3D Semantic Segmentation of Modular Furniture using rjMCMC

Ishrat Badami[1]* Manu Tom [2]* Markus Mathias[3] Bastian Leibe[4]

Visual Computing Institute, Computer Vision Group
RWTH Aachen University [1,3,4]

Photogrammetry and Remote Sensing Group
ETH Zurich [2]

{badami, mathias, leibe}@vision.rwth-aachen.de manu.tom@geod.baug.ethz.ch

Abstract

In this paper we propose a novel approach to identify and label the structural elements of furniture e.g. *wardrobes, cabinets* etc. *Given a furniture item, the subdivision into its structural components like doors, drawers and shelves is difficult as the number of components and their spatial arrangements varies severely. Furthermore, structural elements are primarily distinguished by their function rather than by unique color or texture based appearance features. It is therefore difficult to classify them, even if their correct spatial extent were known. In our approach we jointly estimate the number of functional units, their spatial structure, and their corresponding labels by using reversible jump MCMC (rjMCMC), a method well suited for optimization on spaces of varying dimensions (the number of structural elements). Optionally, our system permits to invoke depth information* e.g. *from RGB-D cameras, which are already frequently mounted on mobile robot platforms. We show a considerable improvement over a baseline method even without using depth data, and an additional performance gain when depth input is enabled.*

1. Introduction

Visual understanding of indoor scenes is a crucial task in robotics. Accurate semantic labeling and object classification provides rich information about the complex indoor environment, which is crucial for navigation, manipulation, and interaction with the scene. Most methods focus on coarse scene understanding. They identify walkable surfaces [11] and objects [8], estimate the 3D geometry [41, 25, 33], and provide rough labels for the different entities [37].

Only a limited amount of work aims at a more detailed object level segmentation [26]. Such a detailed analysis of the object semantics will allow autonomous robots to perform more advanced, human-like interactions in indoor environments, like arranging groceries in the kitchen, sorting books on the book shelves, opening/closing a locker *etc*. These tasks require a detailed structural inference and the classification of the spatially variable parts of the furniture.

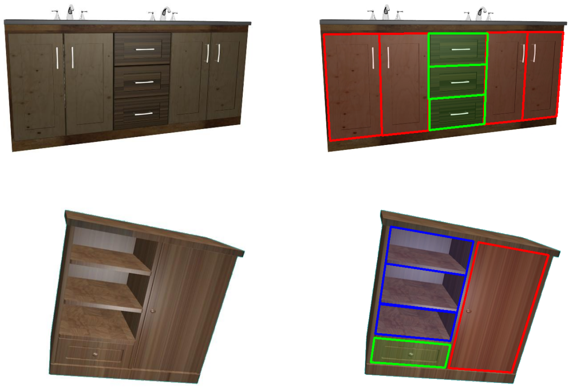


Figure 1: Semantic segmentation of modular furniture in RGBD images: Left column is the input front face of the furniture and right column is the segmentation output (door, drawer and shelf).

Our segmentation problem differs from conventional semantic segmentation of the entire scene. A noisy, pixelwise segmentation would not be sufficient in order to infer the structural information that is required for an interaction. Furthermore, as also noted by Zheng *et al*. [40], traditional visual cues such as color and texture are not particularly useful for labeling furniture items due to their often uniformly colored and textured appearance.

In this paper, we present an approach to perform fine grained segmentation of *modular* furniture like cabinets, wardrobes, cupboards, or lockers into their functional units, namely drawers, doors and shelves. These furniture items follow a modular design as their entire volume is composed of a variable number of functional units, the so called *interaction elements* (IEs) (see Figure 1). Additionally, most fur-

*indicates equal contribution.

niture items are not only rectangular as a whole, but also the internal structure follows a rectangular subdivision scheme. We exploit these modular properties in our optimization and propose a two stage segmentation approach. In the first stage we generate an overcomplete set of rectangle proposals such that each true IE is represented by a rectangle in the set. A rectangle proposal consists of the rectangle itself, and a class label distribution for that rectangle. In the second stage we select a subset (with unknown size) of the proposals that represent our final semantic segmentation of the furniture into interaction elements. We formulate the proposal selection as rjMCMC based energy minimization problem.

After the advent of 3D cameras, many previously proposed RGB based methods are improved by using additional depth information [13, 33, 30]. Undoubtedly, depth provides powerful additional information when estimating the real object size or geometry of the scene. We show that using RGB-D images also improve the segmentation of furniture significantly. Our method is able to include such depth data when available.

Contributions.

1. We propose a novel rjMCMC based furniture segmentation method which achieves state-of-the-art results. Unlike [26], our method allows to predict structure and labels jointly within a single optimization.
2. We introduce a new data-driven augmentation method to generate rectangle proposals leading to a significant higher recall, *i.e.* the set of IE proposals better reflect the true IEs.
3. We present a new 3D furniture dataset with corresponding ground truth annotations.

2. Related Work

Segmentation approaches. Segmentation can be performed with or without using the semantic information. Algorithms that do not invoke semantic information, cluster the image pixels based on feature similarities [2, 14, 24, 4]. The resulting segments naturally do not carry any semantic label information. A wide range of segmentation methods also aim to get semantic information from the segments, *e.g.* by classifying grouped pixels using supervised learning techniques [5]. Alternatively, semantic knowledge may influence the segmentation itself [29, 1, 15, 16]. Most instance segmentation methods are based on combining an object detector output/region proposals with segmentation [10, 6]. Such approaches are difficult to apply in our case, as we exhibit a high inter class similarity and object detectors do not capture relationships between the instances. Approaches based on Hough voting [27] struggle with classes of unconstrained size and aspect ratios. Finally, approaches based on deep learning require significantly more data to train.

