2018 IEEE International Conference on Robotics and Automation (ICRA)
May 21-25, 2018, Brisbane, Australia

Semantic Labeling of Indoor Environments from 3D RGB Maps

Manuel Brucker[*1] Maximilian Durner[*1] Rareş Ambruş[*2] Zoltán Csaba Márton[1]
Axel Wendt[3] Patric Jensfelt[2] Kai O. Arras[3] Rudolph Triebel[1,4]

***Abstract*— We present an approach to automatically assign semantic labels to rooms reconstructed from 3D RGB maps of apartments. Evidence for the room types is generated using state-of-the-art deep-learning techniques for scene classification and object detection based on automatically generated virtual RGB views, as well as from a geometric analysis of the map's 3D structure. The evidence is merged in a conditional random field, using statistics mined from different datasets of indoor environments. We evaluate our approach qualitatively and quantitatively and compare it to related methods.**

I. Introduction

One of the main challenges in robotic applications for household environments is to provide an accurate and reliable semantic perception system. Concretely, the robot must be able to identify all relevant parts of the environment, including room types (e.g. kitchen, bathroom), furniture (tables, cupboards, etc.) and movable objects such as dishes or clothing. Our aim is to solve this complex perception task generally with as little manual intervention as possible. Thus, we want to avoid that a human supervisor must label all parts of an apartment manually before the robot can be deployed, because this approach does not scale well for general applications. Also, it can hardly adapt to changes, for example if a particular room changes its purpose.

To achieve this, we propose an automatic labeling approach for household environments. The main difficulties here are the frequent ambiguities and the contextual dependencies of object and room type labels. A single room is often used for different purposes (e.g. kitchen and dining room), and objects can be found in different contexts (e.g. cushions, cups, etc.). This means that a single source of information such as the geometric features of a room or the occurrence of certain objects is not sufficient to determine a correct labeling. Therefore, in our approach, we rely on three different sources of information, and we provide a labeling that is more fine-grained than the pure geometric layer of abstraction, i.e. it can assign multiple labels to the same geometrical room. Both concepts are novel compared to state-of-the-art approaches, and we show experimentally that they lead to a much more intuitive and human-like labeling.

An illustration of our general framework is given in Fig. 1. Here, a particular robot task is used as a motivating example, and the core components of the system are shown. These are an object classifier applied to different views of a given room, and a geometric segmentation of the entire appartment. For the latter, we rely on our own recent work [1], and for the former we train a deep CNN architecture with a large set of artificial views obtained by projections of fully textured 3D environment models from real apartments[1]. This, in combination with prior information about general object occurances and corresponding contexts, is then used in a Conditional Random Field (CRF) approach to obtain realistic, fine-grained and reliable scene labels.

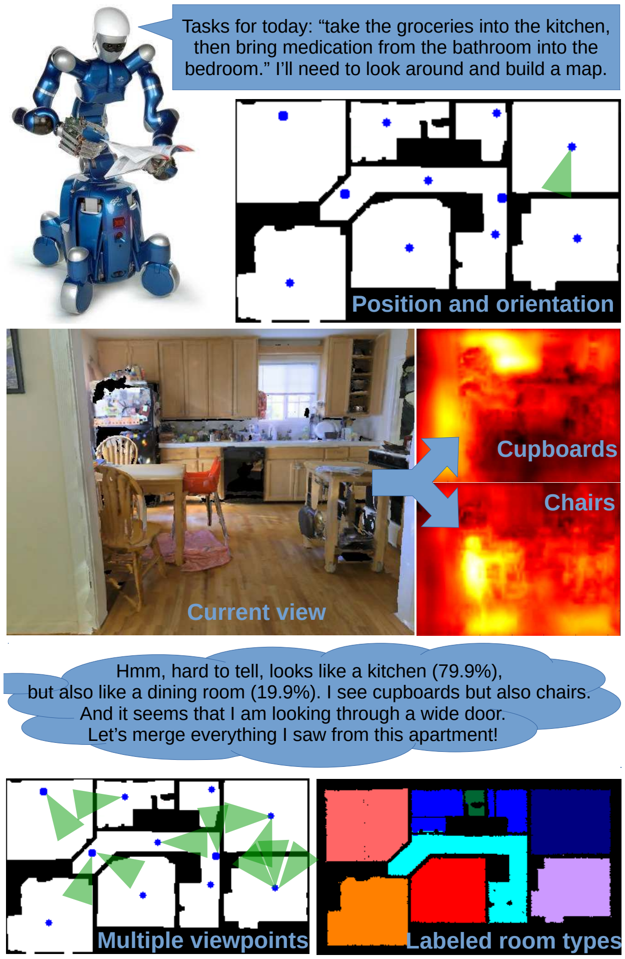
Tasks for today: "take the groceries into the kitchen, then bring medication from the bathroom into the bedroom." I'll need to look around and build a map.
Position and orientation
Cupboards
Chairs
Current view
Hmm, hard to tell, looks like a kitchen (79.9%), but also like a dining room (19.9%). I see cupboards but also chairs. And it seems that I am looking through a wide door. Let's merge everything I saw from this apartment!
Multiple viewpoints
Labeled room types

Fig. 1: Autonomous household service robots need to build and maintain semantic maps under complex conditions.

