Efficient Volumetric Mapping Using Depth Completion
With Uncertainty for Robotic Navigation

Marija Popović*,1, Florian Thomas*,1, Sotiris Papatheodorou1, Nils Funk1, Teresa Vidal-Calleja2, Stefan Leutenegger1

Abstract—In robotic applications, a key requirement for safe and efficient motion planning is the ability to map obstacle-free space in unknown, cluttered 3D environments. However, commodity-grade RGB-D cameras commonly used for sensing fail to register valid depth values on shiny, glossy, bright, or distant surfaces, leading to missing data in the map. To address this issue, we propose a framework leveraging probabilistic depth completion as an additional input for spatial mapping. We introduce a deep learning architecture providing uncertainty estimates for the depth completion of RGB-D images. Our pipeline exploits the inferred missing depth values and depth uncertainty to complement raw depth images and improve the speed and quality of free space mapping. Evaluations on synthetic data show that our approach maps significantly more correct free space with relatively low error when compared against using raw data alone in different indoor environments; thereby producing more complete maps that can be directly used for robotic navigation tasks. The performance of our framework is validated using real-world data.

I. INTRODUCTION

In recent years, depth sensors have become a core component in a variety of robotic applications, including scene reconstruction, exploration, and inspection. However, commodity-grade RGB-D cameras, such as Microsoft Kinect and Intel RealSense, suffer from limited range and produce images with noise and missing data in view of surfaces that are too shiny, glossy, bright, or simply too far away. In robotic scenarios, this may lead to inefficient and inaccurate mapping performance when only the raw sensor data is used.

This paper examines the problem of depth completion, which aims at filling in holes and reducing the noise in relatively dense depth images. Our motivation is to improve upon the quality of raw images to build more complete spatial maps for robotic tasks in cluttered 3D environments.

Recently, several deep learning-based approaches for depth completion using RGB-D images have been proposed [1, 2] which effectively use colour information to enhance depth. However, propagating the completed depth into robotic frameworks for Simultaneous Localisation and Mapping (SLAM) remains an open challenge. A key issue is associating the completed areas with reliable measures of depth uncertainty, such that they can be used as an input for probabilistic mapping. Though several works have tackled uncertainty estimation for depth completion, they do not address using this information for 3D reconstruction [3] and largely focus on LiDaR-based sensors in outdoor environments [4–6].

To address this, we propose a new pipeline for mapping with depth completion; thus bridging the gap between computer vision algorithms and robotic applications. Inspired by the methods of Huang et al. [2], we introduce a network architecture that jointly predicts depth and depth uncertainty from RGB-D images by leveraging principles of Bayesian deep learning. Our approach exploits the processed images online as an additional input in the occupancy-based volumetric framework of Vespa et al. [7] and Funk et al. [8]. This procedure enables us to produce maps with more discovered obstacle-free space in the environment compared to using the raw images alone, as visualised in Figure 1, which is useful for robotic navigation tasks.

The contributions of this work are:

1) A new deep learning architecture providing uncertainty estimates for the depth completion of RGB-D images.

2) The integration of our network in the volumetric map-
ping framework of Vespa et al. [7] and Funk et al. [8]. We use the completed depth images with the predicted depth uncertainties in online probabilistic occupancy mapping to obtain more complete free space maps for robotic navigation tasks.

3) The extensive evaluation of our framework using synthetic and real-world datasets demonstrating its performance.

We plan to open-source our network implementation for usage and further development by the community.

II. RELATED WORK

Algorithms for depth estimation and spatial mapping play a key role in many robotic applications and are the subject of a large and growing body of research. In this section, we review previous studies most related to our work.

Traditional methods for depth completion adopt hand-crafted kernels or features to compute the missing values [9, 10]. More recent algorithms [1, 2, 4–6, 11, 12] exploit deep learning for improved performance and generalisation capabilities. Our work focuses on the task of guided depth completion, where the goal is to predict the dense depth values at every pixel based on the raw depth and a paired colour image. Uhrig et al. [11] propose a sparse convolution layer which explicitly handles missing data to allow for inputs with varying degrees of sparsity. In a similar problem setup, Ma and Karaman [12] use an encoder-decoder network to combine RGB and depth information within the underlying feature space. Recently, Eldesokey et al. [6] present a network based on normalised convolution layers which supports very sparse depth inputs and also provides confidence measures for the depth predictions. However, the aforementioned studies focus on completing sparse LiDaR-based data in outdoor scenarios and are thus not applicable to the types of degradation obtained with commodity-grade RGB-D cameras, as considered in our work.