Indoor scene parsing approaches. The majority of manmade objects can be modeled as a combination of different geometric shapes [9, 39]. Han *et al.* [9] and Zhao *et al.* [39] advances the pixel grouping to higher level of parametric shape clustering in a hierarchical manner, such that at each level the corresponding cluster represents a predefined geometric shape. These approaches are very successful at capturing geometric structure but they lack semantic label information. Gupta *et al.* [8] utilizes general and object-class specific appearance features as well as contextual information *e.g.* object boundaries for semantic segmentation. Their approach mainly focuses on local patterns rather than a global structure.

Facade parsing approaches. Parsing building facades into the architectural elements *e.g.* windows, walls, roof and parsing furniture into interaction elements *e.g.* door, drawer, shelf appear quite similar. Both contain rectangular grid-like structures which have to be determined. The Facade parsing problem is tackled from different directions. Müller *et al.* [22] detect repetitive structures in large, grid like facades in order to obtain meaningful hierarchical facade subdivisions. Several methods exploit high-level information in the form of shape grammars [32] combined with low-level appearance cues derived from an image [36, 34, 28]. The underlying grammars can either be designed manually [20] or learned from data [18]. Mathias *et al.* [19] combine low-level, mid-level and high-level cues in form of a pixel-wise semantic segmentation, the output of an object detectors, and a shape grammar respectively. These shape grammar based techniques assume a strong, style specific structure and do not generalize well to different architectural styles [21]. In case of furniture parsing, we would require a very generic grammar, which could only weakly impose a structural layout. In case of facade parsing, architectural elements show significant inter-class variance in color and texture and often exhibit regular and repetitive structure. Both of these properties are absent in case of furniture parsing problems.

Furniture Parsing. Lim *et al.* [17] addresses the problem of instance level furniture detection and pose estimation by using a predefined 3D CAD model. The approach targets to find the same model within a 3D scene. Our goal is to parse any modular furniture. Pohlen *et al.* [26] addresses the problem of furniture segmentation from a single image. A huge set of possible furniture elements is generated from the input image; the final segmentation results from selecting a suitable element subset. Our approach closely follows the method described in [26]. We adopt the described appearance model by incorporating depth information which improves the label inference performance by almost 33%. As the number of furniture elements is variable, [26] performs optimization independently for various possible numbers of elements using MCMC. In order to find the best solution among the different Markov chains, the most modular solution is chosen. In contrast to this we perform a single multi objective optimization using a trans-dimensional variant of Markov chain namely rjMCMC [7]. This seamlessly combines the optimization of correct number of parts, their spatial arrangements, and the class label inference. Our approach therefore reduces overall computational cost and results in faster convergence.

Varying dimension problems. There are a number of challenging inference problems *i.e*. segmentation [35], multi-object tracking [31], scene parsing [38] *etc*. where the dimension of the model of inference is not fixed. Usually, Bayesian approaches are suitable for such problems. Reversible jump MCMC is capable of computing such inference by jumping between subspaces of differing dimensionality. Tu *et al*. [35] propose a data driven MCMC for image segmentation. Here the Markov chain dynamics is governed by importance probabilities designed using the image data. There are seven image models for intensity and color which describe the segments. The solution is obtained by maximizing the joint posterior of these segments using the defined image models. Zhao *et al*. [38] propose a scene parsing approach using a stochastic grammar model. This model is a hierarchical structure which includes scene category, functional groups, functional objects, functional parts, and 3D geometric shapes in a top down fashion. Starting from extracted 3D shapes from the image, the objects at every level are clustered according to their function and appearance in a bottom up fashion using rjMCMC. Smith *et al*. [31] developed the rjMCMC particle filter framework for robust tracking of a variable number of targets. In each of the discussed methods, a set of four reversible jump moves such as birth, death, update, and swap are designed to search though trans-dimensional space. The different move types are selected based on a time varying prior which depends on the previous state of the Markov chain.

3. Proposed Approach

In the first stage of the algorithm we generate an over-complete set of proposals with the goal to generate at least one matching proposal for each true interaction element (IE) of the furniture item. Having an over-complete set of proposals allows us to compute the semantic segmentation by performing subset selection. This is formulated as an energy minimization problem in a high dimensional state space detailed in Section 3.2.

3.1. Proposal Generation

We assume that the front face of the furniture item has already been extracted and rectified during pre-processing (*e.g*. using [12]). As such our search space is restricted to axis-aligned, rectangular IEs. Following the approach in [26] first we *detect* rectangles from the edge map and then assign each IE candidate a *weight* and a class *label* probability.

3.1.1 Rectangle Detection

We follow two strategies to generate a multitude of rectangles that serve as IE candidates.

Rectangle Set Generation by Pohlen *et al*. [26]. A semantic edge map is generated from the image in a supervised manner using random forests [3]. In the edge map horizontal and vertical lines are detected through Hough transform. Rectangle hypothesis are then generated as a convex hull formed by iteratively sampling two horizontal and two vertical lines. A hypothesis is accepted as a valid rectangle if the maximum distance from any boundary pixel of the rectangle to the closest edge pixel in the image is below the set threshold. This procedure leads to a good initial set of possible IEs, but needs further refinement which we achieve by the following augmentation method.

