Implicit Object Mapping With Noisy Data

Jad Abou-Chakra
QUT Centre for Robotics
Queensland University of Technology
Brisbane QLD, Australia, 4000
jad.chakra@hdr.qut.edu.au

Feras Dayoub
School of Computer Science
University of Adelaide
Adelaide SA, Australia, 5005
feras.dayoub@adelaide.edu.au

Niko Sünderhauf
QUT Centre for Robotics
Queensland University of Technology
Brisbane QLD, Australia, 4000
niko.suenderhauf@qut.edu.au

***Abstract*—Modelling individual objects as Neural Radiance Fields (NeRFs) within a robotic context can benefit many downstream tasks such as scene understanding and object manipulation. However, real-world training data collected by a robot deviate from the ideal in several key aspects. (i) The trajectories are constrained and full visual coverage is not guaranteed – especially when obstructions are present. (ii) The poses associated with the images are noisy. (iii) The objects are not easily isolated from the background. This paper addresses the above three points and uses the outputs of an object-based SLAM system to bound objects in the scene with coarse primitives and – in concert with instance masks – identify obstructions in the training images. Objects are therefore automatically bounded, and non-relevant geometry is excluded from the NeRF representation. The method's performance is benchmarked under ideal conditions and tested against errors in the poses and instance masks. Our results show that object-based NeRFs are robust to pose variations but sensitive to the quality of the instance masks.**

I. Introduction

Robots that construct semantically meaningful maps rich with geometric data can better facilitate decision making and control tasks [7, 5, 3]. Creating geometrically expressive object representations that would populate such a map is a critical stepping stone towards increasing the utility and flexibility of robots [7, 4, 10]. Neural Radiance Fields (NeRFs) [8] are recent advancements in implicit representations that have become popular due to their remarkable success on the view-synthesis task. In this paper, we show how NeRFs can be used as object rather than scene representations and analyze their sensitivity to noisy and constrained input typical of robotic applications. We demonstrate that NeRFs are a natural extension to object-based SLAM systems and that they are complimentary – object-based SLAM provides tractability and NeRFs provide accuracy. NeRFs are neural networks that are trained with posed images and represent a manually configured area with high fidelity. In contrast, object-based SLAM lacks geometric fidelity but is capable of quickly associating input images with poses and automatically detecting areas of interest in a scene. Using the two systems in concert to remove the disadvantages of either is the approach we take in this paper. We present such a method and investigate how well it performs under constraints and noise typical in robotic contexts.

II. Method

We assume the presence of an object-based SLAM system that localises the camera, identifies objects in the scene, and bounds them using a course primitive. We present a method that extends this system and enriches its representation of objects through the use of Neural Radiance Fields (NeRFs). We use the hash encoding from [9] to make training and inference as fast as possible. Given a set of images I_i and their corresponding poses X_i, depth maps D_i, and instance masks S_i, we construct a single NeRF for every object j present in the scene – Figure 1. A ray $\boldsymbol{r}$ that intersects an image plane i at a pixel coordinate (u, v) is associated with the color $\boldsymbol{c}_{gt}(\boldsymbol{r}) = I_i(u, v)$, the depth $d_{gt}(\boldsymbol{r}) = D_i(u, v)$, and the instance ID $S_i(u, v)$.

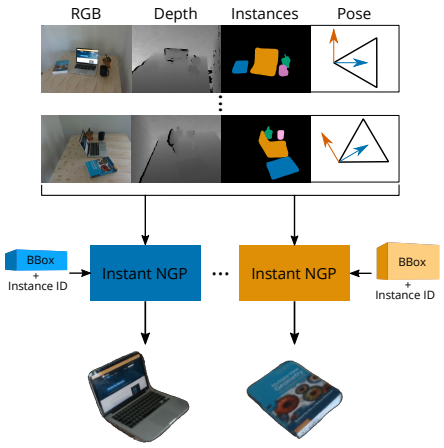

Fig. 1. We generate a neural radiance field (NeRF) with a hash encoding (Instant NGP) for each object in the scene. We use noisy instance masks and loose bounding boxes – assumed to be provided by an object-based SLAM system – to bound each NeRF and isolate the object from its background. A scene with 4 objects can thus be decomposed into 4 NeRFs, each representing the geometry of a single object.