For hole-filling with RGB-D cameras, Zhang and Funkhouser [1] exploit the encoder-decoder architecture using dense occlusion boundaries and surface normals predicted from the colour image as secondary features to aid depth completion. Their approach involves an expensive loss optimisation step, making it unsuitable for real-time mapping. Building upon their ideas, Huang et al. [2] introduce a network with a self-attention mechanism and boundary consistency to improve completion accuracy and speed. We propose an extension of their architecture which also predicts the uncertainties in the completed depth.

While significant work has been done on depth completion in the 2D image plane, applying these concepts to 3D mapping in robotics is a relatively unexplored area of research. Recently, Teixeira et al. [4] introduced a depth completion algorithm for real-time aerial robotic applications. Similar to us, they obtain probabilistic depth predictions by estimating pixelwise uncertainties. However, they consider LiDaR-based sensing and do not use the completed images for 3D mapping. Most resembling our work is the approach of Fehr et al. [13], which uses an augmented depth sensor based on sparse inputs for robotic navigation. They show that their system uncovers more free space in the environment when compared against using raw depth alone, thus leading to better planning performance. Although our work shares the same motivation, a key difference is that, instead of feeding the completed depth directly into a dense mapping framework, we adopt a fully probabilistic strategy based on the depth uncertainties provided by our new modified network.

Uncertainty in depth completion is crucial as it provides a reliability measure for fusing new predicted measurements into the map. One approach is to exploit confidence as a process internal to deep learning [14] to obtain more accurate dense depth outputs, i.e. by leveraging uncertainty as a weight map within the prediction network architecture. An alternative is to treat uncertainty as an auxiliary output of the network to obtain pixelwise uncertainties [3] or confidence maps [4–6]. We follow the second class of approaches to extract explicit uncertainty values as inputs for mapping. Although, like us, Kendall and Gal [3] learn the uncertainty in depth regression problems, to our knowledge, no prior work has applied these ideas in the context of probabilistic robotic mapping.

Another line of work focuses on volumetric scene completion directly in 3D space. For example, Song et al. [15] predict volumetric occupancy and semantic labels from a single-view depth map. Dai et al. [16] introduce an approach for completing 3D geometry with per-voxel semantic labels from partial scans. However, as these methods require significant computational processing, they are not viable for real-time, online applications.

III. APPROACH

In this section, we propose a new approach for efficient probabilistic tracking and mapping using completed depth images with predicted depth uncertainty. A system overview is depicted in Figure 2. As shown, we process the raw images from a RGB-D camera using a depth completion pipeline online to improve the input for SLAM. Note that, while our approach is applicable for any SLAM scenario, in this paper, we focus on mapping only to show improvements for scene reconstruction in unknown environments. The following sub-sections describe our strategy for probabilistic depth completion before outlining the SLAM framework.

A. Network Architecture

Our goal is to complete the depth channel of an RGB-D image and predict the associated pixelwise depth uncertainty, to be used as input for probabilistic tracking and mapping. To
achieve this, we develop a pipeline based on the depth completion network proposed by Huang et al. [2]. An overview of the depth completion sub-system is shown in Figure 3. Figure 4 details our new network architecture.

The main features of the depth completion network are the use of a self-attention mechanism and boundary consistency to produce depth maps with high quality and structure. Following Zhang and Funkhouser [1], we predict surface normals and occlusion boundaries from the raw RGB-D image and use them as additional input features to the network. To estimate surface normals, we employ the hierarchical RGB-D fusion network of Zeng et al. [17], which has state-of-the-art performance. Our boundary estimation network is based on the approach of Zhang et al. [18], using only RGB channels. The normals and boundaries are concatenated with the raw RGB-D image and used for the learning task.

