

Efficient and Robust Semantic Mapping for Indoor Environments

Daniel Seichter, Patrick Langer, Tim Wengefeld, Benjamin Lewandowski, Dominik Höchemer, and Horst-Michael Gross

Abstract—A key proficiency an autonomous mobile robot must have to perform high-level tasks is a strong understanding of its environment. This involves information about what types of objects are present, where they are, what their spatial extend is, and how they can be reached, i.e., information about free space is also crucial. Semantic maps are a powerful instrument providing such information. However, applying semantic segmentation and building 3D maps with high spatial resolution is challenging given limited resources on mobile robots. In this paper, we incorporate semantic information into efficient occupancy normal distribution transform (NDT) maps to enable real-time semantic mapping on mobile robots. On the publicly available dataset Hypersim, we show that, due to their sub-voxel accuracy, semantic NDT maps are superior to other approaches. We compare them to the recent state-of-the-art approach based on voxels and semantic Bayesian spatial kernel inference (S-BKI) and to an optimized version of it derived in this paper. The proposed semantic NDT maps can represent semantics to the same level of detail, while mapping is 2.7 to 17.5 times faster. For the same grid resolution, they perform significantly better, while mapping is up to more than 5 times faster. Finally, we prove the real-world applicability of semantic NDT maps with qualitative results in a domestic application.

I. INTRODUCTION

As the application scenarios of mobile robots are getting more complex and challenging, semantic scene understanding becomes increasingly crucial. A robot always needs to understand its environment including the significance of a scene and all objects within. One way to accomplish this scene understanding is to build a semantic 3D representation of the environment through mapping [1–4] and pass it to subsequent analysis modules.

Building on previous own work [5,6], in our ongoing research project MORPHIA, our robots are acting autonomously in domestic environments and allow relatives as well as caregivers to stay in contact with people in need of care in order to let them participate in social life. In this scenario, scene understanding gained from a semantic 3D representation is supposed to improve navigation and to enable semantic instructions such as “wait for me next to the chair”. Moreover, it is to allow our robots to be aware of the current scene, i.e., by not blocking doors or the line of sight to objects such as a TV. To match the character of these dynamic environments, we build upon a strong long-term localization determined via RTAB-Map [7] and build short-term semantic 3D representations of the current environment

Authors are with Neuroinformatics and Cognitive Robotics Lab, Technische Universität Ilmenau, 98693 Ilmenau, Germany. Contact: daniel.seichter@tu-ilmenau.de, ORCID: 0000-0002-3828-2926

This work has received funding from the German Federal Ministry of Education and Research (BMBF) to the project MORPHIA (16SV8426).

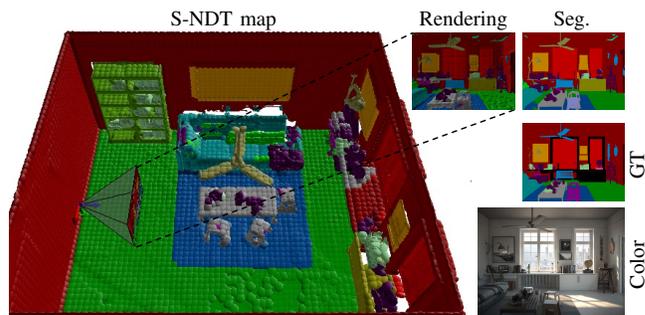


Fig. 1. Semantic occupancy NDT (S-NDT) map built with predicted segmentation of ESANet [8] and cell size of 10cm for scene *ai_048_010* of the Hypersim [9] test set. Right: for the given pose counterclockwise from bottom: color image, ground-truth annotation (GT), semantic segmentation output of ESANet (Seg.), and 2D rendering visualizing the map from the robot’s pose (Rendering). Best viewed in color at 300%. Black indicates the void class, see Fig. 6 for remaining colors.

periodically. This way, we are able to react to small but important changes in the environment, such as rearranged chairs, clothes, or other dynamic objects.

In the field of mobile robotics, voxel-based 3D representations are most common due to their efficient processing [10–12]. Approaches for creating semantic maps often extend these classical mapping approaches by additionally incorporating semantic information [1–4]. However, none of the aforementioned approaches is able to meet our requirements for efficient and robust semantic mapping.

Therefore, in this paper, we aim at developing an approach that can be used in our scenario. We first propose to modify the update step of the recently published semantic Bayesian kernel inference mapping approach (S-BKI) [4] in order to improve its robustness when it is combined with noisy outputs of a semantic segmentation. Unfortunately, this optimized version still does not allow precise semantic mapping at a frame rate of at least 3Hz on our mobile robot. Therefore, we also propose to integrate semantic information into the fast and robust occupancy normal distribution transform (NDT) mapping of [13, 14], enabling precise semantic mapping at sub-voxel level as shown in Fig. 1.

Evaluating semantic mapping approaches for indoor environments is challenging as the number of suitable datasets is limited. In order to compare both proposed approaches to the state of the art, we conduct experiments on the synthetic large-scale dataset Hypersim [9]. We first evaluate their capability to represent semantics by using the provided ground-truth annotation for mapping, i.e., perfect data is assumed. Subsequently, we combine the mapping with our efficient RGB-D semantic segmentation approach [8] in order

to allow a comparison in a more realistic setting when semantic information is noisy. Finally, we show qualitative results of our complete pipeline for efficient and robust semantic mapping in one of our domestic environments and, thus, give an impression of how it performs on real data. In summary, the main contributions of this paper are:

- a modified update step for the semantic Bayesian kernel inference (S-BKI) mapping [4] improving its robustness when it is combined with noisy outputs of a segmentation
- the integration of semantic information into the occupancy normal distribution transform (NDT) mapping [13, 14] realizing semantic NDT maps
- a comparison of the aforementioned approaches on the synthetic large-scale dataset Hypersim [9].