The object-centric nature of the task presents two challenges that do not appear when mapping whole scenes. The first challenge is bounding the NeRF to an area so that its representational power can be focused on the object of interest. In this, we rely on the object-based SLAM system to provide a bounding box B_j that loosely contains the object. We use a ray-box intersection algorithm to ignore rays that do not pass through the bounding box and to limit sampling to the points therein.

The second challenge is to isolate that object in the presence of clutter. The aim is to construct the object while suppressing

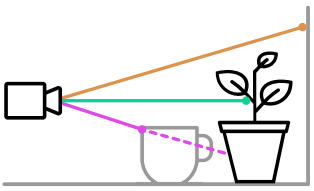

Fig. 2. Each training image can be decomposed into three regions through which rays are cast: (i) Rays cast through the positive region – shown as green – represent the object and have their computed color optimized towards the groundtruth color in the image. (ii) Rays cast through the negative region – shown as orange – represent the background and are optimized towards a zero density distribution. (iii) Rays cast through the masked region – shown in pink – represent possible obstructions and are not included in the training.

any other geometry around it. Rays $\mathcal{R}_p$ that are known to hit the object should encourage geometry to be formed along them. Conversely, rays $\mathcal{R}_n$ that do not hit the object should discourage it. Rays $\mathcal{R}_m$ that are obstructed from hitting the object by another are not included in the training because their groundtruth values are not known – the colors that correspond to them are not of the object of interest but rather of the object that is obstructing it. Therefore, rays are cast from each training image through three different regions (illustrated in Figure 2): (i) the negative region – corresponding to the background – that discourages geometry formation, (ii) the positive region – corresponding to the object – that promotes it, and (iii) the masked region – corresponding to potential obstacles – that does neither. This method assumes that the geometries obstructing the training views are recognized by the instance masks and the object-based SLAM systems.

Our photometric loss is formulated as:

$$L_{\text{rgb}} = \sum_{\boldsymbol{r}\in\mathcal{R}} \|\boldsymbol{e}_{\text{rgb}}(\boldsymbol{r})\|_2 \tag{1}$$

where

$$\boldsymbol{e}_{\text{rgb}}(\boldsymbol{r}) = \begin{cases} \hat{C}(\boldsymbol{r}) - \boldsymbol{c}_{\text{gt}}(\boldsymbol{r}) & \text{if} \quad \boldsymbol{r} \in \mathcal{R}_p \\ \hat{C}(\boldsymbol{r}) - \boldsymbol{c}_{\text{random}} & \text{if} \quad \boldsymbol{r} \in \mathcal{R}_n \\ 0 & \text{if} \quad \boldsymbol{r} \in \mathcal{R}_m \end{cases} \tag{2}$$

$\mathcal{R}$ is the set of all rays that pass through the training images. $\boldsymbol{c}_{gt}(\boldsymbol{r})$ is the color of the pixel in the training image that the ray $\boldsymbol{r}$ intersects. $\boldsymbol{c}_{\text{random}}$ is a vector drawn from a uniform distribution $U(0,1)$.

In some experiments, we show how depth supervision affects the results. In those cases, the depth loss is given as:

$$L_{\text{depth}} = \sum_{\boldsymbol{r}\in\mathcal{R}_d} |\boldsymbol{e}_{\text{depth}}| \tag{3}$$

where

$$\boldsymbol{e}_{\text{depth}}(\boldsymbol{r}) = \begin{cases} \hat{D}(\boldsymbol{r}) - \boldsymbol{d}_{\text{gt}}(\boldsymbol{r}) & \text{if } \boldsymbol{r} \in \mathcal{R}_p \text{ and } T_N < 10^{-4} \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

$\mathcal{R}_d$ is the set of rays for which groundtruth depth information is available. The method is not sensitive to the empirically chosen threshold condition $T_N < 1e^{-4}$.

Joint Optimization: The learnable parameters Φ of the NeRF framework and the poses X of the cameras are jointly optimized.

$$\Phi^*, X^* = \arg\min_{\Phi,X} L_{\text{rgb}} + w_{\text{depth}} L_{\text{depth}} \tag{5}$$

III. Evaluation

Metrics In contrast to those works that use NeRFs for novel view synthesis, we evaluate primarily on the accuracy of the geometry rather than that of the color. We differentiate between areas which have been correctly categorized as part of the object of interest and areas which have not. For correctly categorized areas, we use the mean average depth error (MAE) between the groundtruth and rendered depth maps to measure accuracy. We also use an intersection-over-union (IoU) between the ideal and rendered instance masks to measure how much of the total shape is represented in the NeRF.

