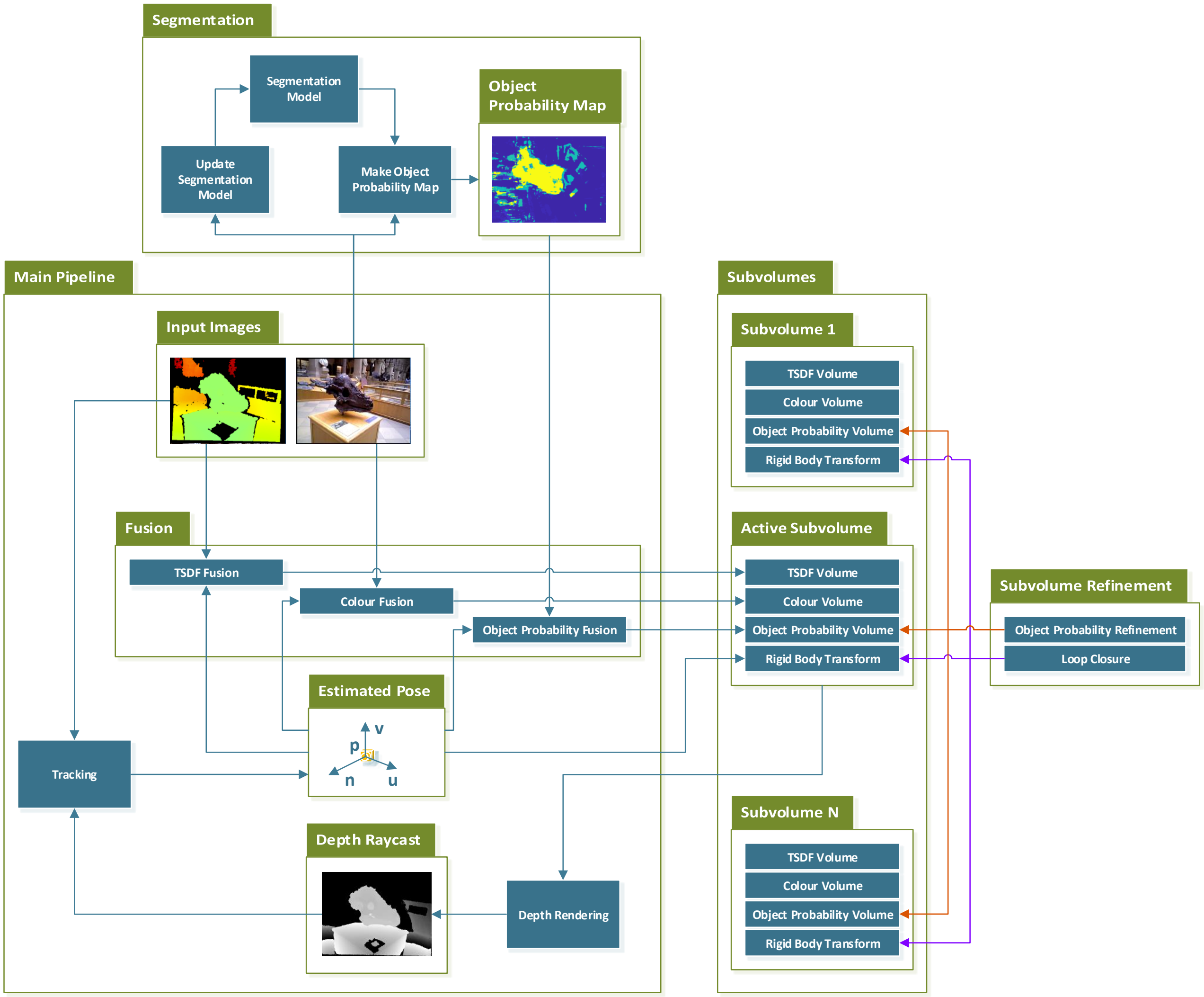

Figure 2: The pipeline of our object reconstruction approach.

