# Joint 3D Object and Layout Inference from a single RGB-D Image

Andreas Geiger[1] Chaohui Wang[1,2]

[1]Max Planck Institute for Intelligent Systems, Tübingen, Germany
[2]Université Paris-Est, Marne-la-Vallée, Paris, France

**Abstract.** Inferring 3D objects and the layout of indoor scenes from a single RGB-D image captured with a Kinect camera is a challenging task. Towards this goal, we propose a high-order graphical model and jointly reason about the layout, objects and superpixels in the image. In contrast to existing holistic approaches, our model leverages detailed 3D geometry using inverse graphics and explicitly enforces occlusion and visibility constraints for respecting scene properties and projective geometry. We cast the task as MAP inference in a factor graph and solve it efficiently using message passing. We evaluate our method with respect to several baselines on the challenging NYUv2 indoor dataset using 21 object categories. Our experiments demonstrate that the proposed method is able to infer scenes with a large degree of clutter and occlusions.

## 1 Introduction

Robotic systems (e.g., household robots) require robust visual perception in order to locate objects, avoid obstacles and reach their goals. While much progress has been made since the pioneering attempts in the early 60's [33], 3D scene understanding remains a fundamental challenge in computer vision. In this paper, we propose a novel model for holistic 3D understanding of indoor scenes (Fig. 1). While existing approaches to the 3D scene understanding problem typically infer only objects [16, 17] or consider layout estimation as a pre-processing step [25], our method reasons jointly about 3D objects and the scene layout. We explicitly model visibility and occlusion constraints by exploiting the expressive power of high-order graphical models. This ensures a physically plausible interpretation of the scene and avoids undercounting and overcounting of image evidence.

Following [17,25,38], our approach also relies on a set of 3D object proposals and pursues model selection by discrete MAP inference. However, in contrast to previous works, we do not fit cuboids to 3D segments in a greedy fashion. Instead, we propose objects and layout elements by solving a set of "inverse graphics" problems directly based on the unary potentials in our model. This allows us to take advantage of the increasing availability of 3D CAD models and leads to more accurate geometric interpretations. We evaluate the proposed method in terms of 3D object detection performance on the challenging NYUv2

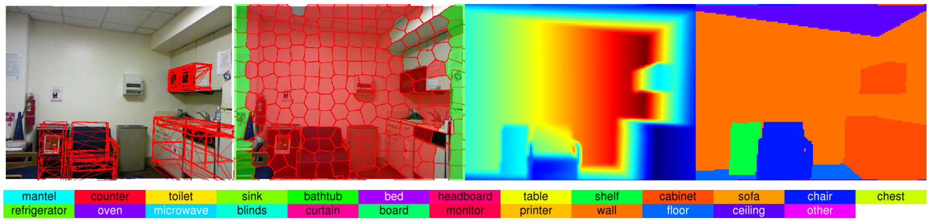

**Fig. 1. Illustration of our Results.** Left-to-right: Inferred objects, superpixels (red=explained), reconstruction (blue=close to red=far) and semantics with color code.

dataset [38] and compare it to [25] as well as two simple baselines derived from our model. Our code and dataset are publicly available[1].

## 2 Related Work

3D indoor scene understanding is a fundamental problem in computer vison and has recently witnessed great progress enabled by the increasing performance of semantic segmentation and object detection algorithms [6, 10] as well as the availability of RGB-D sensors. Important aspects of this problem include 3D layout estimation [15, 36], object detection [35, 39], as well as semantic segmentation [13, 32]. A variety of geometric representations have been proposed, including cuboids [17, 25, 46], 3D volumetric primitives [8, 47], as well as CAD models [1,24,35,39]. While the problem has traditionally been approached using RGB images [1, 8, 23, 36, 46] and videos [42], the availability of RGB-D sensors [30] and datasets [38] nourish the hope for more accurate models of the scene [11, 16, 18, 47]. Towards this goal, a number of holistic models have been proposed which take into account the relationship between objects (often represented as cuboids) and/or layout elements in the scene [4,22,37,45]. While CRFs provide a principled way to encode such contextual interactions [43], modeling visibility/occlusion rigorously is a very challenging problem [37, 41].