Rectangle Set Augmentation. Due to complex textures, bad lighting conditions, or skewed perspective angles, the initial strategy for the rectangle set generation is insufficient. We propose to extend the set of rectangles including *splitting* and *merging* operations on the existing rectangle set. As an additional benefit, this step of proposal set augmentation mimics costly online data-driven split and merge moves usually defined in rjMCMC optimizations.

To keep the problem tractable we cluster the initially detected rectangles such that all rectangles of a cluster overlap (IoU) by more than 95% and only keep one representative rectangle of each cluster. We then perform two types of rectangle set augmentations:

- **Split augmentation** divides a rectangle into two rectangles. We iterate over each rectangle in the proposal pool. First, the horizontal and vertical edge pixel histogram are computed. A rectangle is subdivided into two new rectangles at every peak in edge histogram greater than a predefined threshold.
- **Merge augmentation** combines pairs of rectangles. All neighboring rectangles of nearly the same height are merged horizontally, rectangles of similar width vertically.

All newly generated rectangles are only added to the pool if they do not considerably overlap with already existing rectangles (using a 85% IoU threshold).

Inclusion of Depth Information To further improve IE detection recall, our method is able to include depth information in this stage of the algorithm. To that end we extended the semantic edge detection forest to learn semantic edges based on depth maps. We fuse the resulting edge with the initial edge map via the pixel-wise `or` operator. Although we will incorporate depth information throughout our algorithm, we are able to disable depth and roll back to an RGB-only setup.

3.1.2 Rectangle Weighting

Given the image I_m and rectangle r, the weight of the IE with label $l \in \{door, drawer, shelf\}$ is quantified by the conditional probability as in Equation 1.

$$\underbrace{p(l \mid r, I_m)}_{\text{Label posterior}} \propto \underbrace{p(I_m \mid r, l)}_{\text{Appearance likelihood}} \underbrace{p(r \mid l)}_{\text{Shape prior}} \underbrace{p(l)}_{\text{Label prior}} \tag{1}$$

Due to a high visual inter-class similarity between the IEs of a single furniture instance, color and texture appearance cues are not sufficiently discriminative. We therefore exploit a set of common traits that can be observed for IEs of the same class. These traits are the rectangle's aspect ratio, the position of the handle, and the edge profile.

Following Pohlen *et al.* [26], we learn a codebook representation over J codewords $p^{(1,l)}, ..., p^{(J,l)} \in \mathbb{R}^{M^2}$ based on the $M \times M$ rescaled gradient magnitude image for each of the classes independently. The objective for the training procedure is to approximate each training element as a linear combination of codebook entries. Depending on how well a new rectangle defined over an image region f_{r,I_m} can be expressed by any of the learned codebooks, the **appearance likelihood** is defined as follows:

$$p(I_m \mid r, l) \propto \max_{\substack{\pi \in [0,1]^J \\ \sum_j \pi_j = 1}} \exp\left(- \left\| f_{r,I_m} - \sum_{j=1}^{J} \pi_j p^{(j,l)} \right\|_2^2 \right), \tag{2}$$

where $\pi_1, \ldots, \pi_J$ are the codebook coefficients.

The **shape prior** is estimated with a probabilistic support vector machine using relative height, width and aspect ratio as features.

Finally, the **label prior** represents the observed label frequencies in the training data.

Additional details can be found in [26].

Inclusion of Depth Information From the results in [26] it is apparent that proposed weighting scheme works well for doors and drawers but suffers a high confusion between the drawer and the shelf class. While these classes are difficult to differentiate visually, depth cues should clearly improve performance. Here, we incorporate depth information over the shape prior term by using the relative depth of each rectangle compared to the front face of the furniture, and the aspect ratio of depth in relation to width and height.

3.2. Proposal Selection

From the set of detected rectangles we wish to choose a subset of rectangles that best explains the image I_m. Let $P := \{(r_k, l_k) | k = 1, \ldots, K\}$ be a subset of K rectangles. Our goal is to find the best subset of rectangles $\hat{S} \subset P$ such that

$$\hat{S} = \operatorname*{argmax}_{\hat{S}} p(\hat{S} | I_m). \tag{3}$$

We jointly estimate the true number of IEs K, their spatial arrangements r_k and their respective class labels l_k. The optimization is formalized as a multi-objective optimization problem:

$$p(S|I_m) \propto e^{-E_{total}(S)}, \tag{4}$$

where

$$\begin{aligned} E_{total}(S) = E_c(S) + E_o(S) + E_w(S) + \\ E_{ls}(S) + E_{lv}(S) + E_s(S) \end{aligned} \tag{5}$$

Maximizing $p(S|I_m)$ is equivalent to minimizing the energy of Equation 5. Each energy term in $E_{total}(S)$ captures a dedicated property as described in the following.