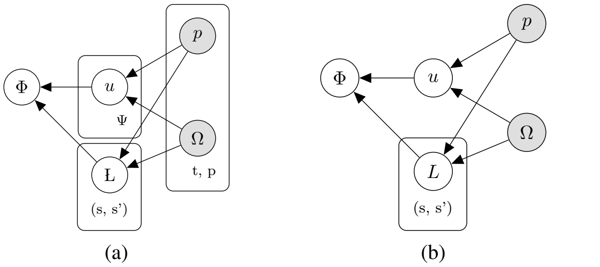

Figure 3: (a) An initial probabilistic formulation of the object reconstruction pipeline, and (b) a simpler formulation based on various simplifying assumptions (see text).

tion 3.1). We also update the TSDF and colour volume of the active subvolume, and track against it using ICP, as in normal KinectFusion [12]. At the end of each frame, we run our online model correction algorithm (see Section 3.3), which optimises the relative poses between the subvolumes to mitigate tracking drift. Once the entire reconstruction process is finished, we perform a CRF-based optimisation to refine the resulting object shape (see Section 3.5). In what follows, we describe each of these procedures in more detail. Our approach is not tied to the use of any one segmentation model, so we defer discussion of the model we use to the supplementary material.

3.1. Object Probability Fusion Procedure

The surface map and camera pose are estimated using the standard pipeline of [12, 17]. The surface location is represented as the zero level set of a truncated signed distance function (TSDF) over voxels, with online weighted-mean fusion of new observations. Pose estimation via ICP is run quasi-simultaneously against the evolving map. Here, inspired by [8], we augment this procedure by estimating the posterior probability, per map voxel, of belonging to the object. This volume of posterior probabilities is updated on each frame, parallel to the fusion process in the mapping and pose estimation subsystems. Similarly to [7], the representation of the reconstructed object comprises multiple 'subvolumes', each pertaining to some patch on the object surface. New subvolumes are started when sufficiently many new voxels have been allocated and had data integrated, and exactly one is 'active' at any time. By ensuring overlap between subvolumes, transformations between them can be found and pose inconsistencies addressed, online.

At each frame, the object probabilities for the visible voxels in the active submap are updated via fusion of an appearance-derived probability map for that frame. Under the assumption of conditional independence between frames (for sake of tractability), the posterior probability of a given voxel $\psi \in \Psi$ belonging to the object has the following form:

$$P(\psi \in \Phi | \Omega, p) = \prod_{t=0}^{\infty} P(\psi_t \in \Phi_t | \Omega_t, p_t) \tag{1}$$

where Ψ is the volume of voxels for which measurements are accumulated, Φ is the volume of voxels pertaining to the object, Ω_t is the current image observation at time t and p_t is the currently tracked pose at time t. This encodes the probability of a voxel belonging to the object as the product of the instantaneous appearance-derived pixel-wise conditionals over the times at which it was observed. (In the above, Φ is a discretisation of the continuous Φ in the probabilistic formulation that follows.)

3.2. Probabilistic Formulation of Object Reconstruction

As previously highlighted, central to the proposed system is a volume of posterior probabilities pertaining to a voxel wise membership of either the object set or the non object set. This allows one to formulate the full joint distribution over the object as the Probabilistic Graphical Model of Figure 3a.

Where Φ is the shape to be reconstructed, u is the appearance model volume, L is the set of consistency constraints for each adjacent sub volume pair in the form of rigid transformations, Ω is the set of RGBD image pixels and p the set of poses over time.