Fig. 3. Real scene shown on the right and its synthetic counterpart shown on the left.

Fig. 4. Renders from four object NeRFs extracted from a real scene show various geometry artifacts that may be due to insufficient view coverage, noisy instance masks, and inaccuracies in the poses. The training data is taken from a constrained trajectory and instance masks are produced by MaskRCNN. Depth supervision is not used in this example. Large simple geometries – as seen in the laptop and book renders – are reconstructed more accurately than their counterparts. In general, the objects are well-isolated from their background, however attributing the artifacts to different sources of error is difficult in real scenes. This begs the question: "To what extent does each noise factor contribute to imperfections in the NeRF?"

The method is run on the real scene shown in Figure 3 and qualitative data is collected and shown in Figure 4. A camera is used to capture 100 images along a constrained trajectory. In the absence of a robust open-source implementation of object-based SLAM, loose bounding boxes are manually estimated around objects. Image poses are estimated with a SLAM system [1] and instance masks are generated by MaskRCNN [6]. Because groundtruth data is unavailable when using real-world data, it is difficult to quantitatively assess the method

TABLE I
EFFECT OF NOISY DATA AND DEPTH SUPERVISION ON THE RECONSTRUCTION OF OBJECTS FROM A CONSTRAINED TRAJECTORY.

Masks	Depth	Poses	Depth MAE (cm)	IoU (%)
Ideal	True	Ideal	0.5 ± 0.3	98 ± 1.5
	True	ORB-SLAM	0.6 ± 0.3	98 ± 0.6
	False	Ideal	1.1 ± 0.3	98 ± 0.8
	False	ORB-SLAM	1.4 ± 0.3	98 ± 1.1
MaskRCNN	True	ORB-SLAM	1.2 ± 0.1	85 ± 5.8
		Ideal	1.3 ± 0.2	85 ± 7.4
	False	ORB-SLAM	2.3 ± 0.4	83 ± 6.3
		Ideal	2.3 ± 0.4	82 ± 7.8

and to understand which sources of error are contributing negatively to the reconstruction. Therefore, we replicate the real scene in Blender [2] – Figure 3 – and measure the reconstruction quality of the object NeRFs under various conditions. The affects of using depth supervision, ORBSLAM, and MaskRCNN on the synthetic scene with a constrained camera trajectory are shown in Table I. We observe that most of the error is due to the use of MaskRCNN.

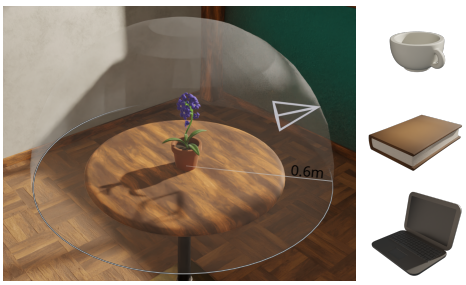

Fig. 5. Training images are generated from a camera looking at the center of the table and placed on the sphere shown. The four table-top models "bluebell", "laptop", "book", and "cup" are used to calculate confidence intervals.

To further analyze the relationship between noise in the instance masks and poses, we construct ideal scenes of the four objects using well distributed viewpoints – Figure 5. Three experiments are conducted. In the first experiment, we show how object NeRFs perform under ideal conditions. The results in Figure 7 reveal that the reconstruction quality plateaus to a baseline of approximately 0.6cm as the number of training views increase. With the baseline calculated, the second experiment progressively adds noise to the ideal instance masks used in training. Noise is introduced by removing or adding patches to the mask border until a desired IoU is met. The outputs of this process are shown in Figure 6. The results of the second experiment – Figure 8 – confirm that object NeRFs are sensitive to the quality of the instance masks. Increasing the number of images in the training set alleviates that sensitivity to some degree. The final experiment – Figure 9 – adds noise to the poses of the camera and shows that object NeRFs have a remarkable resilience to rotational and translational errors. On the scenes tested, deviations of up to 2 cm of translation and 3° of rotation can be tolerated – which is inline with what is expected from SLAM systems.

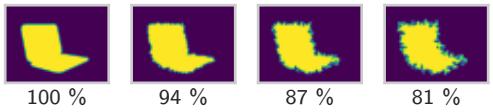

Fig. 6. Example outputs from the instance noise generator. This is used to controllably introduce noise unto ideal instance masks and analyze its affect on the NeRF reconstruction. The numbers shown are the intersection-over-union of the resultant mask relative to the groundtruth.