II. RELATED WORK

Representations for 3D environments are diverse, ranging from raw point clouds over meshes [15] and voxels [10] to implicit descriptor-based [16] representations. However, in the field of mobile robotics, representations are most commonly based on voxels due to their efficient processing by subsequent algorithms and controllable need for memory.

A. Voxel-based Data Representations

Voxel-based approaches divide the space into a regular 3D grid of cells with a specific size. These cells can be organized in an octree. OctoMap [10], for example, builds a 3D map by estimating the status of a voxel either as occupied, free, or unknown given range measurements of a sensor. However, when point density is low, e.g., due to large distances, naïve map updates with raw sensor readings may lead to holes in the final map. This issue is targeted by Gaussian processes in GPOctoMap [11] or kernel-based approaches such as Bayesian generalized kernel OctoMap (BGKOctoMap) [12]. Nevertheless, the map precision of these approaches strongly depends on the resolution of the 3D grid. As the cell size also affects the required memory and computation time, a trade-off must be found. Therefore, other approaches do not only estimate an occupancy value per voxel but also attempt to model the surface within the cells. This allows using coarser grid resolutions and, thus, retains real-time requirements while still modelling accurate representations of the environment. Surfel maps [17], for example, use so called surface elements that are characterized by the eigenvectors of the data distributed inside a voxel. However, this representation makes use of a flat surface assumption that becomes insufficient when modelling objects smaller than the grid resolution. A similar representation is the normal distribution transform (NDT) [13, 18, 19] that maintains the whole covariance matrix of the distribution, enabling to model the spatial extend inside a voxel.

B. Semantic Voxel-based Data Representations

When semantic information has to be integrated into grid-based environment models, approaches typically extend non-semantic approaches. Semantic Octree [1] extends OctoMap by integrating semantic information using a class histogram

per voxel. Semantic labels are retrieved from RGB images segmented using a conditional random field (CRF). Finally, the map is further refined using a higher-order CRF. In [2], this approach is improved by incorporating a convolutional neural network for semantic segmentation. [3] extends the idea of using Gaussian processes for inferring not only a binary label (free vs. occupied) but a label in a multi-class problem (free + semantic classes). However, they do not make use of an octree-based data structure like GPOctoMap, which makes this approach less adaptable for larger environments. Semantic Bayesian kernel inference mapping (S-BKI) [4] as generalization of BGKOctoMap models a multi-class problem similar to [3] but using faster kernel-based inference. However, the free space class, which is sampled from range measurements, and any other semantic information from the segmentation are handled in the same way. This can lead to over-representing the free space class and, thus, to inconsistencies in the map, especially when combined with noisy outputs of a semantic segmentation.

Another related field of research is 3D scene reconstruction. Approaches such as [20–23] reconstruct entire semantic scenes using volumetric mapping based on voxel hashing [24] and truncated signed distance fields (TSDF) [25]. However, for efficiency reasons, TSDFs explicitly model free or occupied space only in a truncated region around the surface of objects. Information about unseen areas are missing. Moreover, these approaches rely on computationally intensive CRFs for map refinement that require to be heavily skipped in application to enable at least near real-time capability even on non-mobile CPUs and GPUs. Thus, these approaches are not suitable for mobile robotic applications with dynamic environments and limited resources.

In conclusion, integrating semantic information into 3D grid-based data structures has been targeted by several approaches. However, to the best of our knowledge, none of these approaches make use of the advanced NDT data representation that is able to model sub-voxel precise semantic information accurately. For the sake of completeness, in [26], semantics has already been used to improve the scan registration process by building class-specific NDT maps but no consistent semantic global map is generated.

III. EFFICIENT AND ROBUST SEMANTIC MAPPING

Given a precise localization in the environment, our semantic mapping pipeline is a two-step approach as shown in Fig. 2. We first apply a semantic segmentation on the current set of images (color and depth). Afterwards, the obtained per-pixel class information are passed together with the depth image and the current pose to the mapping stage. Unlike other approaches [2, 20–23], we do not apply any post-processing to the resulting maps, such as CRFs, as this would notably increase runtime. For semantic segmentation, we rely on our recently proposed RGB-D approach for efficient indoor scene analysis, called ESANet [8]. Semantic mapping can be performed using either semantic Bayesian kernel inference mapping (S-BKI), the proposed optimized version of this approach (OS-BKI), or the proposed semantic

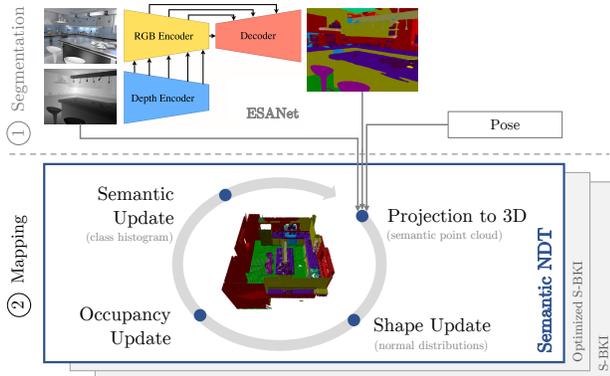


Fig. 2. Our two-step approach for efficient and robust semantic mapping.

occupancy normal distribution transform mapping (S-NDT). Next, we describe each part of the pipeline in detail.