*The authors contributed equally to this work.
[1]Institute of Robotics and Mechatronics, German Aerospace Center (DLR), 82234 Oberpfaffenhofen, Germany {name.lname}@dlr.de
[2]Centre for Autonomous Systems, KTH Royal Institute of Technology, Stockholm, SE-100 44, Sweden, {raambrus,patric}@kth.se
[3]Robert Bosch, Corporate Research, USA & Germany {axel.wendt, kaioliver.arras}@bosch.com
[4]Dep Comp Science, TU Munich, Germany, triebel@in.tum.de

[1] The data set can be obtained through www.dlr.de/rm/Bosch_Semantic_Interpretation_Challenge.

978-1-5386-3080-8/18/$31.00 ©2018 IEEE

II. Related Work

In this work we assume a pre-segmentation of the environment based on geometric primitives, which we obtain using the method from [1]. As the generation of this segmentation is outside the scope of the current work, we will not be reviewing relevant related work here. For an overview of relevant methods, please refer to [2].

Semantic mapping is an active area of research in the mobile robotics community and can be split depending on the complexity/size of the scene into methods that run on one frame (RGB or RGBD) and methods that label an entire map. Traditional image recognition approaches for labeling single frames typically involve image descriptors for representing the frames, such as e.g. the *Spatial Envelope* of Torralba et al. [3], or the histogram descriptor proposed by Rehg et al. [4]. More recently, *Convolutional Neural Network* (CNN) architectures have surpassed traditional computer vision methods at tasks, such as recognizing the scene [5] or the objects [6] in an image.

More relevant to our work are methods which combine multiple sources of information in a probabilistic fashion to yield a consistent label. Liao et al. [7] propose a CNN architecture which combines object level information to improve scene classification. Uršič et al. [8] fuse a laser-based and a vision-based part detector to reason probabilistically on room categories. However, they do not employ an object detector as we propose here. In contrast, Shi et al. [9] do use vision-based object classification and combine it with cues from laser range data, but they do not exploit the room geometry as we do. Furthermore, our setup is more challenging as we use test and training data from very different environments. Yao et al. [10] use a conditional random field (CRF) for jointly reasoning about object presence, semantic segmentation and scene type in an RGB image. Lin et al. [11] further extend the CRF formulation to also incorporate contextual and geometric relations between scenes and object candidates (parametrised through cuboids) to jointly estimate the scene and objects present in an RGBD image. Although we use similar energy terms (such as distributions over types of objects, and object to scene type weights) which are combined in a CRF, we differ in that we are looking for pixel-wise semantic segmentation of the scene type over the entire map of the environment.

Koppula et al. [12] propose a Markov Random Field (MRF) model for semantically labeling a 3D map at the object level. They consider a number of features, such as visual appearance, local shape and geometry, etc. Hermans et al. [13] also label 3D point cloud maps at the object level, however, they use random forests for the object potentials which are fused in a CRF formulation. Sjoo et al. [14] cast the problem of semantic map segmentation as one of energy maximization, combining a number of features such as convexity, connectivity etc. The problem is solved using simulated annealing. Pronobis et al. [15] propose a probabilistic framework for combining various properties (shape, appearance), the presence of objects and doorways for semantically labeling a 2D map. Sunderhauf et al. [16] also propose an online system for semantic mapping which uses distributions over scene types generated by a CNN. Like us, they also employ a Bayesian step to incorporate predictions when observing the same part of the environment. For an overview of SLAM approaches relevant to the semantic mapping problem, please refer to [17].

The output of our work also is a semantic labeling of a 2D map at the scene level. However, we differ from previous work in that we use CNN distributions over both scene types and object types in the environment, as well as a pre-segmentation of the map based on geometric primitives. We relate object types and scene types via statistics learned from data, and we obtain the final labeling by doing an inference step in a CRF which learned the importance of different cues.

III. Overview

The input to our system is a mesh representation of the environment. In [1] we presented a method for constructing a 2D segmentation of the data into functional building elements, such as rooms or corridors. In the current work we further augment this segmentation with semantic labels on a per-pixel basis, as well as further split the segmentation if enough semantic information is present (e.g. a studio room could be split into a kitchen area and a bedroom area).

Figure 2 shows an overview of the proposed method. Specifically, using [1] we generate the 2D segmentation as well as a set of synthetic viewpoints distributed in the environment. Then we render realistic-looking images to be used in the following classification steps.

The reconstructed viewpoints are meant to simulate possible sensor placements where images can be taken by rotating one or more cameras mounted on a tripod around the upright axis, as was the case in our work in which the data was obtained using a Matterport scanner (see `https://matterport.com/`). It is important to note that we do not have access to the original data used to create the environment mesh, and therefore rendering images inside the mesh is the only way to generate the data for the following steps of our method. To ensure an unbiased and realistically simulated set of images, these are rendered halfway between the floor and the ceilings. At each viewpoint 36 partly overlapping images are rendered with a horizontal rotational step size of 10 degrees to cover 360 degrees. An aspect ratio of 4:3 is used to mimic conventional cameras and therefore the data typically used for training CNNs. A resolution of 640x480 was chosen, since, due to the 3D model quality, higher resolutions could not capture more detail. Gaps in the rendered images, caused by imperfect mesh reconstruction, are filled with black (see examples in Figure 3). A simple filtering on the rendered views discards a rendered image if it is too close to a wall, or the color is too homogeneous.

Next, we learn from training data the statistics correlating object presence to scene types (see Sec. V-A) and we combine these sources of information in a conditional random field (see Sec. V-B). In Sec. VI we show our results as well as compare with other state-of-the-art methods.