To predict depth uncertainty, we leverage the Bayesian deep learning concepts of Kendall and Gal [3]. Our key idea is to introduce a second decoder on the output of the network to learn the mapping to the input uncertainty in the completed depth. This is illustrated in the bottom branch of the architecture in Figure 4. We use a SoftPlus activation function (purple layer) to constrain the output uncertainty to be non-negative. By using a network with two different branches, the encoder of the original network captures the latent features common to both the completed output depth and associated uncertainty, before they are processed separately to account for individual information. Moreover, we increase the number of channels per layer to 64 from 48 in the original network to provide a larger latent space for learning in the dual prediction task.

B. Loss Function

We assume a Gaussian likelihood to model our aleatoric uncertainty [3]. Our loss function for depth completion with uncertainty is then the weighted sum of errors:

\[
L = \frac{1}{N} \sum_{p=1}^{N} \frac{1}{\sigma(x_p)^2} [y_p - f(x_p)]^2 + \log(\sigma(x_p)^2) \\
+ \lambda_{BC} ||\delta_{\text{Sobel},p} - \delta_{\text{Sobel}}(f(x_p))|| \\
+ \lambda_{\text{SSIM}} \text{SSIM}(y_p, f(x_p)) ,
\]

where \(N\) is the number of pixels \(p\) in an image, \(x_p\) is the input vector of features for the depth completion network (raw depth, RGB, estimated normals and boundaries), \(y_p\) is the ground truth depth, \(f(\cdot)\) and \(\sigma(\cdot)\) are the completed depth and associated uncertainty output by the network, respectively, following directly from the negative log likelihood assuming Gaussian uncertainty, and \(\lambda_{BC}\) and \(\lambda_{\text{SSIM}}\) are tunable parameters. We only consider pixels in areas observed, i.e. non-zero, in the ground truth depth image.

Following Huang et al. [2], we also include a structural related loss based on the Structural Similarity Index (SSIM) measure [19] to reduce distortion and enhance structural quality (third term) and a boundary related loss (fourth term) to enforce boundary consistency. The latter is computed by training a model to learn the Sobel edges associated with the completed depth supervised by those computed from the ground truth depth. The different components of the loss function are depicted in the dashed blue box in Figure 3.

C. Mapping

We use an occupancy map to model the environment, as this representation is suitable for integrating noisy sensor measurements and explicitly captures free space for robotic path planning applications. Specifically, our approach leverages the multi-resolution occupancy mapping (‘MultiResO-Fusion’) and dense volumetric SLAM framework from Funk et al. [8]. This pipeline is an extension of supereight [7] that allows for integrating data at multiple levels of the octree.
and explicitly maps the free space, while retaining real-time performance for robotic applications.

To explain the role of depth uncertainty for mapping in our approach, we briefly overview the ‘MultiresOFusion’ probabilistic inverse sensor model, which is used to fuse new depth measurements into the map. The inverse sensor model is inspired by Loop et al. [20] but uses a piecewise linear function instead of a B-spline in log-odds space. The model produces probabilities expressed in log-odds directly to match the representation of occupancy probabilities used in the map. Given a noisy depth measurement \( z_r \), we assume its standard deviation is given by:

\[
s_{r}(z_{r}) = \min \left( \max \left( \frac{k_{\sigma} z_{r}^{2}}{\sigma_{\min}}, \sigma_{\max} \right) \right),
\]

as shown in the left plot in Figure 5, where \( k_{\sigma} \) and \( \sigma_{\min} \) are constants. This corresponds to a triangulation-based depth model for fusing new data into the map. The inverse sensor model is used to compute the log-odds occupancy probability given the radial distance \( r \) from the camera and the depth measurement \( z_{r} \), as shown in the right plot. The log-odds occupancy probability starts from a constant value \( l_{\min} \) near the camera, gradually increases as distance increases, reaches 0 at \( z_r \), and peaks halfway through the surface thickness \( \tau(z_r) \). The surface thickness is computed as:

\[
\tau(z_r) = \min \left( \max \left( k_{\tau} z_{r}, \tau_{\min} \right), \tau_{\max} \right),
\]

where \( k_{\tau} \), \( \tau_{\min} \) and \( \tau_{\max} \) are constants. No voxels further than \( z_r + \tau_r \) from the camera are updated. Larger values of \( \sigma_r \) result in a more gradual increase of the occupancy probability. In this work, our aim is to exploit the completed depth and depth uncertainty provided by our network in Section III-A to complement the raw sensor data captured by this model and thereby improve mapping performance. Specifically, we propose using the depth uncertainty predicted by our network instead of the measurement uncertainty above for the completed depth areas.