The Cover energy E_c secures a maximum coverage of the area Ω of the furniture's face (Figure 3(a)).

$$E_c = -\frac{1}{\Omega} \left(\bigcup_{k=1}^{K} |r_k| - \sum_{k \neq j} |r_k \cap r_j| \right) \tag{6}$$

The Overlap energy E_o ensures minimum overlap between all pairs of rectangles in a state (Figure 3(b)).

$$E_o = \frac{1}{\lambda_o \cdot \binom{K}{2}} \sum_{k \neq j} \frac{|r_j \cap r_k|}{\min(|r_j|, |r_k|))} \tag{7}$$

where $\lambda_o = 0.15$ is an empirically determined overlap parameter.

The Rectangle weight energy E_w choses rectangles with a high appearance likelihood (Figure 3(c)).

$$E_w = \frac{1}{K} \sum_{k=1}^{K} 1 - p(l_k \mid r_k, I_m) \tag{8}$$

The Label smoothing energy E_{ls} encourages label consistency given the structure of the furniture. For modular furniture, a modularity tree Γ can be built. The entire face of the furniture shown in Figure 2 defines the root of the tree. Elements that are similar in structure are clustered and

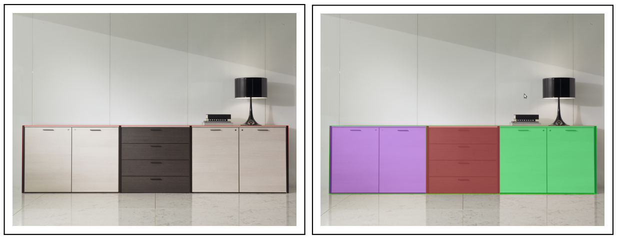


Figure 2: Rectangle clusters within a furniture. Two child rectangles (door) in purple cluster, four child (drawer) rectangles in red cluster and two child (door) rectangles in green cluster.

define the first three child nodes (denoted in purple red and green). Each child node can be further divided into, smaller nodes sharing height and/or width. As can be observed in the example, all leaf nodes that share a parent node tend to be of the same class. The label smoothing energy favors such label configurations.

$$E_{ls} = \frac{1}{M} \sum_{n \in \Gamma} \frac{1}{\binom{C_n}{2}} \sum_{\substack{c_1, c_2 \in \text{child}(n) \\ c_1, c_2 \text{ are leaf nodes}}} \mathbb{I}\left[l(c_1) \neq l(c_2))\right] \tag{9}$$

where M is the total number of leaf clusters in an image tree Γ, C_n is the number of children of node n and $l(\cdot)$ is the class label of a child IE.

The Layout variance energy E_{lv} incites the structural modularity in the tree and penalizes shape and position deviations within a tree branch.

$$E_{lv} = \frac{1}{M} \sum_{n \in \Gamma} \sum_{c_i \in \text{child}(n)} [h(c_i) - h_m]^2 + [w(c_i) - w_m]^2 \tag{10}$$

where $h(\cdot)$ and $w(\cdot)$ determine the height and width of a child rectangle, h_m and w_m denote the average cluster width and height.

The State size energy E_s favors a higher number of IEs.

$$E_s = -\frac{K}{N} \tag{11}$$

where N is the number of rectangles in the proposal pool.

3.2.1 Reversible Jump MCMC Moves

The solution space of the energy function described above represents a high dimensional space. Optimization in this space can be achieved efficiently by sampling the Markov chain with simulated annealing, an MCMC based stochastic optimization method. The Markov chain with a stationary distribution is constructed such that the majority of the probability mass concentrates at the global minimum of the total energy. By sampling different states of the chain one can traverse through the multi-dimensional state space. From the current state S, the new state S^* is sampled with proposal probability $p(S^*|S)$. If $p(S^*)p(S^*|S) > p(S)p(S|S^*)$, then we accept the "better" state S^*. Otherwise, we accept the new state with a probability proportional to $p(S^*)^{\frac{1}{T_i}} p(S|S^*)/p(S)^{\frac{1}{T_i}} p(S^*|S)$ where $i \in \mathbb{N}$ is the current iteration and T_i is the temperature at step i. Effectively, the acceptance probability of the new state is:

$$a\left(S, S^*\right) = \min\left\{1, \frac{p(S^*)^{\frac{1}{T_i}} p(S|S^*)}{p(S)^{\frac{1}{T_i}} p(S^*|S)}\right\} \tag{12}$$

Pohlen *et al*. [26] performs a simulated annealing based optimization in multiple rounds, each time with a different, fixed dimension. Bounds on the dimension are estimated in a separate process, solely based on rectangle shapes, *i.e.* decoupled from appearance. The repeated optimization process is prone to errors in the bounds estimation and inefficient.

In contrast, we perform optimization using a trans-dimensional variant of MCMC, namely reversible jump MCMC [7]. The idea behind rjMCMC is to allow sampling the trans-dimensional space with a stationary distribution. To achieve a stationary distribution a careful balance of the chain dynamics must be fulfilled. In rjMCMC, this balance is obtained through dimension matching reversible jump moves. In our architecture, we design one jump move pair (*birth* and *death* move) to search over the variable dimensional space and one diffusion (*exchange*) move to explore a fixed dimensional space:

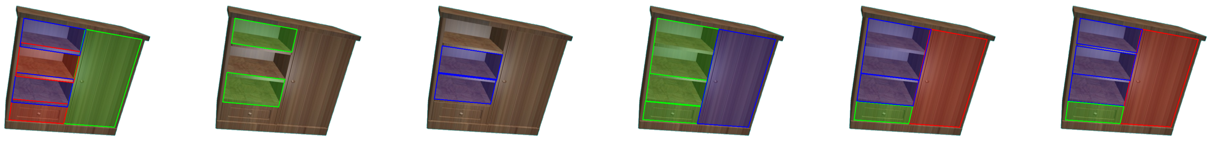
Birth move. The new state S^* is generated from S by adding a new rectangle r from the rectangle pool while keeping all the other rectangles fixed. The birth move increases the dimension of S by the dimension of the added rectangle $\dim(r)$. The acceptance probability for this move is:

$$a^B\left(S, S^*\right) = \min\left\{1, \theta^B(S, S^*)\right\} \tag{13}$$

where

$$\theta^B(S, S^*) = \underbrace{\frac{p(S^*)^{\frac{1}{T_i}}}{p(S)^{\frac{1}{T_i}}}}_{\substack{\text{posterior}\\\text{ratio}}} \cdot \underbrace{\frac{q_D(S, r \mid S^*) \cdot p(D)}{q_B(S^* \mid S, r) \cdot p(B)}}_{\substack{\text{proposal}\\\text{ratio}}} \cdot \underbrace{J_B}_{jacobian} \tag{14}$$

where, $J_B = \left|\frac{\partial(S^*)}{\partial(S,r)}\right|$ is a Jacobian for the transformation from (S, r) to S^*. $p(B)$ and $p(D)$ are the probabilities for choosing birth and death moves respectively, $q_B(S^* \mid S, r)$, is the probability to add rectangle r to the current state S, and similarly $q_D(S, r \mid S^*)$ defines the probability to delete a certain rectangle r from the current state S. $p(S)$ and



(a) cover (E_c) (b) overlap (E_o) (c) weight (E_w) (d) E_o + E_c (e) E_o + E_c + E_w (f) total

Figure 3: Effect of different energies on the proposal subset selection (from left to right): cover (E_c), overlap (E_o), rectangle weight (E_w), E_o + E_c, E_o + E_c + E_w, Total (E). We show the most important energy terms and their combination that contribute most of the segmentation performance during the proposal selection step.

$p(S^*)$ are determined from the energy E according to Equation 4. We assume uniform proposal probability for selecting the rectangle, hence the proposal ratio only depends on the number of rectangles in the current state (k) and in the proposal pool ($N-k$). The probabilities for death and birth moves are set so that the overall acceptance rate is high. The Jacobian is derived to be 1 (Please refer to supplementary for the derivation). Equation 14 simplifies to:

$$\theta^B(S,S^*) = \frac{p(S^*)^{\left(\frac{1}{T_i}\right)}}{p(S)^{\left(\frac{1}{T_i}\right)}} \cdot \frac{N-k}{k} \cdot \frac{p(D)}{p(B)} \tag{15}$$

Death move. This move is the reverse of a birth move. It removes one rectangle while keeping all the other rectangles fix. Death and birth moves are a reversible move pair, ensuring balance in the chain. The acceptance probability of a death move can be given as:

$$a^D\left(S,S^*\right) = \min\left\{1, \theta^D(S,S^*)\right\} \tag{16}$$

where

$$\begin{aligned}\theta^D(S,S^*) &= \left(\theta^B(S,S^*)\right)^{-1}\\ &= \frac{p(S^*)^{\left(\frac{1}{T_i}\right)}}{p(S)^{\left(\frac{1}{T_i}\right)}} \cdot \frac{k}{N-k} \cdot \frac{p(B)}{p(D)}\end{aligned} \tag{17}$$

Another popular reversible jump move pair is the split and the merge move. As these moves are computationally expensive, we avoid using them during optimization. Instead, the split/merge augmentation as described in Section 3.1.1 serve as a proxy.

Exchange move. Is a diffusion move which preserves dimensions. A rectangle is randomly selected from the current state and is exchanged with another rectangle which is randomly sampled from the proposal pool:

$$a^E\left(S,S^*\right) = \min\left\{1, \theta^E(S,S^*)\right\} \tag{18}$$

where

$$\theta^E(S,S^*) = \frac{p(S^*)^{\left(\frac{1}{T_i}\right)}}{p(S)^{\left(\frac{1}{T_i}\right)}} \tag{19}$$

4. Evaluation

To the best of our knowledge, we are the first to present a furniture dataset including both, RGB and depth. For the evaluation of our proposed method we introduce a new synthetic dataset consisting of 160 images with a resolution of 640×480. The ground truth structures and labels are annotated manually. In our experiments we perform a 4-fold cross validation. In each round, 75% of the images are used to train the appearance codebook and shape priors and 25% images are used for testing. We are only aware of the work of Pohlen *et al.* [26] which tackles this problem of furniture segmentation and therefore serves as a baseline for comparison in the experiments. We generate our dataset by modifying readily available 3D furniture models in blender. All the given models are oriented in aesthetically beautiful orientation and lighting condition. We change the orientation, texture and lighting condition of these models. Additionally, we add artificial axial and lateral Kinect noise depending on depth and orientation as described in [23].

4.1. Quantitative Results

In this section we evaluate various aspects of our pipeline and present the overall quantitative results.

Rectangle Set Augmentation. During the generation of our over-complete set of IE proposals, we use the previously suggested proposal generation by Pohlen *et al.* [26] and then extend the resulting proposal set by our rectangle set augmentation. We evaluate the maximum achievable recall at this stage of the pipeline and the resulting overall improvement. The initial proposal generation step achieves a recall of 79.8%, our augmentation improves this recall to 83.3%. For our full pipeline we observed an improvement of 1% point by adding IE augmentation.