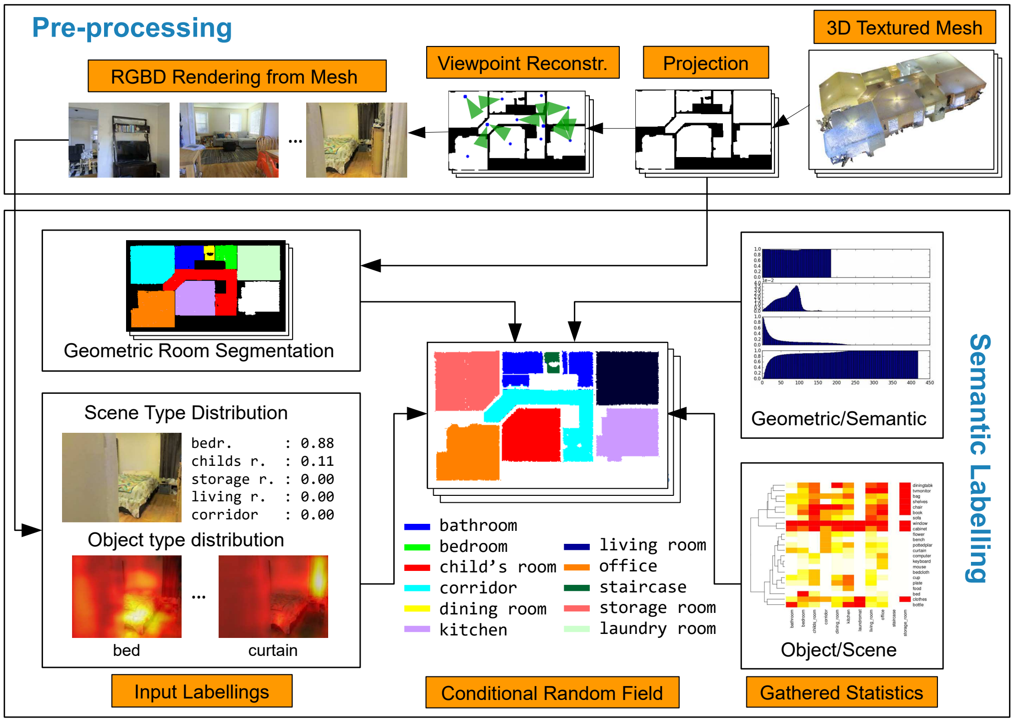

Fig. 2: Overview of the proposed system for semantic room labeling. A geometric room segmentation, CNN results on views rendered from automatically selected viewpoints and co-occurrence frequencies are used as potentials in our CRF.

1) **bathroom**, shower	2) **bedroom**
3) **childs room**, nursery	4) **corridor**
5) **dining room**	6) **kitchen**, galley
7) **laundromat**	8) **living room**
9) **office**, home office	10) **staircase**
11) **storage room**, closet, pantry	12) **"background"**, snowfield, sky, garage indoors

TABLE I: Used subset of Places 365 scene types. Classes in regular font were merged into the color-coded ones.

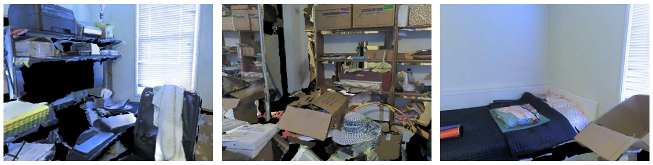

Fig. 3: The appearance can change drastically with the viewpoint in a single room. The Scene Classification scores the images from left to right with: 0.99 office, 0.99 storage room and 0.96 bedroom (cf. Figure 9 row 1 bottom left).

IV. Cues for Semantic Room Labeling

The two most important decisions when using deep learning for image classification are which network architecture to use and which dataset to train on. Since relevant datasets are in general not large enough to support training, preference was given to available pre-trained networks. Of these we focused on the ones most relevant to household scenes and object classification and segmentation, but when these were found to be insufficient, our own networks were trained as well. The training datasets, used architectures and results for scene and object level classification will be presented below.

A. *Scene Level Classification*

Arguably the most straightforward method to get information about the room type in a given 3D model is to directly classify images rendered in each of the rooms. Currently the biggest dataset for scene classification is the Places 365 dataset [5]. It contains 365 different scene categories.

In order to classify the rendered images, first we identified relevant categories in the Places 365 dataset which also appear in our ten point cloud data sets of indoor scenes [1]. We merged the training images of several visually similar scene types, as for example bathroom and shower, to cover more possible variances in our training data. Furthermore, we introduced a so called *"background"* class to be able to label views that hold few distinguishing features and cannot clearly be assigned to any of the known classes (see Table I). A ResNet-50 [18] was trained on this data, reaching a validation accuracy of 88% after 40 Epochs of training, after which the validation accuracy stagnated while validation losses increased, indicative of overfitting. The resulting CNN was able to generalize on the rendered views. However, it must be considered that many scenes are ambiguous or are difficult to label (e.g. if multiple room types are visible in a single view due to an open door). What made the task difficult as well was the fact that some rooms were either ambiguously furnished (see Figure 3) or not furnished at all.

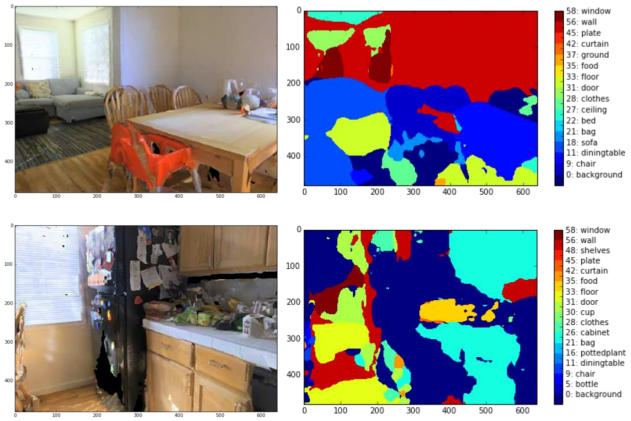

Fig. 4: Results of the object detection module. Each pixel in the input images is labeled with the highest scoring class.