A. Training Procedure

We trained our depth completion system end-to-end on Matterport3D, a large-scale RGB-D dataset [21] representative of an indoor exploration task. For training, (complete) ground truth depth was obtained from Zhang and Funkhouser [11] based on multi-view reconstruction. Unless otherwise specified, 129816 and 36252 images from the dataset were used for training and testing, respectively, with an image resolution of 320 px × 240 px. We used the Adam optimiser with a weight decay of \( 10^{-3} \), a learning rate of \( 10^{-5} \), and set \( \lambda_{SSIM} = \lambda_{BC} = 1 \) in Equation (1). The models were implemented in PyTorch and training was done on a NVIDIA GeForce RTX 2080 Ti GPU with 16GB of RAM. On this machine, one forward pass through the pipeline takes \( \sim 0.2 \) s.

B. Ablation Study

Our first aim is to evaluate our new two-decoder network for depth completion with uncertainty. To this end, an ablation study is conducted to investigate the benefits of separating the two outputs in the proposed architecture and training the model end-to-end for both depth completion and uncertainty. We compare: (i) our proposed architecture with two decoders (Figure 4); (ii) the original architecture of Huang et al. [2] simply extended with a single shared output decoder for depth and uncertainty; (iii) a smaller variant of our architecture, using 48 channels per layer as in [2] instead of 64; and (iv) the same architecture as in (iii), but freezing the weights of the encoder and depth completion decoder parts using the trained model in [2], such that only the uncertainty output features are learned. Apart from (iv), we initialised each model with random weights, then let it train and report the best epoch. We also experimented with initialising (iii) using the pretrained weights from (iv) for training and confirmed that this has no significant impact on the final results, but does speed up convergence.

To evaluate prediction accuracy, we consider the Root Mean Squared Error (RMSE), the Mean Absolute Error (MAE), and the percentages of predicted pixels \( p_{\text{true}} \) within an interval \( \delta = \left| p_{\text{true}} - p_{\text{pred}} \right| \), where \( p_{\text{true}} \) are the corresponding pixel values obtained from the ground truth image and \( \delta \in \{1.05, 1.10, 1.25, 1.25^2, 1.25^3\} \). To evaluate the quality of the uncertainty, we measure the Area Under the Sparsification Error curve (AUSE). As explained by Ilg et al. [22], this metric captures the correlation between the estimated uncertainty and prediction error. Our AUSE values are computed based on all pixels in the test set.

Table I summarises our results. With the lowest AUSE, the single-decoder network produces the best uncertainty measure at the cost of lowest prediction accuracy. In contrast, training only a second decoder yields low error, but poor uncertainty, as the shared encoder weights are fixed and optimised for the depth completion problem. By increasing the model latent space and training the network end-to-end, our proposed model obtains relatively low AUSE without compromising on prediction accuracy with respect to the original implementation. For visual validation, Figure 6 presents example results of our proposed larger 2-decoder network.
TABLE I: Comparison of our proposed 2-decoder, 64-channel network for depth completion with uncertainty (top) against benchmarks derived from Zhang and Funkhouser [1] on the Matterport3D test set. Our architecture achieves good depth uncertainty measures without compromising on depth prediction accuracy. The number of channels per layer in the network is in parentheses.