A. Semantic Segmentation

Unlike most other approaches [1–4, 22, 23], we follow [20, 21] and use an RGB-D approach for semantic segmentation. Especially for indoor environments, cluttered scenes may impede semantic segmentation. Incorporating depth images can alleviate this effect by providing complementary geometric information. In [8], we have shown that RGB-D segmentation is superior to processing RGB images solely and that it can still be performed in real time if the network architecture is carefully designed. More precisely, we use ESANet-R34-NBt1D (enhanced ResNet34-based encoder utilizing the Non-Bottleneck-1D block (NBt1D) [27]) in our pipeline. It enables semantic segmentation with up to 29.7Hz on a NVIDIA Jetson AGX Xavier making it well suited for efficient and robust semantic mapping on mobile robots. For application in real-world scenarios, pretrained weights for both datasets NYUv2 [28] and SUNRGB-D [29] are publicly available, reaching a mean intersection over union of 51.58 and 48.17 on the test sets, respectively.

B. Mapping: Optimized S-BKI (OS-BKI)

S-BKI [4] extends BGKOctoMap [12] by additionally incorporating semantic labels. Both use Bayesian generalized kernel inference¹ for determining the label of a voxel in order to improve mapping in regions with low point density. The label is inferred by weighting all observations up to a specified maximum distance to the current voxel (kernel length l) using a kernel function. Since this does not consider information about free space inherently, additional points labeled as free are sampled along the sensor’s rays and are incorporated into the kernel inference.

However, S-BKI distinguishes $1+n$ classes (free + n semantic classes) in the same way as BGKOctoMap only distinguishes free and occupied. As shown in Fig. 3 (left), this leads to major problems in terms of holes in the map. Especially when combining with noisy outputs of a semantic segmentation, the frequency for the free class tends to be the highest in areas where multiple semantic labels occur. This is

due to the fact that each semantic class is counted separately, although all semantic observations indicate occupancy. A larger kernel length or an even more noisy segmentation further increase this effect.

To tackle this issue, we propose to split the update in two consecutive steps. In the first step, the kernel is applied to a two-class problem ignoring the semantic at all and distinguishing only free and occupied as done in BGKOctoMap. The second step is used to infer semantic labels but only considering semantic observations. Although applying the kernel two times sounds like more overhead, the computation time is only slightly affected as shown in our experiments. This is due to the fact that the complexity of the first step and the number of observations in the second step are reduced. As shown in Fig. 3 (right) and our experimental results, OS-BKI greatly improves mapping quality.

C. Mapping: Semantic Occupancy NDT (S-NDT)

The proposed OS-BKI approach improves mapping with noisy outputs of a semantic segmentation. Nevertheless, mapping with sufficient resolution is not possible in real time on our mobile robot without relying on multiple CPU cores. However, saving computational resources is important since our mobile robot typically has to perform other tasks in parallel, like navigation or person perception and tracking.

Therefore, we propose semantic occupancy NDT (S-NDT) mapping that allows efficient semantic mapping and, moreover, representing the environment on a sub-voxel level. Compared to traditional voxel maps [10], 3D occupancy NDT maps not only store an occupancy value per voxel but also a three-dimensional mean vector and a 3×3 covariance matrix, describing the underlying surface in terms of a normal distribution. These normal distributions can be updated efficiently in an incremental fashion as shown in [13]. We integrate semantic information by additionally introducing a class histogram in each voxel. This histogram is updated using the top- k labels of the segmentation output. However, in accordance with other approaches [3, 4], the top-1 labels already seem to be sufficient.

In [13, 14, 30], it is shown that occupancy NDT maps have a higher effective resolution, i.e., they are able to represent a surface more accurately at the same grid resolution due to their sub-voxel surface description. In other words, occupancy NDT maps can achieve a similar accuracy with a reduced grid resolution. This does not only save

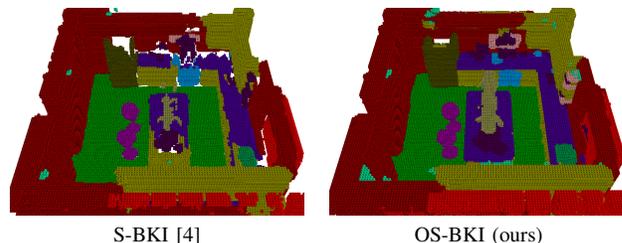


Fig. 3. Semantic maps built with S-BKI (left) and OS-BKI (right) for scene *ai_001_010* of the Hypersim test set. Mapping was done with predicted segmentation of ESANet [8], cell size of 5cm, and kernel length of 20cm. Best viewed in color at 200%, see Fig. 6 for label colors.

¹For further precise mathematical descriptions, we refer to [4, 12].

computational resources during mapping but also enlarges the spatial volume that is covered per voxel. As shown with the countertop on the right in Fig. 4, this also compensates for input data with low point density, since the resulting map is less incomplete. In large parts, the semantic occupancy NDT map with cell size 15cm (Fig. 4, bottom) is similarly accurate as its OS-BKI counterpart with 5cm (Fig. 3, right).

IV. EXPERIMENTS

Evaluating semantic mapping approaches is challenging. Especially for indoor environments, the number of datasets is limited. NYUv2 [28] and SUNRGB-D [29] are used for benchmarking semantic segmentation approaches. However, they only provide semantics for single frames of a scene and, thus, are not suitable for evaluating semantic mapping. Therefore, we conduct mapping experiments on the recently published synthetic large-scale Hypersim dataset [9].