B. Object Level Detection

Scene level classification networks learn the occurrence of class-specific patterns implicitly. We, however, know that certain objects are highly indicative of the scene type, and we can employ detectors for them to provide additional information for the scene type labeling.

When detecting objects in the rendered mesh views, one major problem is the varying image quality. As can be seen in Figures 3 and 4, bigger objects like sofas and tables are generally less affected by the rendering errors. An additional problem is the high degree of clutter and variability in natural household scenes. CNNs, however, are becoming very apt at dealing with these kinds of problems.

Most of the publicly available methods for object detection/segmentation are trained on the popular PASCAL VOC dataset. Unfortunately, most of the 20 categories in this dataset are not relevant for our problem (e.g. aeroplane, bicycle, cat, etc). At the time of writing, there was only one network that was trained with fitting classes for our problem, a Fully Convolutional Network (FCN) [6] based on VGG16 and trained using the 59 labels of the PASCAL Context [19] annotations of the original PASCAL dataset.

As can be seen in Figure 4, the detection results are varying. While the top image is explained reasonably well, the lower image shows significantly more errors. It must be noted, however, that the visualization shows only the most likely class, while in reality a full probability distribution holding additional information is predicted for each pixel. This probabilistic output (cf. video supplement and Figures 1&2) will be used to enrich the scene level classification.

V. Information Fusion with a CRF

This section defines the different statistics and scores that are used in our CRF's potential functions. Then, the optimization of the inference and learning steps are described.

A. Objects to Room Type Correlation

In order to learn a model for the link between object detections and room types, we evaluated several datasets like COCO, SUN, ImageNet and NYU-Depth V2. The requirement was to find a dataset, that provides scene information, is densely labeled (since we want to get the probabilities of objects appearing in a scene, sparse labeling is not an option), and uses relevant indoor object categories.

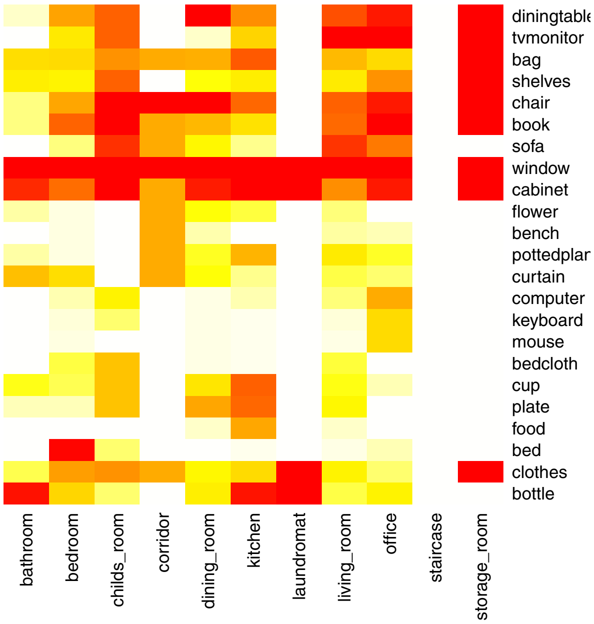

Fig. 5: Occurrence frequencies of the most important object categories in scene types, computed for the NYU-Depth V2 dataset. The object categories are ordered based on a complete linkage hierarchical clustering, which places the most similar (groups of) objects near to each other.

The only dataset that satisfied all the requirements was NYU-Depth V2 [20]. Figure 5 visualizes the probabilities of different objects appearing in each of the eleven scene types. Object categories that are detectable by the FCN but that do not appear at all in NYU-Depth V2 (e.g. airplanes) are left out, as well as categories that appear in all scene types (e.g. floors, ceilings, walls and doors), since they would not influence the CRF's inference results.

B. Proposed Conditional Random Field

In this section we show how to merge the different cues to generate a more accurate and robust final scene classification. In particular, we frame the problem as a conditional random field in which the vertices are defined as the pixels of the final labeled floor plan. CRFs are capable of representing complex correlation between input data, hence a convenient approach for fusing heterogeneous information. Furthermore it is robust against data noise.

Given the geometric room pre-segmentation (from [1]) $\mathbf{G} \in \mathbb{R}^{mxn}$ we model each pixel i as a set of random variables consisting of the corresponding probability distributions of the scene class labeling $\mathbf{s}_i \in [0,1]^L$ and the object $\mathbf{o}_i \in [0,1]^K$, where L/K represents the number of scene/object classes. Let $\mathbf{y} = \{y_0, \dots, y_V\}$ denote the set of vertices with $y_i \in \{0, \dots, L\}$ encoding the final semantic scene label of the ith pixel and V the number of pixels to be labeled. We formulate the energy $E(y)$ for final scene class labeling as the sum of potentials representing the scene classification ϕ_{scene}, the object occurrence in a room ϕ_{obj} and the relation between geometric and semantic labels ϕ_{hom}.