The approach that we present is particularly related to several recent works which model the 3D scene using geometric primitives (e.g., cuboids) [17,25]. Despite their promising performance, these works ignore some important aspects in their formulation. In [25], a pairwise graphical model is employed to incorporate contextual information, but visibility constraints are ignored, which leads to overcounting of image evidence. In [17], undercounting of image evidence is addressed by enforcing "explained" superpixels to be associated with at least one object. However, occlusions are not considered (e.g., an object which explains a superpixel might be occluded by another object at the same superpixel), which can lead to implausible scene configurations. Besides, semantic labels and related contextual information are ignored.

While 3D CAD models have been primarily used for object detection [24, 35, 39, 48], holistic 3D scene understanding approaches typically rely on simpler

[1] http://www.cvlibs.net/projects/indoor_scenes/

cuboid models [17, 25]. In this work, we leverage the precise geometry of CAD models for holistic 3D scene understanding. The advantages are two-fold: First, we can better explain the depth image evidence. Second, it allows for incorporating visibility and occlusion constraints in a principled fashion.

## 3 Joint 3D Object and Layout Inference

We represent indoor scenes by a set of layout elements (e.g., "wall", "floor", "ceiling") and objects (e.g., "chairs", "shelves", "cabinets"). Given an RGB-D image $\mathbf{I}$ partitioned into superpixels $\mathcal{S}$, our goal is to simultaneously infer all layout and object elements in the scene. In particular, we reason about the type, semantic class, 3D pose and 3D shape of each object and layout element. Towards this goal, we first generate a number of object and layout proposals given the observed image $\mathbf{I}$ (see Section 3.4), and then select a subset of layout elements and objects which best explain $\mathbf{I}$ and $\mathcal{S}$ via MAP inference in a CRF.

More formally, let $\mathcal{L}$ and $\mathcal{O}$ denote the set of layout and object proposals, respectively. Each proposal $\rho_i = (t_i, c_i, m_i, r_i, z_i)$ $(i \in \mathcal{L} \cup \mathcal{O})$ comprises the following attributes: the proposal type $t_i \in \{layout, object\}$, its semantic class $c_i \in \{mantel, \ldots, other\}$, a 3D object model indexed by $m_i \in \{1, \ldots, M\}$, the image region $r_i \subset \mathbf{I}$ which has generated the proposal, as well as a set of pose parameters $z_i$ which characterize pose and scale in 3D space. For each proposal, the semantic class variable $c_i$ takes a label from the set of classes corresponding to its type $t_i \in \{layout, object\}$. We pre-aligned the scene with the camera coordinate axis using the method of Silberman et al. [38] and assume that layout elements extend to infinity. Thus, for $t_i = layout$, $m_i$ indexes a 3D plane model, and $z_i$ comprises the normal direction and the signed distance from the camera center. For $t_i = object$, $m_i$ indexes one of the 3D CAD models in our dataset or a 3D cuboid if no CAD model is available for an object category. Furthermore, $z_i$ comprises the 3D pose (we only consider rotations around the up-vector). and scale parameters of the object, i.e., $z_i \in \mathbb{R}^3 \times [-\pi, +\pi) \times \mathbb{R}^3_+$.

We associate a binary random variable $X_i \in \{0, 1\}$ with each layout/object proposal $\rho_i$, taking 1 if scene element $i$ is present and 0 otherwise. To impose visibility/occlusion constraints and avoid evidence undercounting, we also associate a binary random variable $X_k$ $(k \in \mathcal{S})$ with each superpixel $k$ to model if the superpixel is explained $(X_k = 1)$ or unexplained $(X_k = 0)$. A valid scene configuration should explain as many superpixels as possible while at the same time satisfying Occam's razor, i.e., simple explanations with a small number of layouts and objects should be preferred. We specify our CRF model on $\mathbf{X} = \{X_i\}_{i \in \{\mathcal{L} \cup \mathcal{O} \cup \mathcal{S}\}}$ in terms of the following energy