A. Dataset & Network Training

The Hypersim dataset [9] contains 77,400 images of 461 photo-realistic synthetic indoor scenes (see Fig. 1 bottom right). In total, 774 camera trajectories have been rendered using random walks for these scenes. Each trajectory consists of 100 camera poses, ensuring reasonable visual coverage. For each camera pose, among others, camera extrinsics, color and depth image as well as per-pixel semantic labels following the NYUv2 classes are available. The dataset comes with splits for training, validation, and test. We used the training set with a random subsampling of 20% per epoch for training our ESANet-R34-NBt1D [8]. The validation set was used to determine the best epoch. All mapping experiments are conducted on the test set. Unfortunately, the dataset contains some invalid scenes and trajectories (void/single label only, missing textures, invalid depth) that we filtered from training and validation set manually. However, although containing invalid scenes as well, we did not touch the test set.

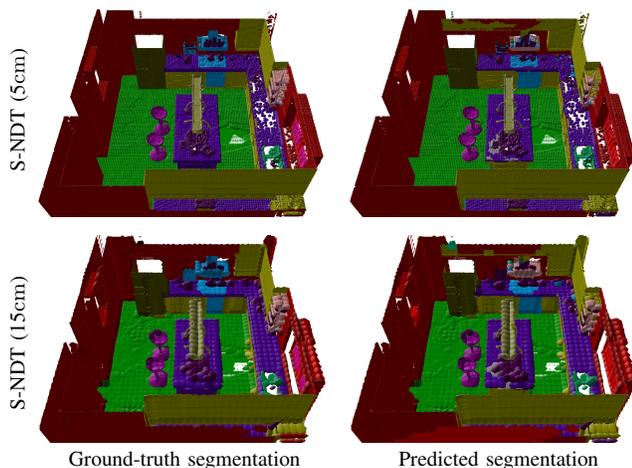


Fig. 4. Semantic maps built with S-NDT with cell sizes of 5cm (top) and 15cm (bottom) for scene *ai_001_010* of the Hypersim test set. Columns show results when mapping with ground-truth segmentation (left, perfect input for mapping) or predicted segmentation of ESANet [8] (right, noisy input for mapping). Best viewed in color at 200%, see Fig. 6 for label colors.

B. Implementation Details & Evaluation Protocol

Since each trajectory ensures reasonable visual coverage of the respective scene, we map each trajectory independently in our experiments. We use either the ground-truth annotation or the output of our ESANet-R34-NBt1D as segmentation labels. Similar to [2,4], we limit the maximum mapping distance in our experiments to a reasonable value of 20m. For all approaches, we evaluate cell sizes of 5, 10, 15, and 20cm. For S-BKI and OS-BKI, we vary the kernel length (denoted in brackets) in steps of 10cm between 10cm and 40cm.

For evaluating the resulting semantic maps, we follow [2, 4]. We project the semantic map back to the camera plane for all camera poses of a trajectory and compute the mean intersection over union (mIoU) with the ground-truth annotation ignoring pixels labeled as void in the ground truth and classes that do not appear in the test set at all. However, the back-projections can include another class of invalid pixels, i.e., pixels indicating free space and no semantic class (see Fig. 5). As the IoUs can only account for these results as false negatives and not as false positives for this invalid class (there is no ground truth), the mIoU might be misleading. Therefore, we further report the overall ratio of invalid pixels (invR) and the mean pixel accuracy (mPAcc) considering only true positives and false negatives per class.

In order to enable the community to have a similar setup, our code for preparing the Hypersim dataset, the weights for our ESANet-R34-NBt1D reaching a mIoU of 41.17 on the Hypersim test set, and the code for evaluation are publicly available at: <https://github.com/TUI-NICR/semantic-mapping>.

C. Mapping with Perfect Ground-truth Segmentation

We first evaluate the mapping approaches in a setting assuming perfect semantic segmentation, i.e., using the provided ground-truth annotation. The goal is to determine how well semantics can be represented by the resulting maps. The results are listed in Tab. I (top). Tab. II further reports the mean map update rates for each configuration.

With perfect semantic segmentation, S-BKI and the proposed OS-BKI perform similarly. However, the invalid ratios (invR) and the mean pixel accuracy (mPAcc) in Tab. I show that the reported issue of S-BKI increases as the kernel length increases. For OS-BKI, the invR remains stable enabling better results for these settings without notably lowering the update rate. However, both approaches cannot compete with the proposed S-NDT that performs better and is much faster across all cell sizes. Due to the sub-voxel precise semantics, S-NDT allows the same or even higher semantic map quality while using the next coarser cell size. Fig. 6 (left) visualizes the IoUs for S-BKI and OS-BKI with a cell size of 10cm and a reasonable kernel length of 30cm and compares them to S-NDT counterparts with same update rate or same map quality. The IoUs shows that this statement holds true for almost all classes.

Finally, the results of this set of experiments show that, even with perfect segmentation as input to the mapping

TABLE I: Results on the Hypersim test set when mapping with ground-truth (GT) segmentation (top) and predicted segmentation of ESANet [8] (bottom). For each configuration, four cell sizes are considered, and the mean intersection over union (mIoU), the ratio of invalid pixels (invR), and the mean pixel accuracy (mPAcc) over all trajectories are listed as percentages. For S-BKI and OS-BKI, the kernel length is denoted in brackets. For each cell size, the best result for mIoU and mPAcc is printed in bold.