Structural Inference. Compared to [26], in our method we remove the costly rectangle pruning step and effectively add a corresponding pruning criteria to the objective function, Equations 5, 6, 7. This improves the overall speed considerably (1.9x faster) and avoids hard decisions before optimization, leading to better recall and hence overall segmentation. We report precision, recall and F_1 measure of

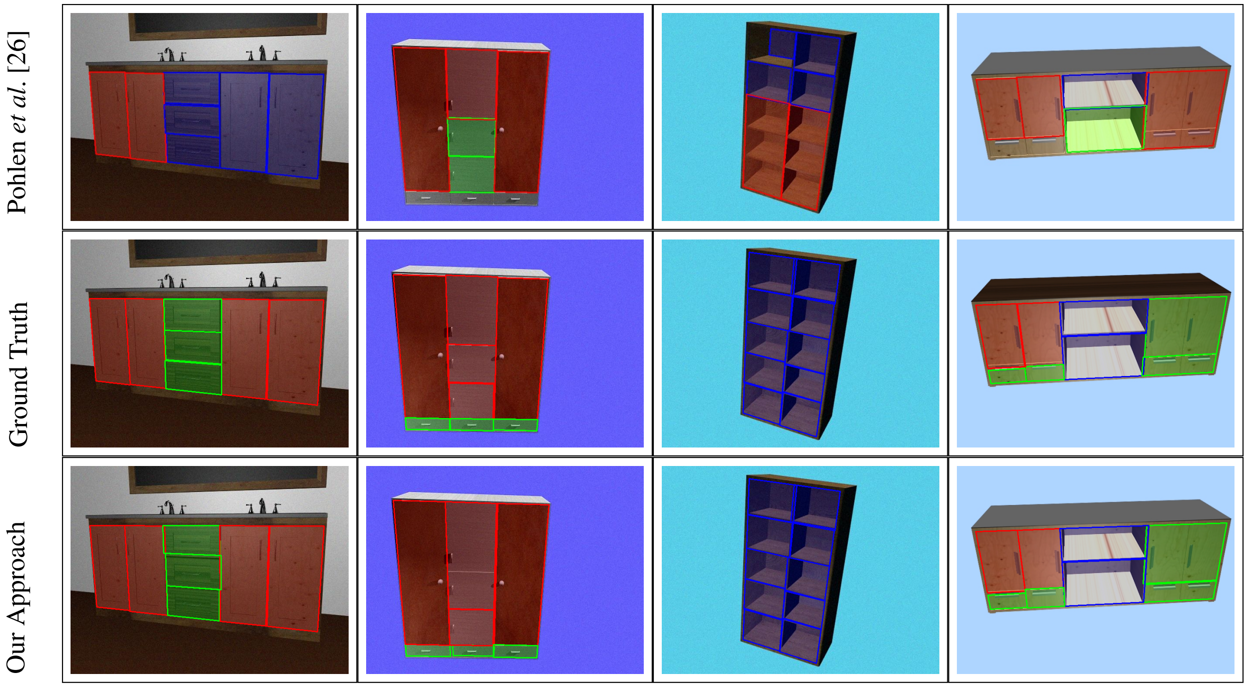


Figure 4: Qualitative comparison of segmentation on 3D synthetic data: The first row is the input RGB images. Second row is ground truth annotation. Third row shows result of segmentation by [26]. The last row shows our segmentation result using depth. (door, drawer and shelf).

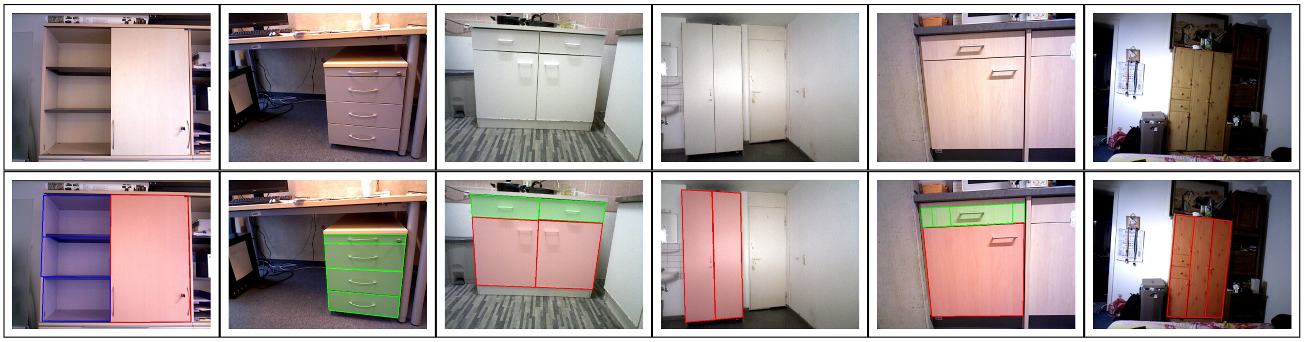


Figure 5: Qualitative results: The input real Kinect images and corresponding segmentation are displayed respectively on row 1 and row 2. Columns 1-5 shows the success cases while columns 6 displays the failed cases. The two main reasons for failure are missed edges and high amount of texture. (door, drawer and shelf).

our structure inference. At this stage, we are only interested in the subdivision of the furniture, not in the resulting semantic labeling. For comparison, Table 1 serves the 2D and 3D versions of our method and the work of [26]. Table 1 shows that using 3D improves our method by a large margin, yet even our 2D version sets a new state-of-the-art.