This gives rise to the following analytical formulation of the above distribution:

$$P(\Phi, \Omega, p, u, L) = \prod_{\psi \in \Psi} \prod_{(s,s') \in \mathcal{S}} P(\Phi | u_v, L_{(s,s')}) \prod_{t=0}^{\infty} \prod_{p \in \mathcal{P}} P(u_v | \Omega_{p,t}, p_t) P(L_{(s,s')} | \Omega_{p,t}, p_t) P(L_{(s,s')}) P(p_t) P(\Omega_{p,t}) \tag{2}$$

where $\mathcal{V}$ is the set of voxels across all sub volumes, $\mathcal{P}$ is the set of RGBD pixels for a given frame and $\mathcal{S}$ is the set of sub volumes.

However, if one were to assume temporal and pixel wise independence in the RGBD observations and temporal independence in the poses, the plate containing Ω and p can be removed:

$$P(\Phi, \Omega, p, u, L) = \prod_{v \in \mathcal{V}} P(\Phi | u_v) \prod_{(s,s') \in \mathcal{S}} P(u_v | \Omega, p, L_{(s,s')}) P(L_{(s,s')} | \Omega, p) \tag{3}$$

In practice this temporal independence assumption causes no issues.

Furthermore, if one assumes voxel wise independence, the plate over voxels can be removed. Finally, assuming $P(p)$ and $P(\Omega)$ are uniform distributions, then we have the simpler distribution given by Figure 3b.

Where the simpler distribution takes the following analytical form:

$$P(\Phi, \Omega, p, u, L) = \prod_{(s,s') \in \mathcal{S}} P(\Phi | u, L_{(s,s')}) P(u | \Omega, p) P(L_{(s,s')} | \Omega, p) P(L_{(s,s} \tag{4}$$

The above formalisms describe a probabilistic framework in which online corrections can be made to the reconstructed model to counteract errors incurred by pose tracking inconsistencies. As with previous dense SLAM systems [12, 17, 13], our system follows a pipeline that consists of a tracking stage and an integration stage. However, our formulation of this pipeline consists of an additional estimation module that relies on the use of a subvolume representation to correct tracking errors by applying transformations to the subsegments to align them when there are intra subsegment tracking inconsistencies. As previously described, during reconstruction the object is split in to subsegments, also referred to as subvolumes, with the pose estimation performed in each of the active, visible subsegments. The pose estimation stage for each of these subsegments follows the standard ICP(Iterated Closest Point) approach. As inference on the joint distribution of our model is intractable, conditional independence assumptions are made that do not appear to cause any functional issues. The estimation phase of the pipeline is described in the following section.

3.3. Online Model Correction

The tracking consistency constraint denoted by the variable L in the graphical model given by Figure 3a and Figure 3b can be enforced in terms of minimising the transformations between adjacent submaps, such that the camera poses tracked in each subvolume are consistent. Given instantaneously inferred transforms between subvolumes obtained from tracking results, the objective is to infer a robust, consistent deformation transformation for the subvolume pair. As such, for each pair of visible subvolumes (s, s'), the following posterior must be maximised:

$$P(\Omega, p|L_{(s,s')}) = \frac{P(L_{(s,s')}|\Omega, p)P(\Omega|p)P(p)}{P(L_{(s,s')})} \qquad (5)$$

The intuition behind the above equation is that the deformation $L_{(s,s')}$ applied to the probability field u should increase the probability of observing the current pose p given the current RGBD frame Ω by reducing the variance of the camera tracking result. As such, global tracking variance is reduced by enforcing local consistency, improving the quality of the reconstruction.

Gradient-based maximisation of the above posterior to yield an optimal deformation is a highly nonlinear optimisation problem. As such, it is suited to second-order gradient-based optimisation routines, e.g. Gauss-Newton or Levenberg-Marquardt. It should be noted that in our implementation the $P(\Omega|p)$ term is assumed to be uniform in the case of an RGBD sensor being used, however for applications such as monocular SLAM this term may be replaced with a noise model when there is significant uncertainty about the given depth map at each frame. The following proportionality to the distribution over deformations is made:

$$P(L_{(s,s')}|\Omega, p) \propto P(\Psi_s(x)|\Psi_{s'}(\Lambda(x))) \qquad (6)$$