	Cell size 5cm			Cell size 10cm			Cell size 15cm			Cell size 20cm			
	mIoU \uparrow	invR \downarrow	mPAcc \uparrow	mIoU \uparrow	invR \downarrow	mPAcc \uparrow	mIoU \uparrow	invR \downarrow	mPAcc \uparrow	mIoU \uparrow	invR \downarrow	mPAcc \uparrow	
GT segmentation	S-BKI (10cm)	69.18	0.00	84.35	65.10	0.37	80.60	59.56	2.49	74.19	49.16	16.63	60.71
	S-BKI (20cm)	68.66	0.36	83.12	63.53	0.36	79.80	60.42	0.38	77.07	57.10	0.13	72.47
	S-BKI (30cm)	68.10	0.43	80.05	63.53	0.44	77.01	59.75	0.45	74.22	55.99	0.52	70.65
	S-BKI (40cm)	64.15	0.57	74.26	59.27	0.57	69.53	56.91	0.61	67.95	53.27	0.71	64.59
	OS-BKI (10cm)	67.91	0.33	83.39	64.28	0.33	79.94	61.71	0.40	76.81	51.88	6.75	64.44
	OS-BKI (20cm)	65.82	0.33	81.85	60.95	0.33	77.90	59.13	0.33	76.58	56.19	0.34	72.18
	OS-BKI (30cm)	66.02	0.33	81.52	62.47	0.33	79.23	59.27	0.33	76.89	55.12	0.33	72.27
	OS-BKI (40cm)	65.82	0.33	80.16	63.55	0.33	79.08	60.47	0.33	76.81	56.13	0.33	72.65
S-NDT	78.28	2.93	88.30	72.04	0.83	85.12	66.53	0.73	81.13	61.96	0.78	78.10	
Predicted segmentation	S-BKI (10cm)	37.95	3.23	50.79	35.48	3.53	47.77	32.99	6.66	44.87	26.89	21.88	35.80
	S-BKI (20cm)	35.92	4.16	47.37	34.14	4.56	45.67	33.11	4.70	44.62	31.19	4.94	42.28
	S-BKI (30cm)	33.94	5.18	43.99	32.37	5.92	42.42	31.37	6.10	41.62	29.88	5.98	39.59
	S-BKI (40cm)	31.55	6.08	40.60	28.98	7.60	37.22	28.42	8.09	36.76	27.11	8.26	35.44
	OS-BKI (10cm)	36.78	0.33	52.24	35.30	0.33	50.61	34.00	0.40	49.08	28.58	0.66	41.46
	OS-BKI (20cm)	36.26	0.33	51.47	34.27	0.33	49.52	33.21	0.33	48.86	31.85	0.34	46.80
	OS-BKI (30cm)	36.52	0.33	51.26	35.03	0.33	50.23	33.55	0.33	49.03	31.58	0.33	46.35
	OS-BKI (40cm)	36.53	0.32	50.58	35.70	0.33	50.40	34.13	0.33	49.04	32.51	0.33	46.58
S-NDT	41.12	2.93	56.18	39.62	0.83	54.73	37.68	0.73	52.82	35.56	0.78	50.37	

stage, the discretization of the environment leads to an unavoidable drop in performance compared to the ground-truth annotation. Fig. 6 (left) illustrates that a large part of this drop is mainly caused by small or thin objects (otherprop, lamp, books, shelves, towel, box, sink, paper). Mapping such classes accurately would require even much finer cell sizes.

D. Mapping with Predicted Segmentation

Next, we use the output of our ESANet-R34-NBt1D as input for mapping. This allows comparing the approaches in a more realistic scenario. The ESANet-R34-NBt1D (without mapping) reaches a mIoU of 41.17 on the test set.

As shown in Tab. I (bottom) and further visualized by the back-projections in Fig. 5, unlike our proposed OS-BKI, S-BKI does not allow satisfying mapping in this scenario. In areas where multiple labels occur, free space tends to be over-presented resulting in holes in the map (invalid pixels in

the back-projection). For OS-BKI, the ratio of invalid pixels is smaller, leading to much better results.

However, again, S-NDT performs much better, almost reaching the performance of the ESANet-R34-NBt1D for a cell size of 5cm. The IoUs in Fig. 6 (right) further indicate that the results are close for almost all classes. Moreover, for some classes (cabinet, counter, blinds, bed, door, refrigerator, bathtub, ...) integrating predictions over time in semantic NDT maps seems to improve results.

The results in Tab. I for both experiments on Hypersim show that the semantic quality of maps created with S-NDT is mainly limited by the quality of the preceding semantic segmentation.

TABLE II: Mean map update rate (Hz) over all trajectories of the Hypersim test set for all approaches listed in Tab. I. As the update rate depends on the size of the scene, further, the mean for the slowest (top) and fastest (bottom) scene is reported to the right. The update rate is determined for single-thread processing on our robot featuring an Intel i7 6700HQ mobile CPU. The best result is printed in bold.