$$E(\mathbf{y}) = \sum_{i=0}^{V} w_c \phi_{scene}(y_i, \mathbf{s}_i) + \sum_{i=0}^{V} \mathbf{w}_{obj}^T \phi_{obj}(y_i, \mathbf{o}_i) + \sum_{i=0}^{V} w_{hom} \phi_{hom}(\mathbf{y}, \mathbf{G}) \quad (1)$$

The terms $w_c, w_{hom} \in \mathbb{R}$ and $\mathbf{w}_{obj} \in \mathbb{R}^4$ represent the weights of the CRF obtained during the training phase by applying the structural SVM presented in [21]. While the first two potentials (ϕ_{scene} and ϕ_{obj}) are unary potentials and refer just on the corresponding pixel information, the ϕ_{hom} expression represents a pairwise potential. Therefore we defined the CRF as a fully connected graph. We will now explain the respective potentials in more detail.

a) Scene-CNN: There are two alternative options when considering scene classification results, denoted ϕ_{scene}^{cnn} or ϕ_{scene}^{conf}. In the simplest case, based on the outcome of the scene label classifier, the potential ϕ_{scene}^{cnn} is the confidence s_y of the algorithm being the current label y.

$$\phi_{scene}^{cnn}(y, \mathbf{s}) = s_y \quad (2)$$

Although the scene label classifier by itself shows good results, it still produces confusions due to the ambiguous occurrence of views in different rooms. For instance, a bed can be observed in a bedroom as well as in a child's room. To overcome this issue, we include the confusions of the training data into the potential. Given the confidence s_l of the scene label classifier being the label l and the training probability $p_{trn}(y, l)$ (obtained over all pixels in the training data) that label l is confused with the current pixel label y, we define the potential as:

$$\phi_{scene}^{conf}(y, \mathbf{s}) = \sum_{l=0}^{L} p_{trn}(y, l) s_l \quad (3)$$

b) Object Occurrence: In addition to the scene label, another valuable information is the combination of objects detected and their statistical occurrence for the current scene label. For probability distribution $\mathbf{o}_i$ for the ith pixel (corresponding to the ith semantic node y_i) over K object categories we define the object occurrence potential as:

$$\phi_{obj}(y, \mathbf{o}) = [\phi_{obj}^1, \phi_{obj}^2, \phi_{obj}^3, \phi_{obj}^4] \quad (4)$$

$$\phi_{obj}^i = \mathbf{o}^T \mathbf{p}_{stat} \quad (5)$$

The term $\mathbf{p}_{stat}$ describes the statistical correlation between the object groups and scene labels. To reduce the number of weights that need to be trained, we divide objects into four groups that have separate potentials ϕ_{obj}^{1-4}, i.e. weights. We can see in Figure 5 that some objects have similar occurrence frequencies. Thus, we grouped objects approximately corresponding to the top, middle and bottom thirds in the ordering, i.e (mostly) larger furniture pieces, smaller objects of daily use, and kitchen items. Bed and sofa form a separate fourth cluster (as those two are easily confused by the FCN but constitute important cues). Note that per-object predictions are considered by the CRF, the grouping is only forcing some categories to share their learned weights, due to efficiency.

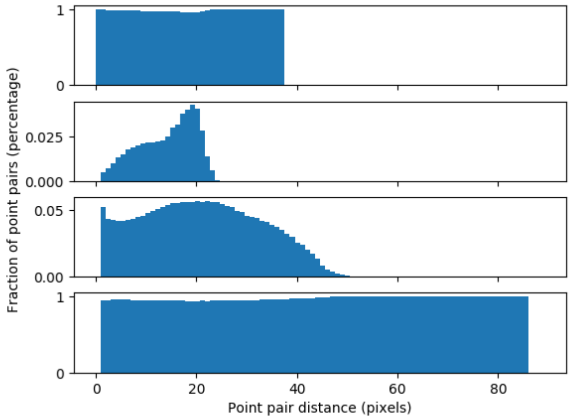

Fig. 6: Distance-dependent proportion of pixel pairs that have the same or different semantic labels in the case of shared or distinct geometric pre-segmentation clusters. The plots from top to bottom correspond to the equations in Eq. 6.

c) Homogeneity: Rather than enforcing a strict Gaussian smoothness potential that would loose smaller details, the homogeneity potential is split into four different events depicted in Eq. 6. The events represent the relation of the semantic labeling (y_i and y_j) and geometric clusters (g_i and g_j) of a pixel pair (i, j).

$$\begin{aligned} &y_i = y_j \ and \ g_i = g_j \\ &y_i \neq y_j \ and \ g_i = g_j \\ &y_i = y_j \ and \ g_i \neq g_j \\ &y_i \neq y_j \ and \ g_i \neq g_j \end{aligned} \quad (6)$$

As shown in Eq. 7, the potential is defined by function $f(y_i, g_i, y_j, g_j)$.

$$\phi_{hom}(\mathbf{y}, \mathbf{G}) = \sum_{j=0}^{V \setminus \{i\}} f(y_i, g_i, y_j, g_j) \quad (7)$$

As for the scene classification result, also for the homogeneity two alternative options are implemented which differ in the computation of function f: Firstly, f is defined as a distance-dependent statistic f on all pixel pairs representing the four events (this case will be referred to as ϕ_{hom}^{stat}). These statistics were learned during training (see Figure 6) over all training samples. The second option, f is represented by a fixed empirically generated cost value for each event in 6 weighted by a 2D Gaussian (referred to as ϕ_{hom}^{gauss}).