With the log likelihood function taking the following form:

$$\ln P(\Psi_{s'}(\Lambda(x))) = m \ln \frac{1}{\sqrt{2\pi}\sigma} - \frac{1}{2\sigma^2} \sum_{(s,s')\in\mathcal{S}} \Big(\Psi_s(x) - \Psi_{s'}(\Lambda(x))\Big)^2 \qquad (7)$$

Where $\Psi(.)$ is a scalar valued SDF(Signed Distance Function), a discretised field of Φ, as previously described. x is a point represented by a 3-vector and $\Lambda(.)$ is a transformation function taking the following form:

$$\Lambda(x) = R(\rho_1, \rho_2, \rho_3)x + t \qquad (8)$$

Where $R(.)$ is a rotation matrix from the Special Orthogonal group $\mathbb{SO}(3)$ paramaterised by the three Rodrigues Parameters [21] ρ_1, ρ_2 and ρ_3. Note that the logarithmic form of the likelihood is suitable to Nonlinear Least Squares optimisation, allowing the posterior of equation 4 to be maximised in terms of the likelihood term of equation 4. To perform MLE(Maximum Likelihood Estimation) over this likelihood using an optimisation routine such as Levenberg Marquardt, the following gradients must be computed for the rotational component of the deformation:

$$\frac{\partial E}{\partial r_n} = \frac{\partial E}{\partial \Psi}\frac{\partial \Psi}{\partial \Lambda}\frac{\partial \Lambda}{\partial r_n} \text{for } n \in \{1, 2, 3\} \qquad (9)$$

Similarly for the translational component:

$$\frac{\partial E}{\partial t_d} = \frac{\partial E}{\partial \Psi}\frac{\partial \Psi}{\partial \Lambda}\frac{\partial \Lambda}{\partial t_d} \text{for } d \in \{x, y, z\} \qquad (10)$$

where the gradient $\frac{\partial \Psi}{\partial \Lambda}$ is found via finite differencing.

3.4. Implicit Surface Deformations

In the previous section, a model and estimation procedure was presented to find optimal transformations between the aforementioned subvolumes. The overall object surface Φ is implicitly deformed by a combining function $\zeta(\Phi)$ over each of the subvolumes to which transformations have been applied. As such the surface Φ is given by the following:

$$\Phi = \sum_{\chi \in X} \zeta(\Phi_\chi) \qquad (11)$$

where X is the set of subvolumes contributing to the surface Φ.

3.5. Volumetric Object Segmentation

The final stage in the proposed object reconstruction pipeline is the segmentation of the object voxels from those that have had measurements fused from the background. This segmentation is formulated within a CRF framework, where each node in the CRF represents a set of neighbouring voxels in space, with connections being made between adjacent neighbourhoods. The process of segmentation can be posed as an energy minimisation problem over a cut in voxel space, such that a segmentation in 3D is obtained. The following energy function consists of the unary posterior probabilities over appearance accumulated during the fusion process for a region in space and an additional pairwise smoothing term representing the physical appearance similarity of the object region represented by the voxel neighbourhoods γ and $\gamma^{'}$:

$$E_n = \prod_{t=0}^{\infty} \prod_{\psi \in \Phi_n} P(\psi \in \Psi|\Omega_t, p_t) + P(\mathbb{E}[c]_\gamma|\mathbb{E}[c]_{\gamma^{'}}) \qquad (12)$$

where c represents the set of colour measurements fused in to the voxels within a given neighbourhood, for all N subvolumes.