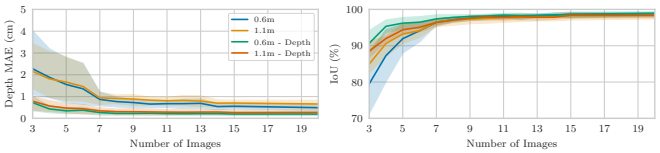

Fig. 7. NeRFs – representing one of four tabletop objects placed in an indoor scene – are trained from an increasing number of well-distributed posed images with groundtruth instance masks. The images are taken from the top half of a sphere with radius 0.6m and 1.1m. Depth supervision is enabled for two of the experiments. The tabletop objects can be reconstructed with an accuracy under 1 cm at 98% IoU. The number of training images has a diminishing effect on the quality of the recontruction whereas depth supervision gives an overall increase in accuracy (< 0.5 cm at 98% IoU).

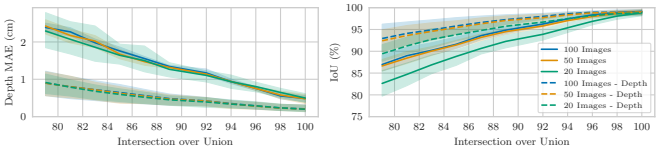

Fig. 8. Well-distributed images with groundtruth poses and noisy instance masks are used to reconstruct the same tabletop objects from the baseline experiment. The reconstruction quality deteriorates quickly with increasing noise levels. Depth supervision is more effective than increasing the number of training images at slowing the deterioration.

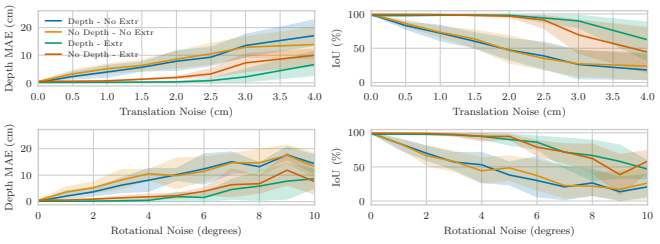

Fig. 9. Well-distributed images with ideal instance masks and inaccurate poses are used to reconstruct the same tabletop objects from the baseline experiment. Translation and rotation errors are treated separately. The resilience to either is shown when depth supervision is included and when the extrinsics – labelled "Extr" in legend – are allowed to be optimized.

IV. CONCLUSION

NeRFs are exciting representations that can be used to extend current object-based SLAM systems. The two are structurally and functionally symbiotic. In this work, we saw that object NeRFs are robust to noise in the extrinsics but not robust to noise in the instance masks. Future work should address the ability of NeRFs to deal with this kind of noise which is characteristically not consistent in 3D.

REFERENCES

[1] Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, and Juan D Tardós. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Transactions on Robotics*, 37(6):1874–1890, 2021.

[2] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. URL http://www.blender.org.

[3] Danny Driess, Jung-Su Ha, Marc Toussaint, and Russ Tedrake. Learning models as functionals of signed-distance fields for manipulation planning. In *Conference on Robot Learning*, pages 245–255. PMLR, 2022.

[4] Philipp Erler, Paul Guerrero, Stefan Ohrhallinger, Niloy J. Mitra, and Michael Wimmer. Points2surf learning implicit surfaces from point clouds. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 108–124, Cham, 2020. Springer International Publishing. ISBN 978-3-030-58558-7.

[5] Wei Gao and Russ Tedrake. kpam-sc: Generalizable manipulation planning using keypoint affordance and shape completion. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6527–6533. IEEE, 2021.

[6] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[7] Eric Jang, Coline Devin, Vincent Vanhoucke, and Sergey Levine. Grasp2vec: Learning object representations from self-supervised grasping. *arXiv preprint arXiv:1811.06964*, 2018.

[8] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, pages 405–421. Springer, 2020.

[9] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *arXiv:2201.05989*, January 2022.

[10] Anthony Simeonov, Yilun Du, Andrea Tagliasacchi, Joshua B Tenenbaum, Alberto Rodriguez, Pulkit Agrawal, and Vincent Sitzmann. Neural descriptor fields: Se (3)-equivariant object representations for manipulation. *arXiv preprint arXiv:2112.05124*, 2021.