C. Inference

We infer our graph model by computing the label y^* resulting in the maximum energy

$$\mathbf{y}^* = \underset{\mathbf{y}}{argmax}\, E(\mathbf{y}) \tag{8}$$

where $E(\mathbf{y})$ is the energy defined in Eq. 1. Since the CRF is fully connected and a pairwise potential is applied, loops can occur. Hence, an exact inference is not possible. Therefore, we employ a Gibbs sampling approach for the inference. The nodes $\mathbf{y}$ are initialized randomly weighted by the probability distribution of the Scene Classification CNN. Furthermore, the inference order of the nodes is randomized.

VI. Evaluation and Comparisons

We use the dataset recorded by [1], which consists of 10 apartments, extending it with 3D ground truth information from which we can generate ground truth for any image rendered inside the mesh. Qualitative results are shown in Fig. 9. Specifically, Fig. 9 a) shows a top-down view of the dataset, while b) shows the pre-segmentation of [1]. The ground truth segmentation is shown in Fig. 9 c), color coded according to semantic label. In Fig. 9 d) we show the results of the raw Scene CNN predictions, while Fig. 9 e), f) and g) show the results of selected CRF variants. In both cases, for each pixel the label of the most likely class is shown.

A. Majority Voting

We use a voting approach as a baseline for assigning semantic labels to the 2D pre-segmentation of the environment based on the output of the Scene Classification CNN for the rendered images. Specifically, we project down each virtual view on the 2D pre-segmentation, and label each pixel in its view cone as the class associated with the highest probability from the CNN distribution. Further, we label each segment in the pre-segmentation as the class that most pixels in the segment are labeled as. Intuitively, this simple method allows us to inspect the quality of the results if we rely solely on the Scene Classification CNN predictions and do not modify the pre-segmentation. We report the labeling results averaged over all datasets in Table II.

B. Fully Connected CRF with Mean-Field Inference

Using the approach of [22], we define a dense CRF with unary and binary potentials. The CRF is defined over the pixels of the pre-segmentation, where each pixel is modeled as a random variable. As unary potentials we use the probability distributions returned by the Scene Classification CNN. As in Sec. VI-A, we first project down each virtual view on the 2D pre-segmentation, and assign the CNN output to each pixel covered by its view cone. In case multiple views cover the same pixel, we select the view with the highest maximum value assigned to one of the classes.

	ϕ_{scene}^{cnn}	ϕ_{scene}^{conf}	ϕ_{hom}^{stat}	ϕ_{hom}^{gauss}	ϕ_{obj}	avg. Acc
scene-CNN						36.55
Maj. Vot.						58.99
G-CRF [22]						63.96
CRF 1	x	–	–	–	–	44.33
CRF 2	–	x	–	–	–	50.37
CRF 3	–	x	–	x	–	67.81
CRF 4	–	x	–	x	x	**67.84**
CRF 5	–	x	x	–	x	64.25

TABLE II: Average per-pixel accuracy on 10 folds for the baselines and our CRF variants; **bold** marks the best score.

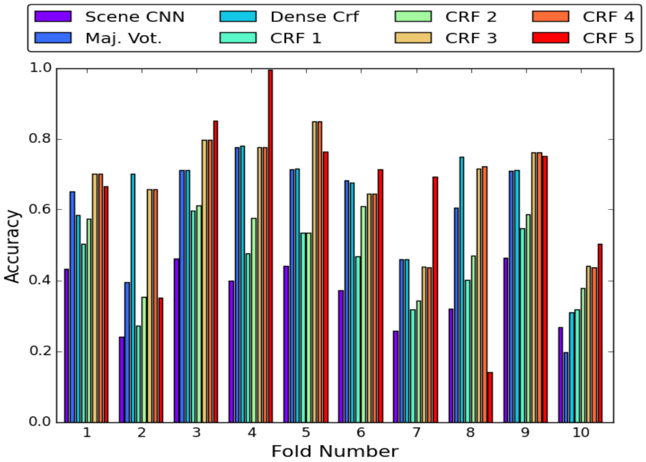

Fig. 7: Fold-wise comparison of the presented methods based on the per pixel accuracy.

The binary potentials are defined over pairs of pixels from the 2D pre-segmentation. They are represented as Gaussian kernels to allow for the efficient mean-field inference (hence we will refer to is as a dense G-CRF). These enforce both appearance consistency (i.e. nearby pixels should have the same label) as well as smoothness (removes small regions):

$$\begin{aligned} k(f_i, f_j) = w^{(1)} exp\left(-\frac{|p_i - p_j|^2}{2\theta_\alpha^2} - \frac{|I_i - I_j|^2}{2\theta_\beta^2}\right) \\ + w^{(2)} exp\left(-\frac{|p_i - p_j|^2}{2\theta_\gamma^2}\right) \end{aligned} \tag{9}$$

Where f_i and f_j are feature vectors for pixels i and j, p_i and p_j represent the pixel positions in the image, and I_i and I_j are the pixel colors. We learn the parameters ($w^{(1)}$, θ_α, $w^{(2)}$, θ_β, θ_γ) from data, through a grid search over the parameters space as suggested by [22]. We report the results averaged over all datasets in Table II (cf. Figure 9 e).

C. Quantiative Results and Discussion

For training the proposed CRF method, a 10-fold leave-one-out cross-validation is applied. In each iteration, the CRF inference is run on the one apartment left out after training on the other nine samples. The different baselines and potential variants are compared in Table II and Figure 7.