Class Label Inference. Here we measure the accuracy of the predicted labels only for the correctly detected IEs. We consider a IE detected if the IoU with a ground truth annotation exceeds 65%. This allows us to measure the efficiency of our appearance model independent of the structural subdivision. Table 2 reports accuracy of class label prediction

Table 1: Comparison of overall structure inference performance. Here we compare our method with and without depth to the approach presented in [26].

	precision	recall	F_1
[26]	68.8%	49.8%	57.8%
our (2D)	63.5%	68.7%	66.0%
our (3D)	**73.5%**	**79.9%**	**76.6%**

for [26] and our approach with and without using depth.

Table 2: Class label accuracy for correctly detected IEs.

	door	drawer	shelf
[26]	91.9%	71.7%	15.5%
our (2D)	76.4%	77.6%	40.9%
our (3D)	**99.3%**	**96.2%**	**98.8%**

It is apparent that depth is a crucial cue to approach perfect class label inference. We show the full confusion matrices for the overall segmentation for both, without and with depth (see Table 3). Using only 2D information leads to a high confusion between "drawer" and "shelf", which can be resolved using depth.

Table 3: Detailed class label performance of our approach with and without using depth.

		Prediction 2D		
		Door	Drawer	Shelf
Truth	Door	75.4%	1.5%	23.1%
	Drawer	3.7%	77.0%	19.3%
	Shelf	12.9%	44.4%	42.7%

		Prediction 3D		
		Door	Drawer	Shelf
Truth	Door	99.3%	0.7%	0%
	Drawer	3.0%	96.2%	0.8%
	Shelf	1.1%	1.1%	98.8%

Segmentation performance. To compare the overall segmentation performance, we combine the structure and labeling accuracy of the algorithm. We multiply the structure accuracy with the label accuracy for each label and take the average. The combined performance of the baseline from Pohlen *et al.* [26] reaches 33.8%, compared to our method (2D) reaching 45.5%. When enabling depth the final result of our full pipeline is 78.3%.

Contribution of Energy Terms. We examine how each of the energy terms affect the structure and label prediction. We perform segmentation using different combinations of energy terms in Figure 6. As expected, the rectangle weight energy (E_w) is crucial for label accuracy, as is the cover energy (E_c) for predicting structure. While the structure prediction reaches competitive results using only single energy terms, the label accuracy highly benefits of energy combinations. The full energy term results in the best overall performance.

4.2. Qualitative Results

Figure 4 shows an overall qualitative comparison between [26] (2D) and our approach (3D). Besides using only our proposed synthetic dataset, we also performed a qualitative study on real world image samples originating from the Kinect sensor. For this experiment, we train the appearance codebook and shape prior on the entire synthetic dataset (160 images). Figure 5 shows example segmentation result.

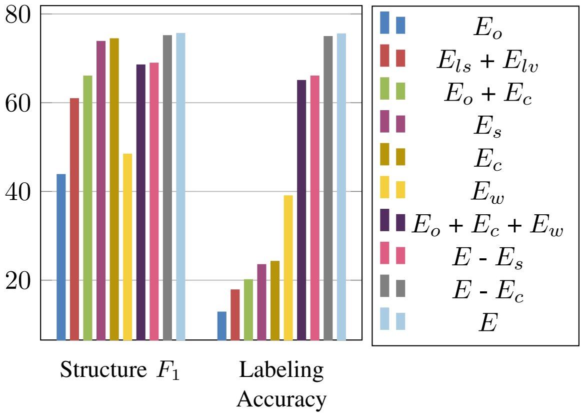


Figure 6: Bar graph showing performance on structure prediction (Left) and label accuracy (right) with different combinations of the energy terms. Best result is achieved when all the energy terms are incorporated.

Contribution of Energy Terms. Figure 3 qualitatively shows the influence of each energy term given a single example.

5. Conclusion

We propose a method for semantic segmentation of furniture into their interaction elements for RGB-D images. We show that depth information is crucial to the structural inference and classification of the IEs. We propose a multi-objective optimization method using an effective energy maximization formulation. We successfully demonstrate the strength of our rjMCMC optimization design for our trans-dimensional model space. Finally, we show considerable improvement on the previous state-of-the-art results for furniture parsing given on novel 3D furniture dataset. This work is also transferable to real Kinect images, opening doors for the advance research in robotics for interaction with furniture. Our code[1] and annotated dataset[2] are publicly available.

Acknowlegment. The work in this paper was funded by the EU project STRANDS (ICT-2011-600623).

[1] www.vision.rwth-aachen.de/publications/

[2] www.vision.rwth-aachen.de/furniture

References

[1] A. Barbu and S. C. Zhu. Graph partition by swendsen-wang cuts. In *ICCV*, 2003.

[2] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *PAMI*, 24(5):603–619, 2002.

[3] P. Dollár and C. Zitnick. Structured forests for fast edge detection. In *ICCV*, 2013.

[4] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, pages 167–181, 2004.