Cell size	Mean map update rate \uparrow			
	5cm	10cm	15cm	20cm
S-BKI (10cm)	0.2 3.8	0.9 8.0	2.0 10.3	3.4 11.0
S-BKI (20cm)	0.2 3.6	0.9 7.9	2.0 10.0	3.4 11.1
S-BKI (30cm)	0.2 3.6	0.8 7.9	2.0 10.1	3.4 11.0
S-BKI (40cm)	0.2 3.7	0.8 7.8	1.9 9.3	3.3 11.1
OS-BKI (10cm)	0.2 3.8	0.9 7.6	1.9 9.0	3.5 11.2
OS-BKI (20cm)	0.2 3.8	0.9 8.1	1.9 9.3	3.1 10.6
OS-BKI (30cm)	0.2 3.6	0.9 7.9	1.9 9.5	3.0 9.8
OS-BKI (40cm)	0.2 3.8	0.8 7.2	1.9 9.7	3.0 9.9
S-NDT	1.0 1.2	3.5 6.8	5.2 14.9	5.4 21.9

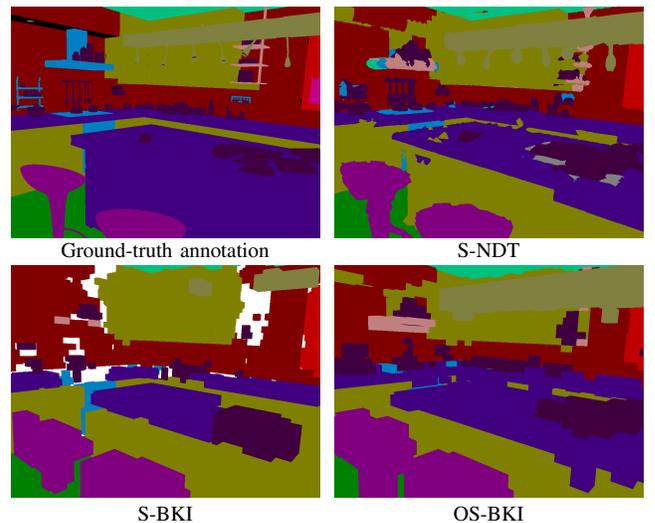


Fig. 5. Ground-truth annotation and back-projections for the same camera pose of scene *ai_001_010/cam_01* of the Hypersim test set when mapping with predicted segmentation and cell size of 10cm. For S-BKI and OS-BKI, a kernel length of 30cm is used. White indicates invalid pixels (free space), black pixels belong to the void class. See Fig. 6 for remaining colors.

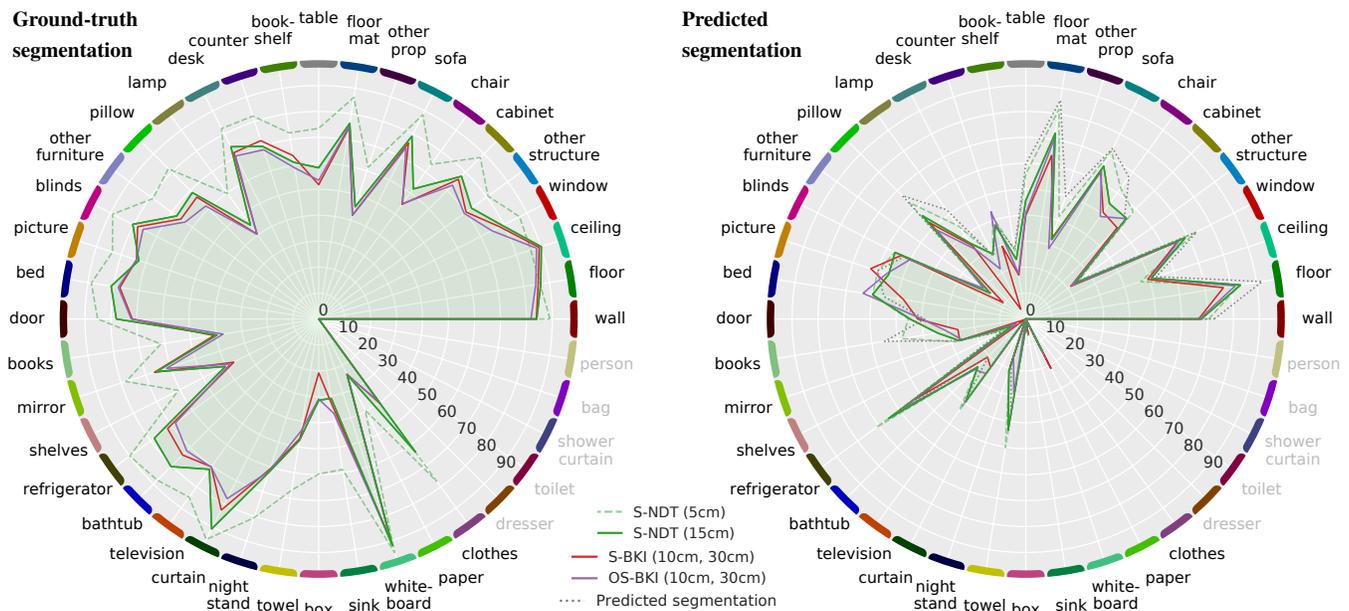


Fig. 6. Per-class intersection over union (IoU) on the Hypersim test set when mapping with the ground-truth segmentation (left) and predicted segmentation of ESANet (right). Classes are ordered descending by frequency counterclockwise. Classes printed in gray do not appear in the test set. Cell size and kernel length (2nd parameter) are denoted in brackets.