As expected, the majority voting approach reflects the scene CNN predictions – accurate only in some cases, and importantly, limited to the segments created by the pre-segmentation. Using the fully connected CRF with Gaussian potentials from [22] we are able to alleviate some of the limitations, by combining the scene CNN predictions with

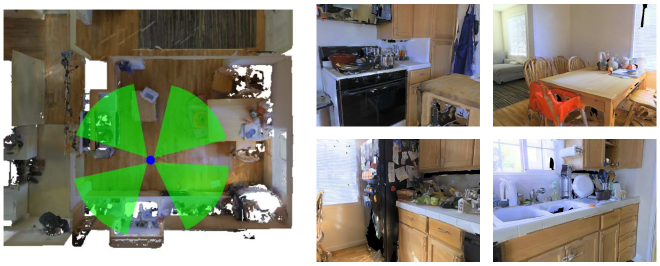

Fig. 8: An ambiguous room that was split by several methods into a separate dining room and kitchen in Figure 9 row 1.

binary potentials encoding smoothness and consistency in the pre-segmentation. Our method shows even greater flexibility, owing to our use of non-Gaussian (and thus more versatile) potentials in the CRF. We attribute our failure cases to the poor performance of the pretrained CNNs, as well as, in some instances, to the ambiguity inherent in choosing a semantic label (i.e. kitchen versus dining room).

Balancing of the strong local cues from the scene classification and geometric pre-segmentation is not straightforward, as exemplified in Figure 8. Inspecting the results in Figure 9 suggests that G-CRF relies too heavily on the geometric cues and our CRF 4 variant on the scene classification. The CRF 5 variant produces visually appealing results thanks to the learned statistics, which yield very good to very bad accuracies (of the top predictions at least) depending on the fold, with slightly lower mean accuracy than CRF 4.

VII. Conclusions

We have presented an automatic method for assigning semantic labels to rooms from RGBD data. In summary, the proposed CRF-based cue integration worked better than existing alternatives, and was able to correctly label a large portion of the tested apartments, even in the case of incorrect geometric priors. Still, challenges remain, partly due to the strong biases in the scene classification (e.g. white walls were classified as bathrooms with very high confidences), but also due to the "hallucinations" of the FCN (e.g. predicting a plate on top of an empty table in Figure 4). Some of these challenges could be addressed by using more training data, and by including more potentials into the energy function, at the cost of added computational complexity. We hope that by making our labeled dataset available we can stimulate the generation of new ideas on how to address this yet unsolved challenge. For future research we will continue to investigate the performance of CRF (e.g. cell complex for node representation), and also make use of the depth information from the rendered views.

Acknowledgment

This work has been funded by Robert Bosch, Corporate Research, the Swedish Foundation for Strategic Research (SSF) through its Center for Autonomous Systems and the Swedish Research Council (VR) under grant C0475401. The authors thank Dr. Gellért Máttyus for the helpful discussions.

References

[1] R. Ambruş, S. Claici, and A. Wendt, "Automatic room segmentation from unstructured 3-d data of indoor environments," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 749–756, April 2017.

[2] R. Bormann, F. Jordan, W. Li, J. Hampp *et al.*, "Room segmentation: Survey, implementation, and analysis," in *IEEE Intern. Conf. on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 1019–1026.

[3] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International journal of computer vision*, vol. 42, no. 3, pp. 145–175, 2001.

[4] J. Wu, H. I. Christensen, and J. M. Rehg, "Visual place categorization: Problem, dataset, and algorithm," in *IEEE Intern. Conf. on Intelligent Robots and Systems (IROS)*. IEEE, 2009, pp. 4763–4770.

[5] B. Zhou, A. Khosla, A. Lapedriza, A. Torralba, and A. Oliva, "Places: An Image Database for Deep Scene Understanding," *Arxiv preprint*, 2016. [Online]. Available: http://arxiv.org/pdf/hep-th/0206225

[6] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 3431–3440.

[7] Y. Liao, S. Kodagoda, Y. Wang, L. Shi, and Y. Liu, "Understand scene categories by objects: A semantic regularized scene classifier using convolutional neural networks," in *IEEE Intern. Conf. on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 2318–2325.

[8] P. Uršič, A. Leonardis, D. Skočaj, and M. Kristan, "Learning part-based spatial models for laser-vision-based room categorization," *The Intern. Jour. of Robotics Research*, vol. 36, no. 4, pp. 379 – 402, 2017.

[9] L. Shi, S. Kodagoda, and M. Piccardi, "Towards simultaneous place classification and object detection based on conditional random field with multiple cues," in *IEEE Intern. Conf. on Intelligent Robots and Systems (IROS)*, 2013.

[10] J. Yao, S. Fidler, and R. Urtasun, "Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012, pp. 702–709.

[11] D. Lin, S. Fidler, and R. Urtasun, "Holistic scene understanding for 3d object detection with rgbd cameras," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1417–1424.

[12] H. S. Koppula, A. Anand, T. Joachims, and A. Saxena, "Semantic labeling of 3d point clouds for indoor scenes," in *Advances in neural information processing systems (NIPS)*, 2011, pp. 244–252.

[13] A. Hermans, G. Floros, and B. Leibe, "Dense 3d semantic mapping of indoor scenes from rgb-d images," in *IEEE Intern. Conf. on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 2631–2638.

[14] K. Sjöö, "Semantic map segmentation using function-based energy maximization," in *IEEE Intern. Conf. on Robotics and Automation (ICRA)*. IEEE, 2012, pp. 4066–4073.

[15] A. Pronobis and P. Jensfelt, "Large-scale semantic mapping and reasoning with heterogeneous modalities," in *IEEE Intern. Conf. on Robotics and Automation (ICRA)*. IEEE, 2012, pp. 3515–3522.