[5] B. Fulkerson, A. Vedaldi, and S. Soatto. Class segmentation and object localization with superpixel neighborhoods. In *ICCV*.

[6] R. Girshick. Fast r-cnn. In *ICCV*, December 2015.

[7] P. J. Green. Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, 82:711–732, 1995.

[8] S. Gupta, P. Arbelaez, and J. Malik. Perceptual organization and recognition of indoor scenes from RGB-D images. In *CVPR*, 2013.

[9] F. Han and S. Zhu. Bottom-up/top-down image parsing by attribute graph grammar. In *ICCV*, 2005.

[10] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. In *CVPR*, 2015.

[11] V. Hedau. Recovering free space of indoor scenes from a single image. In *CVPR*, 2012.

[12] V. Hedau, D. Hoiem, and D. Forsyth. Thinking inside the box: Using appearance models and context based on room geometry. In *ECCV*, 2010.

[13] A. Kanezaki and T. Harada. 3d selective search for obtaining object candidates. In *IROS*, 2015.

[14] M. Kass., A. P. Witkin, and D. Terzopoulos. Snakes: Active contour models. *IJCV*, 1(4):321–331, 1988.

[15] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? In *ECCV*, 2002.

[16] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 2001.

[17] J. J. Lim, H. Pirsiavash, and A. Torralba. Parsing IKEA objects: Fine pose estimation. In *ICCV*, 2013.

[18] A. Martinovic and L. Van Gool. Bayesian grammar learning for inverse procedural modeling. In *CVPR*, 2013.

[19] M. Mathias, A. Martinovic, and L. V. Gool. ATLAS: A three-layered approach to facade parsing. *IJCV*, 118(1):22–48, 2016.

[20] M. Mathias, A. Martinovic, J. Weissenberg, and L. J. V. Gool. Procedural 3d building reconstruction using shape grammars and detectors. In *3DIMPVT*, 2011.

[21] M. Mathias, A. Martinović, J. Weissenberg, S. Haegler, and L. Van Gool. Automatic architectural style recognition. In F. Remondino and S. El-Hakim, editors, *Proceedings of the 4th ISPRS International Workshop 3D-ARCH 2011*. ISPRS, 2011.

[22] P. Müller, G. Zeng, P. Wonka, and L. Van Gool. Image-based procedural modeling of facades. In *SIGGRAPH*, 2007.

[23] C. V. Nguyen, S. Izadi, and D. Lovell. Modeling kinect sensor noise for improved 3d reconstruction and tracking. In *3DIMPVT*, 2012.

[24] S. Osher and N. Paragios. *Geometric Level Set Methods in Imaging,Vision,and Graphics*. Springer, 2003.

[25] B. Pepik, P. V. Gehler, M. Stark, and B. Schiele. 3d2pm - 3d deformable part models. In *ECCV*, 2012.

[26] T. Pohlen, I. Badami, M. Mathias, and B. Leibe. Semantic segmentation of modular furniture. In *WACV*, 2016.

[27] H. Riemenschneider, S. Sternig, M. Donoser, P. M. Roth, and H. Bischof. Hough regions for joining instance localization and segmentation. In *ECCV*, 2012.

[28] N. Ripperda and C. Brenner. Reconstruction of faade structures using a formal grammar and rjmcmc. In *DAGM-Symposium*, 2006.

[29] J. Shi and J. Malik. Normalized cuts and image segmentation. *PAMI*, 22(8):888–905, 2000.

[30] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012.

[31] K. Smith, S. O. Ba, J.-M. Odobez, and D. Gatica-Perez. Tracking the visual focus of attention for a varying number of wandering people. *PAMI*, 30(7):1212–1229, 2008.

[32] G. Stiny. *Pictorial and formal aspects of shape and shape grammars*. 1975.

[33] M. Sun, S. S. Kumar, G. Bradski, and S. Savarese. Object detection, shape recovery, and 3d modelling by depth-encoded hough voting. *Computer Vision Image Understanding*, 117(9):1190–1202, 2013.

[34] O. Teboul, I. Kokkinos, L. Simon, P. Koutsourakis, and N. Paragios. Parsing facades with shape grammars and reinforcement learning. *PAMI*, 35(7):1744–1756, 2013.

[35] Z. Tu and S. C. Zhu. Image segmentation by data-driven markov chain monte carlo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(5):657–673, 2002.

[36] C. A. Vanegas, D. G. Aliaga, and B. Benes. Building reconstruction using manhattan-world grammars. In *CVPR*, 2010.

[37] J. Zhang, C. Kan, A. G. Schwing, and R. Urtasun. Estimating the 3d layout of indoor scenes and its clutter from depth sensors. In *ICCV*, 2013.

[38] Y. Zhao and S. Zhu. Scene parsing by integrating function, geometry and appearance models. In *CVPR*, 2013.

[39] Y. Zhao and S. C. Zhu. Image parsing with stochastic scene grammar. In *NIPS*, 2011.

[40] B. Zheng, Y. Zhao, J. C. Yu, K. Ikeuchi, and S. Zhu. Beyond point clouds: Scene understanding by reasoning geometry and physics. In *CVPR*, 2013.

[41] B. Zheng, Y. Zhao, J. C. Yu, K. Ikeuchi, and S. Zhu. Scene understanding by reasoning stability and safety. *IJCV*, 112(2):221–238, 2015.