| Model                        | RMSE (m) | AUSE | MAE (m) | 1.05↑ | 1.10↑ | 1.25↑ | 1.25^2↑ | 1.25^3↑ |
|------------------------------|----------|------|---------|-------|-------|-------|---------|---------|
| (i) 2 decoders (64) (ours)  | 0.3154   | 0.1849 | 0.1282  | 85.34 | 90.01 | 93.92 | 96.55   | 97.85   |
| (ii) 1 decoder (48)         | 0.3484   | 0.1771 | 0.1538  | 80.19 | 85.08 | 90.55 | 95.16   | 97.22   |
| (iii) 2 decoders (48)       | 0.3187   | 0.2115 | 0.1335  | 85.22 | 89.61 | 93.52 | 96.29   | 97.68   |
| (iv) 2 decoders (48), depth weights from [2] | 0.3166   | 0.1996 | 0.1282  | 80.30 | 88.69 | 94.20 | 96.89   | 98.02   |

model on the Matterport3D test set. This way, we achieve high-quality completed depth with reliable, and, importantly, consistent uncertainty estimates, which we exploit in the next sub-sections to improve probabilistic mapping performance.

C. Evaluation on Synthetic Data

We perform a quantitative evaluation of our approach for occupancy mapping using trajectory sequences from the synthetic RGB-D dataset InteriorNet [23], in which ground truth depth images and pose data are available. Our aim is to show that mapping using the network predicted depth and uncertainty leads to more complete final maps and a greater volume of free space discovered in the environment, which is a key requirement for safe robotic planning and navigation.

The ground truth depth from InteriorNet contains no measurement noise. To simulate realistic noisy depth images, we degrade the ground truth following the quadratic noise model for the Kinect sensor developed by Nguyen et al. [24]:

\[
\sigma_z(z) = 0.0012 + 0.0019 (z - 0.4)^2, \tag{4}
\]

where \(\sigma_z\) is the standard deviation of lateral noise in metres at a pixel and \(z\) is the corresponding depth measurement in metres. Additionally, a Gaussian filter with a standard deviation of 0.5 px is applied on the depth image to blur the noise in the images between adjacent pixels.

The ground truth depth does not contain occlusion holes or any missing depth measurements. To create missing data for depth completion, we set practical sensing limits of 0.8 m–6 m beyond which depth readings are zero. To generate occlusion holes and remove measurements based on the context and structure of the scene within this range, e.g. on textureless/reflective surfaces, we use the generative adversarial framework of Atapour-Abarghouei et al. [25], which learns to predict depth holes from RGB images. This network is trained on the same Matterport3D training set as specified in Section IV-A, since it contains real RGB/raw depth image pairs. We then train CycleGAN, an unpaired image-to-image translation network [26] to learn the visual domain shift between Matterport3D and InteriorNet using 10000 random images from each dataset, and apply the hole predictor on InteriorNet RGB images translated to the Matterport3D style. This procedure allows us to transfer the learnt structures of real holes to the synthetic dataset and thus generate realistically corrupted depth images.

For depth completion, we use our trained 2-decoder, 64-channel network from Section IV-B, further fine-tuned on the corrupted InteriorNet depth data with 110000 and 28000 images for training and testing, respectively. The images are downsampled to 320 px \(\times\) 240 px and we apply the same optimisation algorithm as detailed in Section IV-A.
For spatial mapping, we use supereight ‘MultiresO-Fusion’ \cite{8} with a voxel resolution of 0.0146 m in a 15 m × 15 m × 15 m volume. We use the inverse sensor model in Figure 5 to integrate raw depth data into the map, setting the constants \( k_r, r_{\text{min}}, \) and \( r_{\text{max}} \) as 0.026, 0.06 m, and 0.16 m, respectively. To capture measurement uncertainty, we consider the quadratic uncertainty model in Equation (2) with \( k_r = 0.0016 \) m, \( \sigma_{\text{min}} = 0.005 \) m, and \( \sigma_{\text{max}} = 0.02 \) m resulting in \( \sigma_r = 0.005 \) m at \( r = 1 \) m for the raw depth.