$$\begin{aligned} E(\mathbf{x}|\mathbf{I}) = \sum_{i \in \mathcal{L}} \underbrace{\phi_i^{\mathcal{L}}(x_i|\mathbf{I})}_{\text{layout}} + \sum_{i \in \mathcal{O}} \underbrace{\phi_i^{\mathcal{O}}(x_i|\mathbf{I})}_{\text{object}} + \sum_{k \in \mathcal{S}} \underbrace{\phi_k^{\mathcal{S}}(x_k)}_{\text{superpixel}} + \sum_{i \in \mathcal{L} \cup \mathcal{O}, k \in \mathcal{S}} \underbrace{\psi_{ik}^{\mathcal{S}}(x_i, x_k|\mathbf{I})}_{\text{occlusion/visibility}} \\ + \sum_{k \in \mathcal{S}} \underbrace{\kappa_k(\mathbf{x}_{c_k})}_{\text{occlusion/visibility}} + \sum_{i,j \in \mathcal{O}} \underbrace{\psi_{ij}^{\mathcal{O},\mathcal{O}}(x_i, x_j)}_{\text{object-object}} + \sum_{i \in \mathcal{L}, j \in \mathcal{O}} \underbrace{\psi_{ij}^{\mathcal{L},\mathcal{O}}(x_i, x_j)}_{\text{layout-object}} \end{aligned} \quad (1)$$

where $\mathbf{x}_{c_k} = (x_i)_{i \in c_k}$ denotes a joint configuration of all variables involved in clique $c_k$. The unary potentials $\phi_i^{\mathcal{L}}$ and $\phi_i^{\mathcal{O}}$ encode the agreement of proposal $i$ with the image, and $\phi_k^{\mathcal{S}}$ adds a penalty to the energy function if superpixel $k$ is not explained by any object or layout element. The pairwise potentials $\psi_{ik}^{\mathcal{S}}$ and the high-order potentials $\kappa_c$ ensure consistency between the scene and superpixels while respecting visibility and occlusion constraints. Contextual information such as relative pose or scale is encoded in $\psi_{ij}^{\mathcal{O},\mathcal{O}}$ and $\psi_{ij}^{\mathcal{L},\mathcal{O}}$.

### 3.1 Unary Potentials

We assume that each proposal $\rho_i$ originates from a candidate image region $r_i \subset \mathbf{I}$ which we use to define the layout and object unary potentials in the following. Details on how we obtain these proposal regions will be specified in Section 3.4.

**Layout Unary Potentials:** We model the layout unary terms as

$$\phi_i^{\mathcal{L}}(x_i|\mathbf{I}) = w^{\mathcal{L}} \left( h^{\mathcal{L}}(\rho_i) + b^{\mathcal{L}} \right) x_i \tag{2}$$

where $w^{\mathcal{L}}$ and $b^{\mathcal{L}}$ are model parameters that adjust the importance and bias of this term and $h^{\mathcal{L}}(\rho_i)$ captures how well the layout proposal fits the RGB-D image. More specifically, we favour layout elements which agree with the depth image and occlude as little pixels as possible, i.e., we assume that the walls, floor and ceiling determine the boundaries of the scene. In particular, we define $h^{\mathcal{L}}(\rho_i)$ as the difference between the count of pixels occluded by proposal $\rho_i$ and the number of depth inliers wrt. all pixels in region $r_i$.

**Object Unary Potentials:** Similarly, we define the object unary terms as

$$\phi_i^{\mathcal{O}}(x_i|\mathbf{I}) = w^{\mathcal{O}} \left( h^{\mathcal{O}}(\rho_i) + b^{\mathcal{O}} \right) x_i \tag{3}$$

where $h^{\mathcal{O}}(\rho_i)$ captures how well the object fits the RGB-D image: We consider an object as likely if its scale (last 3 dimensions of $z_i$) agrees with the scale of the 3D object model $s_i$, its rendered depth map agrees with the RGB-D depth image and its re-projection yields a region that maximizes the overlap with the region $r_i$ which has generated the proposal. We assume a log-normal prior for the scale $s_i$, which we learn from all instances of class $c_i$ in the training data.