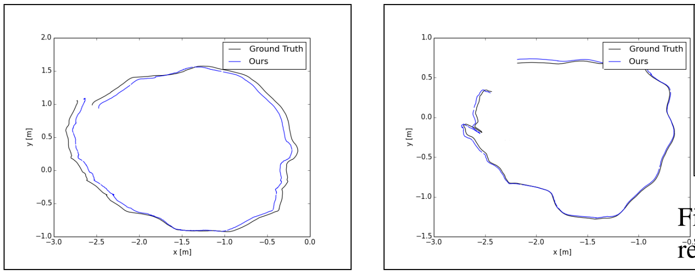

Figure 4: Comparison of the estimated camera trajectory with the ground truth for **(L)** *freiburg3_cabinet* (final ATE: $0.078m$), and **(R)** *freiburg3_teddy* (final ATE: $0.031m$).

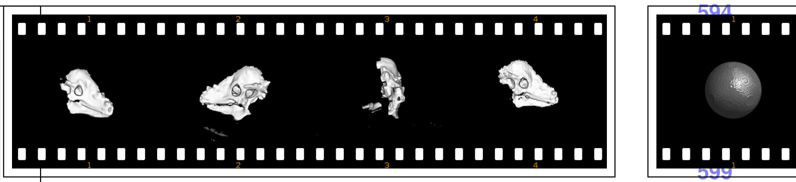

Figure 5: Quarterly interval snapshots of the Dinosaur Head reconstruction using **(L)** our method, and **(R)** the one proposed by Ren et al. [29].

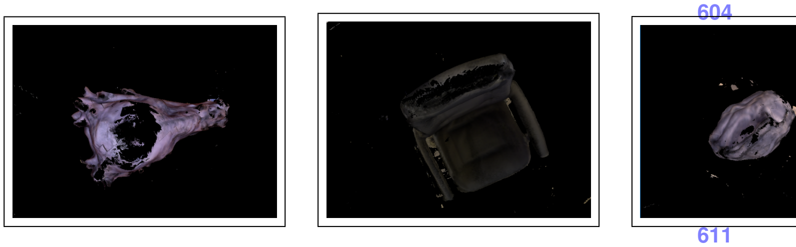

Figure 6: Closed reconstructions of **(L)** a Dinosaur Head, **(M)** a Chair, and **(R)** a Rock.

4. Results

To evaluate our system, we perform quantitative experiments on camera pose estimation accuracy, and qualitatively analyse the obtained reconstructions. Firstly, the pose estimation accuracy is evaluated via a well-established SLAM evaluation benchmark [23]. We like to point out that in traditional dense SLAM systems [17, 13, 12] – for which the benchmark is often employed – the entire contents of the visible scene are used for pose estimation, whereas in our system we rely only on points belonging to the object's surface. Whilst more challenging, this implicitly allows us to track the camera wrt. the object regardless of which of the two is subject to motion. Then, qualitative comparisons are drawn between the reconstructions attained by our system, and those of the method described in [29]. We evaluate our system on multiple frame sequences depicting objects of different sizes.

4.1. Pose Estimation Quality

In this section we present quantitative results of our systems' robustness in estimating the camera motion, by performing tracking against the reconstruction of a single object, instead of the whole scene. The trajectories estimated by our system demonstrate low tracking drift. We perform such evaluation on two sequences of the RGB-D SLAM Dataset [23] depicting static objects observed by a moving camera. Tracking is performed using purely geometric clues, by matching the current depth frame with a rendering of the reconstructed object using a projective ICP tracking approach [7].

At this point, it should be highlighted that our proposed system is at a disadvantage when compared to dense SLAM systems that utilise the entire scene geometry for pose optimisation, since we track the sensor pose against a subset of the observed scene. Nevertheless, as shown by the results in Figure 4, our system is able to robustly estimate trajectories close to the ground truth whilst using only the objects' geometric appearance. Quantitatively, the tracking accuracy is evaluated via the Absolute Trajectory Error (ATE) metric, as outlined by [23]. The cabinet reconstructed in the *freiburg3_cabinet* sequence is lacking in geometric features, as the object is mostly planar, and the small deficit in tracking quality is mostly due to this factor. However, our system remains able to estimate a fairly accurate trajectory. In the *freiburg3_teddy* sequence we determine a trajectory very close to the ground truth. Improvement over the accuracy in *freiburg3_cabinet* is due to the wider availability of geometrical features, such as curves in the teddy's body and head.