V. APPLICATION IN REAL-WORLD SCENARIO

We apply our proposed two-step pipeline for efficient and robust semantic mapping in the context of our MORPHIA project in various apartments of seniors, such as the one shown in Fig. 7. After an initial non-semantic mapping phase using RTAB-Map [7], it is switched to localization mode to provide a reliable long-term localization. For short-term semantic mapping, we choose the ESANet-R34-NBt1D trained on SUNRGB-D [29] and S-NDT with cell size of 10cm. Even though SUNRGB-D does not contain any images of our Azure Kinect sensor setup, the mapping results show a suitable level of quality, modeling the semantics of the environment as well as dynamic objects, such as chairs, doors, or clothes, along with their spatial extend. This enables our robot to be aware of the current scene and to perform high-level semantic control commands or delivery tasks for the senior. For further impressions on the semantic mapping, we refer to the attached video or our repository on GitHub.

VI. CONCLUSION AND FUTURE WORK

In this paper, we have incorporated semantic information into efficient occupancy NDT maps realizing semantic occupancy NDT (S-NDT) maps. Our experiments on the Hypersim dataset [9] revealed that S-NDT mapping outperforms the recent state-of-the-art approach S-BKI and an optimized version of it in terms of both performance and runtime. Our code for preparing the Hypersim dataset, the weights for our applied ESANet-R34-NBt1D, and the code for evaluation are publicly available. Thereby, we hope that we can call the community’s attention to the Hypersim dataset for evaluating semantic mapping approaches.

We also showed that S-NDT mapping combined with our ESANet-R34-NBt1D represents a powerful instrument for

real-world robotic applications, enabling mobile robots to gain a strong understanding of their environment.

In future work, we intend to conduct experiments on further datasets such as ScanNet [31] as well.

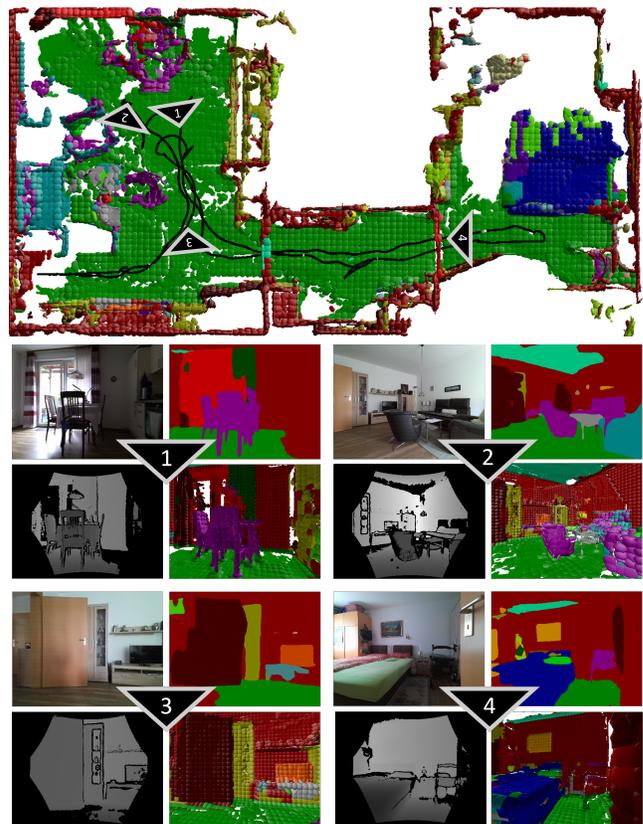


Fig. 7. Top: Bird’s-eye view of the S-NDT map of the apartment of a test participant including the robot’s trajectory (black). Bottom: for selected poses (1-4), clockwise from bottom left: depth and color image, semantic segmentation of ESANet-R34-NBt1D, and 2D rendering visualizing the map from the robot’s pose. Best viewed at 200%, see Fig. 6 for label colors.