We map three trajectories from different InteriorNet scenes not present in the training set\(^1\). We use 400 images for mapping per sequence, picking large rooms with wide ranges of motion to highlight the advantages of applying depth completion when data is missing. Our experiments compare: (i) the raw depth images with a quadratic depth uncertainty as given in Equation (2) (R); (ii) the completed images with the predicted depth uncertainty from our network (C); and (iii) a combination of the two (R+C), where the raw depth and the quadratic sensor model is used in known areas, and the network completes unknown pixels with the corresponding predicted uncertainty. Method ‘R+C’ corresponds to our proposed approach, which is depicted in the top images in Figure 1 and the system diagram in Figure 2. The motivation behind this strategy is to preserve detailed raw data in valid regions and complete the rest with the network. For ‘C’ and ‘R+C’, for pixels where the network-generated depth is used, we compare the network depth uncertainty with the quadratic uncertainty model (Equation (2)) when using the network completed depth. If the network depth uncertainty is more than 2 times greater than the quadratic uncertainty model \( \text{only} \) free space is integrated for this pixel, otherwise normal integration is performed. This prevents us from creating incorrect surfaces for very uncertain completions while still obtaining usable probabilistic free space estimates.

Our evaluation metrics are the volumes of correct and incorrect mapped free space in the environment with respect to the map generated with ground truth depth at a given image frame. Voxels with occupancy probability < 0.04\% are considered as being free in the reconstructions; for ground truth, we use a less conservative threshold of < 3\%. To measure accuracy in the final reconstructions, we create meshes using marching cubes, and compute the average distance from the ground truth mesh to an output mesh.

The evaluation results are summarised in Table II. Using the network completed depth and uncertainty in our mapping pipeline (‘C’ and ‘R+C’) leads to remarkably more discovered free space volume since, thanks to the predictions, we can capture the free space associated with all pixels, instead of only those with valid raw depth measurements. As expected, the incorrect free space using completion is also slightly greater due to imperfect network predictions; however, it is much smaller compared to the gain in correct free space. Our proposed combined approach (‘R+C’) mitigates this issue by preserving the real raw depth where it is available. Finally, though the reconstruction accuracy with completion is slightly worse when compared to using the raw depth alone (‘R’), it is not drastically degraded with respect to the size of the rooms. We emphasise here that our aim is not to achieve higher-quality fine-scale reconstructions on object level, but rather map the free space in the environment suitable for motion planning while preserving its structure.

Figure 7 depicts the evolution of the mapped free space volumes during the three sequences using each mapping strategy. As a qualitative result, Figure 8 illustrates occupancy map cross-sections obtained at image frame 100 of Seq. 1. The plots in Figure 7 verify that the completion methods (yellow, purple) consistently map significantly more

\(^1\) Seq. 1: ‘3FO4K9H4NDAO (7)’; Seq. 2: ‘3FO4JVRHCIC4T (7)’; Seq. 3: ‘3FO4JXILITSO (7)’. Trajectory numbers are given in the parentheses.
free space compared to using only the raw depth (orange), even with a small number of images, while free space error is kept relatively low and grows slowly. Figure 8 depicts visually the greater proportion of free space (blue areas) achieved using depth completion, especially on the side of the room away from the depth camera (bottom). This portrays the benefit of using our completion framework for mapping in large environments where raw depth coverage is limited.

As a final comparison, the bottom images of Figure 1 show the occupancy map cross-sections from Figure 8 overlaid on the output meshes obtained using the ‘R’ and ‘R+C’ approaches. We confirm that the free space in the room is much more complete using our depth completion pipeline, with the reconstruction quality remaining visually similar.

D. Evaluation on Real-World Data

We demonstrate our pipeline for occupancy mapping with depth completion using the fr2/pioneer_slam2 sequence of the TUM RGB-D dataset [27], which contains trajectory ground truth and RGB-D images captured with a Microsoft Kinect on a Pioneer robot. Note that TUM RGB-D does not contain ground truth depth for evaluating accuracy as in Section IV-C. Instead, the aim is to validate qualitatively the benefit of using our trained depth completion system to map free space using real-world images.

For mapping, we use supereight ‘MultiresOFusion’ with a voxel resolution of 0.0146 m in a 15 m × 15 m × 15 m volume. The constants \(k_\tau\), \(\tau_{\text{min}}\), and \(\tau_{\text{max}}\) are 0.05, 0.06 m, and 0.16 m, respectively, and the quadratic uncertainty model in Equation (2) uses \(k_\sigma = 0.0025\) m, \(\sigma_{\text{min}} = 0.0098\) m, and \(\sigma_{\text{max}} = 0.0294\) m. These parameter values correspond to those used by Funk et al. [8] to evaluate ‘MultiresOFusion’ on real-world data. For depth completion, we use our 2-decoder, 64-channel network from Section IV-B trained on Matterport3D and with images downsampled to 320 px × 240 px. Note that applying the weights finetuned on InteriorNet in Section IV-C produced similar results.