4.2. Qualitative Reconstruction Quality

In this section we present a qualitative comparison of our method vs. the approach by Ren et al. [29] in the reconstruction of closed object models. Each sequence is run through both systems; to evaluate the obtained results we regularly take snapshots of the reconstruction, in the case of our system, and the level set evolutions, in the case of [29]. Such snapshots are captured after each quarter of a sequence has been processed.

As depicted in Figure 5, our method is able to successfully reconstruct the Dinosaur Head, whereas the approach by Ren et al. fails to converge towards a feasible shape. In addition, Figure 6 demonstrates that our system is able to generate consistent models (unaffected by camera tracking drift) for a variety of sequences containing several loop closures. Failure of the competing method is also apparent for other sequences evaluated in this work, all presenting failure cases analogous to Figure 5b. Such examples will be presented in the supplementary materials.

The object reconstructions depicted in Figure 1 have been obtained from sequences in which a camera was moved in a loop around each object in order to generate a closed model.

5. Conclusion

As has been demonstrated in this paper and the accompanying supplementary materials, our system is efficacious in 3D object reconstruction. Our system is able to reconstruct closed object models on sequences for which an alternative, state of the art system [29] fails to converge to any reasonable solution. In addition, we show robust odometry on an established SLAM benchmark, in spite of the difficulty of tracking only the objects surface vs the entire scene.

References

[1] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, Feb 1992.

[2] Y. Cui, S. Schuon, S. Thrun, D. Stricker, and C. Theobalt. Algorithms for 3d shape scanning with a depth camera. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(5):1039–1050, May 2013.

[3] B. Curless and M. Levoy. A Volumetric Method for Building Complex Models from Range Images. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, pages 303–312, 1996.

[4] A. Dai, M. Nießner, M. Zollhöfer, S. Izadi, and C. Theobalt. Bundlefusion: Real-time globally consistent 3d reconstruction using online surface re-integration. *arXiv preprint arXiv:1604.01093*, 2016.

[5] N. Fioraio, J. Taylor, A. Fitzgibbon, L. D. Stefano, and S. Izadi. Large-Scale and Drift-Free Surface Reconstruction Using Online Subvolume Registration. pages 4475–4483, 2015.

[6] T. Gupta, D. Shin, N. Sivagnanadasan, and D. Hoiem. 3dfs: Deformable dense depth fusion and segmentation for object reconstruction from a handheld camera. *CoRR*, abs/1606.05002, 2016.

[7] O. Kähler, V. A. Prisacariu, and D. W. Murray. *Real-Time Large-Scale Dense 3D Reconstruction with Loop Closure*, pages 500–516. Springer International Publishing, Cham, 2016.

[8] K. Kolev, T. Brox, and D. Cremers. Robust variational segmentation of 3D objects from multiple views. In K. F. et al., editor, *Pattern Recognition (Proc. DAGM)*, volume 4174 of *LNCS*, pages 688–697, Berlin, Germany, Sept. 2006. Springer.

[9] M. Krainin, P. Henry, X. Ren, and D. Fox. Manipulator and object tracking for in-hand 3D object modeling. volume 30, pages 1311–1327, 2011.

[10] R.-G. Mihalyi, K. Pathak, N. Vaskevicius, T. Fromm, and A. Birk. Robust 3D object modeling with a low-cost RGBD-sensor and AR-markers for applications with untrained end-users. 66:1–17, 2015.

[11] K. S. Narayan, J. Sha, A. Singh, and P. Abbeel. Range Sensor and Silhouette Fusion for High-Quality 3D Scanning. pages 3617–3624, 2015.

[12] 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 *International Symposium on Mixed and Augmented Reality*, pages 127–136, 2011.