REFERENCES

- [1] S. Sengupta and P. Sturgess, "Semantic Octree: Unifying Recognition, Reconstruction and Representation via an Octree Constrained Higher Order MRF," in *Int. Conf. on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 1874–1879.
- [2] S. Yang, Y. Huang, and S. Scherer, "Semantic 3D Occupancy Mapping through Efficient High Order CRFs," in *Int. Conf. on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 590–597.
- [3] M. G. Jadidi, L. Gan, S. A. Parkison, J. Li, and R. Eustice, "Gaussian Processes Semantic Map Representation," *ArXiv preprint arXiv:1707.01532*, 2017.
- [4] L. Gan, R. Zhang, J. W. Grizzle, R. M. Eustice, and M. Ghaffari, "Bayesian Spatial Kernel Smoothing for Scalable Dense Semantic Mapping," *Robotics and Automation Letters (RAL)*, vol. 5, no. 2, pp. 790–797, 2020.
- [5] H.-M. Gross, St. Mueller, Ch. Schroeter, M. Volkhardt, A. Scheidig, K. Debes, K. Richter, and N. Doering, "Robot Companion for Domestic Health Assistance: Implementation, Test and Case Study under Everyday Conditions in Private Apartments," in *Int. Conf. on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 5992–5999.
- [6] H. M. Gross, A. Scheidig, St. Mueller, B. Schuetz, C. Fricke, and S. Meyer, "Living with a Mobile Companion Robot in your Own Apartment - Final Implementation and Results of a 20-Weeks Field Study with 20 Seniors," in *Int. Conf. on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 2253–2259.
- [7] M. Labbé and F. Michaud, "RTAB-Map as an Open-Source Lidar and Visual Simultaneous Localization and Mapping Library for large-scale and long-term Online Operation," *Journal of Field Robotics (JFR)*, vol. 36, no. 2, pp. 416–446, 2019.
- [8] D. Seichter, M. Köhler, B. Lewandowski, T. Wengefeld, and H.-M. Gross, "Efficient RGB-D Semantic Segmentation for Indoor Scene Analysis," in *Int. Conf. on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 13 525–13 531.
- [9] M. Roberts, J. Ramapuram, A. Ranjan, A. Kumar, M. A. Bautista, N. Paczan, R. Webb, and J. M. Susskind, "Hypersim: A Photorealistic Synthetic Dataset for Holistic Indoor Scene Understanding," in *Int. Conf. on Computer Vision (ICCV)*, 2021.
- [10] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: An Efficient Probabilistic 3D Mapping Framework based on Octrees," *Autonomous Robots (AR)*, vol. 34, no. 3, pp. 189–206, 2013.
- [11] J. Wang and B. Englot, "Fast, Accurate Gaussian Process Occupancy Maps via Test-Data Octrees and Nested Bayesian Fusion," in *Int. Conf. on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 1003–1010.
- [12] K. Doherty, J. Wang, and B. Englot, "Bayesian Generalized Kernel Inference for Occupancy Map Prediction," in *Int. Conf. on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 3118–3124.
- [13] E. Einhorn and H.-M. Gross, "Generic 2D/3D SLAM with NDT Maps for Lifelong Application," in *Europ. Conf. on Mobile Robots (ECMR)*, 2013.
- [14] E. Einhorn, "Visuelle Umgebungswahrnehmung und Kartierung zur Navigation mobiler Serviceroboter in realen Einsatzumgebungen," Ph.D. dissertation, Ilmenau University of Technology, 2019.
- [15] Y. Zhou, W. Wang, W. Guan, Y. Wu, H. Lai, T. Lu, and M. Cai, "Visual Robotic Object Grasping Through Combining RGB-D Data and 3D Meshes," in *Int. Conf. on Multimedia Modeling (MMM)*. Springer, 2017, pp. 404–415.
- [16] D. Shin, C. C. Fowlkes, and D. Hoiem, "Pixels, Voxels, and Views: A Study of Shape Representations for Single View 3D Object Shape Prediction," in *Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 3061–3069.
- [17] J. Kläß, J. Stückler, and S. Behnke, "Efficient Mobile Robot Navigation using 3D Surfel Grid Maps," in *ROBOTIK 2012; 7th German Conference on Robotics*. VDE, 2012, pp. 1–4.
- [18] P. Biber and W. Straßer, "The Normal Distributions Transform: A new Approach to Laser Scan Matching," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, vol. 3. IEEE, 2003, pp. 2743–2748.
- [19] M. Magnusson, A. Lilienthal, and T. Duckett, "Scan Registration for Autonomous Mining Vehicles using 3D-NDT," *Journal of Field Robotics (JFR)*, vol. 24, no. 10, pp. 803–827, 2007.
- [20] J. McCormac, A. Handa, A. Davison, and S. Leutenegger, "Semanticfusion: Dense 3D Semantic Mapping with Convolutional Neural Networks," in *Int. Conf. on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 4628–4635.
- [21] J. Jeon, J. Jung, J. Kim, and S. Lee, "Semantic Reconstruction: Reconstruction of Semantically Segmented 3D Meshes via Volumetric Semantic Fusion," in *Computer Graphics Forum*, vol. 37, no. 7. Wiley Online Library, 2018, pp. 25–35.
- [22] Q.-H. Pham, B.-S. Hua, T. Nguyen, and S.-K. Yeung, "Real-time Progressive 3D Semantic Segmentation for Indoor Scenes," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2019, pp. 1089–1098.
- [23] G. Narita, T. Seno, T. Ishikawa, and Y. Kaji, "Panopticfusion: Online volumetric semantic mapping at the level of stuff and things," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 4205–4212.
- [24] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger, "Real-time 3D Reconstruction at Scale using Voxel Hashing," *ACM Transactions on Graphics (ToG)*, vol. 32, no. 6, pp. 1–11, 2013.
- [25] B. Curless and M. Levoy, "A Volumetric Method for Building Complex Models from Range Images," in *Proc. of the ACM on Computer Graphics and Interactive Techniques (PACMCGIT)*, 1996, pp. 303–312.
- [26] A. Zaganidis, M. Magnusson, T. Duckett, and G. Cielniak, "Semantic-assisted 3D Normal Distributions Transform for Scan Registration in Environments with Limited Structure," in *Int. Conf. on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 4064–4069.
- [27] E. Romera, J. M. Alvarez, L. M. Bergasa, and R. Arroyo, "ERFNet: Efficient Residual Factorized ConvNet for Real-Time Semantic Segmentation," *IEEE Transactions on Intelligent Transportation Systems (ITS)*, pp. 263–272, 2018.
- [28] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor Segmentation and Support Inference from RGBD Images," in *Europ. Conf. on Computer Vision (ECCV)*, 2012.
- [29] S. Song, S. P. Lichtenberg, and J. Xiao, "SUN RGB-D: A RGB-D Scene Understanding Benchmark Suite," in *Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 567–576.
- [30] T. Stoyanov, M. Magnusson, H. Almqvist, and A. J. Lilienthal, "On the Accuracy of the 3D Normal Distributions Transform as a Tool for Spatial Representation," in *Int. Conf. on Robotics and Automation (ICRA)*. IEEE, 2011, pp. 4080–4085.
- [31] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "ScanNet: Richly-annotated 3D Reconstructions of Indoor Scenes," in *Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5828–5839.