We compare mapping fr2/pioneer_slam2 (1628 images) using: raw depth with the quadratic sensor model in Equation (2) (denoted by ‘R’ in Section IV-C) and our proposed combined approach (‘R+C’), using the network-generated completed depth and depth uncertainty in areas where raw depth is invalid. Examples of depth completion can be seen in Figure 9. As described in Section IV-C, for our proposed method, we integrate only the free space for completed pixels where the network depth uncertainty is more than 2 times greater than the quadratic uncertainty of the raw sensor model. As before, we measure free voxels based on an occupancy probability of < 0.04%.

Figure 10 shows the free space volume mapped during the sequence using the two approaches. Similar to our InteriorNet experiments, using depth completion for mapping (purple in the plot in Figure 10) yields a faster discovery of free space compared against the raw data alone (orange). The map cross-sections in Figure 10 depict visually more free space (blue) at the end of the sequence using completion. These results validate our pipeline in real-world settings.
V. CONCLUSIONS AND FUTURE WORK

This paper introduced a framework for volumetric mapping using depth completion with uncertainty. A core component of our pipeline is a new network architecture for jointly predicting missing depth and depth uncertainty based on images from commodity-grade RGB-D cameras in cluttered indoor environments. The probabilistic depth is used as an input for mapping to complement the raw depth images, allowing us to obtain more complete free space maps.

We performed an ablation study to validate our framework for depth completion with uncertainty. The complete system for mapping with probabilistic depth was evaluated with various trajectories using synthetic RGB-D data. We showed that our proposed approach using both raw and completed depth achieves the fastest discovery of correct free space without compromising on map accuracy. This property is crucial for robotic planning tasks, by ensuring safe and reliable navigation in obstacle-free regions. Further experiments validate the performance of our approach using real-world images.

One limitation is that our network completions are over-smoothed on depth discontinuities due to the high uncertainties present in these regions. Though our combined approach mitigates this issue by using raw depth data, in more complex environments, one could exploit the edge predictions available from training to preserve sharp boundaries. Another idea is to use recurrent networks to ensure consistency in depth prediction between consecutive images. Finally, we will extend our framework to active mapping problems.

ACKNOWLEDGEMENT

We would like to thank Xiao Gu for his advice on simulating realistic depth cameras in our experiments and Dr. Željko Popović for his help with the datasets.

REFERENCES

[1] Y. Zhang and T. Funkhouser, “Deep Depth Completion of a Single RGB-D Image,” in IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2018, pp. 175–185.

[2] Y. K. Huang, T. H. Wu, Y. C. Liu, and W. H. Hsu, “Indoor Depth Completion with Boundary Consistency and Self-Attention,” in International Conference on Computer Vision. IEEE, 2019, pp. 1070–1078.

[3] A. Kendall and Y. Gal, “What uncertainties do we need in Bayesian deep learning for computer vision?” in Advances in Neural Information Processing Systems. Curran Associates, 2017, pp. 5575–5585.

[4] L. Teixeira, M. R. Oswald, M. Pollefeys, and M. Chli, “Aerial Single-View Depth Completion With Image-Guided Uncertainty Estimation,” IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 1055–1062, 2020.

[5] A. Eldesokey, M. Felsberg, K. Holmquist, and M. Persson, “Uncertainty-Aware CNNs for Depth Completion: Uncertainty from Beginning to End,” in IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2020.

[6] A. Eldesokey, M. Felsberg, and F. S. Khan, “Confidence Propagation through CNNs for Guided Sparse Depth Regression,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 42, no. 10, pp. 2423–2436, 2020.

[7] E. Vespa, N. Nikolov, M. Grimm, L. Nardi, P. H. J. Kelly, and S. Leutenegger, “Efficient Octree-Based Volumetric SLAM Supporting Signed-Distance and Occupancy Mapping,” IEEE Robotics and Automation Letters, vol. 3, no. 2, pp. 1144–1151, 2018.