[16] N. Sünderhauf, F. Dayoub, S. McMahon, B. Talbot, R. Schulz, P. Corke, G. Wyeth, B. Upcroft, and M. Milford, "Place categorization and semantic mapping on a mobile robot," in *IEEE Intern. Conf. on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 5729–5736.

[17] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. Leonard, "Past, present, and future of simultaneous localization and mapping: Towards the robust-perception age," *IEEE Transactions on Robotics*, vol. 32, no. 6, p. 13091332, 2016.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. of the IEEE confer. on computer vision and pattern recognition (CVPR)*, 2016, pp. 770–778.

[19] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. Yuille, "The role of context for object detection and semantic segmentation in the wild," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[20] P. K. Nathan Silberman, Derek Hoiem and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *European Conf. on Computer Vision (ECCV) – LNCS 7576*, 2012, pp. 746–760.

[21] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, "Large margin methods for structured and interdependent output variables," *J. of Machine Learning Research*, vol. 6, pp. 1453–1484, Dec. 2005. [Online]. Available: http://jmlr.org/papers/v6/tsochantaridis05a.html

[22] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected CRFs with Gaussian edge potentials," in *Advances in Neural Information Processing Systems 24*, 2011, pp. 109–117.

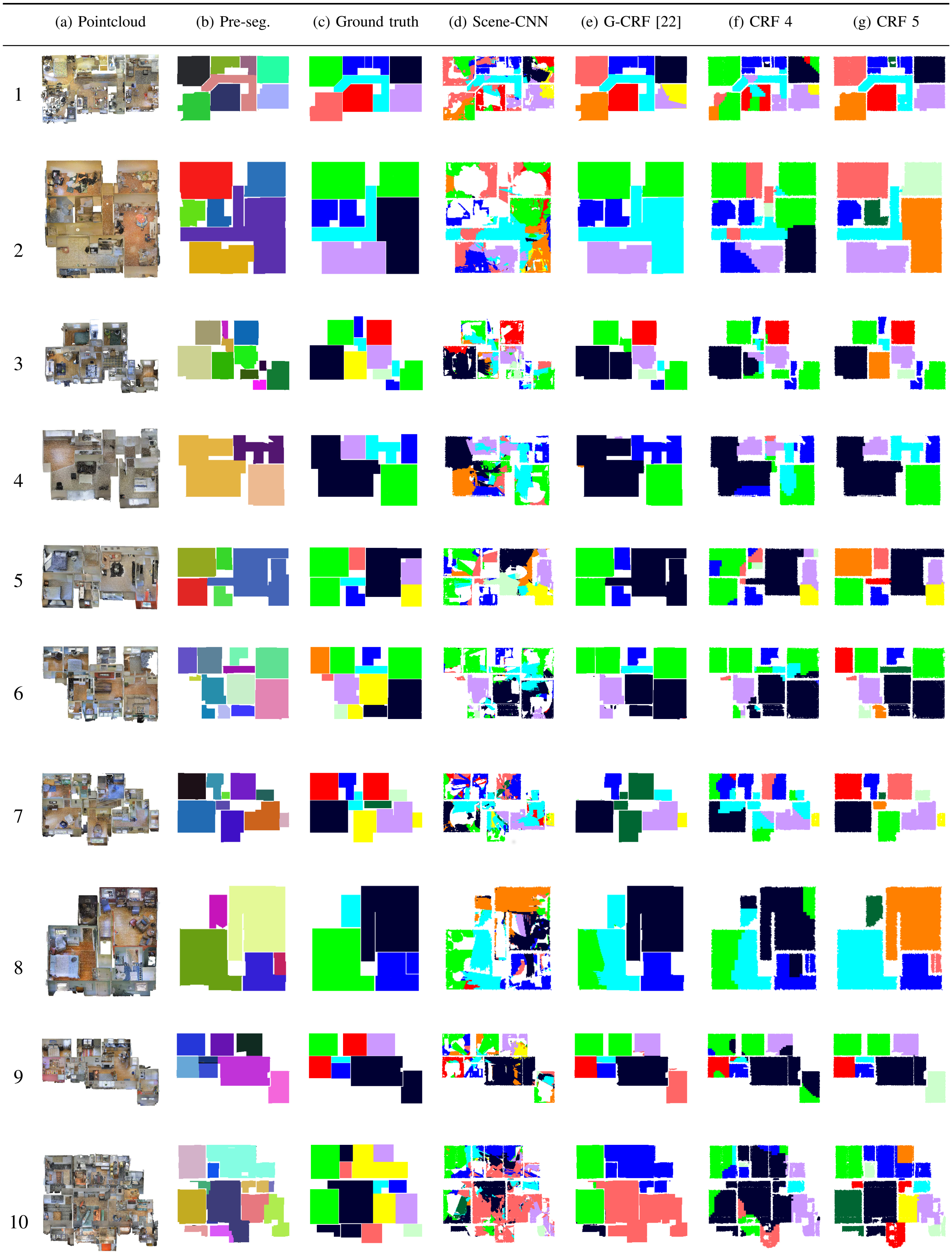

Fig. 9: Qualitative results: (a) original data [1], with ceilings removed for clarity; (b) pre-segmentation based on geometric primitives [1], arbitrarily colored; (c) the ground truth labelling, from here colored according to semantic class, as shown in Table I; (d) raw prediction maximas for the scene view CNN; (e) labelling using the CRF of [22]; (f,g) our results.