[13] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger. Real-time 3D Reconstruction at Scale using Voxel Hashing. *ACM Transactions on Graphics*, 32(6):169, 2013.

[14] K. Ohno, K. Kensuke, E. Takeuchi, L. Zhong, M. Tsubota, and S. Tadokoro. Unknown Object Modeling on the Basis of Vision and Pushing Manipulation. pages 1942–1948, 2011.

[15] P. Panteleris, N. Kyriazis, and A. A. Argyros. 3d tracking of human hands in interaction with unknown objects. In X. Xie, M. W. Jones, and G. K. L. Tam, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 123.1–123.12. BMVA Press, September 2015.

[16] P. Panteleris, N. Kyriazis, and A. A. Argyros. Recovering 3d models of manipulated objects through 3d tracking of hand-object interaction. In *IEEE International Conference on Computer Vision Workshops (OUI 2015 - ICCVW 2015)*, Santiago, Chile, November 2015. IEEE.

[17] V. A. Prisacariu, O. Kahler, M. M. Cheng, C. Y. Ren, J. Valentin, P. H. S. Torr, I. D. Reid, and D. W. Murray. A Framework for the Volumetric Integration of Depth Images. *ArXiv e-prints*, 2014.

[18] C. Y. Ren, V. Prisacariu, O. Kaehler, I. Reid, and D. Murray. 3d tracking of multiple objects with identical appearance using rgb-d input. In *Proceedings of the 2014 2Nd International Conference on 3D Vision - Volume 01*, 3DV '14, pages 47–54, Washington, DC, USA, 2014. IEEE Computer Society.

[19] H. Roth and M. Vona. Moving Volume KinectFusion. In *British Machine Vision Conference*, pages 1–11, 2012.

[20] S. Rusinkiewicz and M. Levoy. Efficient variants of the icp algorithm. In *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*, pages 145–152, 2001.

[21] M. D. Shuster. Survey of attitude representations. *Journal of the Astronautical Sciences*, 41:439–517, Oct. 1993.

[22] M. Slavcheva, W. Kehl, N. Navab, and S. Ilic. SDF-2-SDF: Highly Accurate 3D Object Reconstruction. In *European Conference on Computer Vision (ECCV)*, 2016.

[23] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A Benchmark for the Evaluation of RGB-D SLAM Systems. In *International Conference on Intelligent Robots and Systems*, pages 573–580, 2012.

[24] D. Tzionas and J. Gall. 3D Object Reconstruction from Hand-Object Interactions. pages 729–737, 2015.

[25] T. Weise, T. Wismer, B. Leibe, and L. V. Gool. In-hand scanning with online loop closure. In *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, pages 1630–1637, Sept 2009.

[26] T. Whelan, M. Kaess, M. Fallon, H. Johannsson, J. Leonard, and J. McDonald. Kintinuous: Spatially Extended KinectFusion. Technical Report MIT-CSAIL-TR-2012-020, MIT, 2012.

[27] T. Whelan, M. Kaess, H. Johannsson, M. Fallon, J. J. Leonard, and J. McDonald. Real-time large scale dense

RGB-D SLAM with volumetric fusion. 34(4-5):598–626, 2015.

[28] T. Whelan, S. Leutenegger, R. F. Salas-Moreno, B. Glocker, and A. J. Davison. ElasticFusion: Dense SLAM Without A Pose Graph. 2015.

[29] C. Yuheng Ren, V. Prisacariu, D. Murray, and I. Reid. Star3d: Simultaneous tracking and reconstruction of 3d objects using rgb-d data. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2013.

[30] M. Zeng, F. Zhao, J. Zheng, and X. Liu. A Memory-Efficient KinectFusion Using Octree. In *Computational Visual Media*, pages 234–241. Springer Berlin Heidelberg, 2012.

[31] Q.-Y. Zhou and V. Koltun. Dense Scene Reconstruction with Points of Interest. *ACM Transactions on Graphics*, 32(4):112, 2013.