[8] N. Funk, J. Tarrio, S. Papantheodorou, M. Popović, P. F. Alcantarilla, and S. Leutenegger, “Multi-resolution 3D mapping with explicit free space representation for fast and accurate mobile robot motion planning,” arXiv, 2020.

[9] D. Feixl, C. Reinbacher, R. Ranftl, M. Ruetter, and H. Bischof, “Image-Guided Depth Upsampling Using Anisotropic Total Generalized Variation,” in IEEE International Conference on Computer Vision, 2013, pp. 993–1000.

[10] H. Xue, S. Zhang, and D. Cai, “Depth Image Inpainting: Improving Low Rank Matrix Completion With Low Gradient Regularization,” IEEE Transactions on Image Processing, vol. 26, no. 9, pp. 4311–4320, 2017.

[11] J. Uhrig, N. Schneider, L. Schneider, U. Franke, T. Brox, and A. Geiger, “Sparsity Invariant CNNs,” in International Conference on 3D Vision, 2017, pp. 11–20.

[12] F. Ma and S. Karaman, “Sparse-to-Dense: Depth Prediction from Sparse Depth Samples and a Single Image,” 2018.

[13] M. Fehr, T. Taubner, Y. Liu, R. Siegwart, and C. Cadena, “Predicting Unobserved Space for Planning via Depth Map Augmentation,” in International Conference on Advanced Robotics, 2019, pp. 30–36.

[14] J. Qiu, Z. Cui, Y. Zhang, X. Zhang, S. Liu, B. Zeng, and M. Pollefeys, “DeepLiDAR: Deep Surface Normal Guided Depth Prediction for Outdoor Scene From Sparse LiDAR Data and Single Color Image,” in IEEE Conference on Computer Vision and Pattern Recognition, 2019.

[15] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser, “Semantic Scene Completion from a Single Depth Image,” IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[16] A. Dai, D. Ritchie, M. Bokeloh, S. Reed, J. Sturm, and M. Nießner, “Scancomplete: Large-scale scene completion and semantic segmentation for 3d scans,” in IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[17] J. Zeng, Y. Tong, Y. Huang, Q. Yan, W. Sun, J. Chen, and Y. Wang, “Deep Surface Normal Estimation with Hierarchical RGB-D Fusion,” in IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2019.

[18] Y. Zhang, S. Song, E. Yumer, M. Savva, J.-Y. Lee, H. Jin, and T. Funkhouser, “Physically-based rendering for indoor scene understanding using convolutional neural networks,” IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[19] Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600–612, 2004.

[20] C. Loop, Q. Cai, S. Orts-Escolano, and P. A. Chou, “A Closed-Form Bayesian Fusion Equation Using Occupancy Probabilities,” in International Conference on 3D Vision, 2016, pp. 380–388.

[21] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niessner, M. Savva, S. Song, A. Zeng, and Y. Zhang, “Matterport3d: Learning from rgb-d data in indoor environments,” International Conference on 3D Vision, 2017.

[22] E. Ilg, O. Cicek, S. Galesso, A. Klein, O. Makansi, F. Hutter, and T. Brox, “Uncertainty estimates and multi-hypotheses networks for optical flow,” in European Conference on Computer Vision, 2018, pp. 652–667.

[23] N. Li, S. Sacedi, J. McCormac, R. Clark, D. Tzoumanikas, Q. Ye, Y. Huang, R. Tang, and S. Leutenegger, “InteriorNet: Mega-scale Multi-sensor Photo-realistic Indoor Scenes Dataset,” in British Machine Vision Conference, 2018.

[24] C. V. Nguyen, S. Izadi, and D. Lovell, “Modeling Kinect Sensor Noise for Improved 3D Reconstruction and Tracking,” in International Conference on 3D Imaging, Modeling, Processing, Visualization Transmission, 2012, pp. 524–530.

[25] A. Atapour-Abarghouei, S. Akcay, G. Payen de La Garanderie, and J. Sturm, “Scancomplete: Large-scale scene completion and semantic segmentation for 3d scans,” in IEEE Conference on Computer Vision and Pattern Recognition, 2017.