**Superpixel Unary Potentials:** For each superpixel $k$ we define

$$\phi_k^{\mathcal{S}}(x_k) = w^{\mathcal{S}}(1 - x_k) \tag{4}$$

where $w^{\mathcal{S}} \geq 0$ is a penalty assigned to each superpixel $k$ which is not explained. This term encourages the explanation of as many superpixels as possible. Note that without such a term, we would obtain the trivial solution where none of the proposals is selected. Due to the noise in the input data and the approximations in the geometry model we enforce this condition as a soft constraint, i.e., superpixels may remain unexplained at cost $w^{\mathcal{S}}$, cf. Fig. 1.

### 3.2 Visibility and Occlusion Potentials

To ensure that the selected scene elements and superpixels satisfy visibility and occlusion constraints we introduce the potentials $\kappa_k$ and $\psi_{ik}^{\mathcal{S}}$.
**High-Order Consistency Potentials:** $\kappa_k(\mathbf{x}_{c_k})$ is defined as:

$$\kappa_k(\mathbf{x}_{c_k}) = \begin{cases} \infty & \text{if } x_k = 1 \wedge \sum_{i \in \mathcal{L} \cup \mathcal{O}} x_i = 0 \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Here, the clique $c_k \subseteq \{k\} \cup \mathcal{L} \cup \mathcal{O}$ comprises the superpixel $k$ and all proposals $i \in \mathcal{L} \cup \mathcal{O}$ that are able to explain superpixel $k$. In practice, we consider a superpixel as explained by a proposal if its rendered depth map is within a threshold (in our case 0.2 m) of $\mathbf{I}$ for more than 50% of the comprised pixels. Note that Eq. 5 ensures that only superpixels which are explained by at least one object can take label $x_k = 1$.
**Occlusion Potentials:** Considering $\kappa_k(\mathbf{x}_{c_k})$ alone will lead to configurations where a superpixel is explained by objects which are themselves occluded by other objects at the same superpixel, thus violating visibility. To prevent this situation, we introduce pairwise occlusion potentials $\psi_{ik}^{\mathcal{S}}$ between all scene elements $i \in \mathcal{L} \cup \mathcal{O}$ and superpixels $k \in \mathcal{S}$

$$\psi_{ik}^{\mathcal{S}}(x_i, x_k | \mathbf{I}) = \begin{cases} \infty & \text{if } x_i = 1 \wedge x_k = 1 \wedge \text{“}i \text{ occludes } k\text{”} \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where “$i$ occludes $k$” is true if for more than 50% of the pixels in superpixel $k$ the depth of the rendered object $i$ is at least 0.2m smaller than the corresponding depth value in $\mathbf{I}$. In other words, we prohibit superpixels from being explained if one or more active scene elements occlude the view.

### 3.3 Context Potentials

We also investigate contextual cues in the form of pairwise relationships between object and layout elements as described in the following.
**Object-Object Potentials:** The pairwise potential between object $i$ and $j$ is modeled as the weighted sum

$$\psi_{ij}^{\mathcal{O},\mathcal{O}}(x_i, x_j) = \sum_{\mathrm{t} \in \{\mathrm{p},\mathrm{s},\mathrm{ovlp}\}} w^{\mathrm{t}} \psi_{ij}^{\mathrm{t}}(x_i, x_j) \tag{7}$$

where $\psi_{ij}^{\mathrm{t}}$ is a feature capturing the relative pose, scale or overlap between object $i$ and object $j$. We encode the pose and scale correlation between objects conditioned on the pair of semantic classes. For the pose, let $\mathrm{dist}_{ij}(z_i, z_j)$ and $\mathrm{rot}_{ij}(z_i, z_j)$ denote the distance and the relative rotation (encoded as cosine similarity) between object $i$ and $j$, respectively. For each pair $(c, c')$ of semantic classes, we estimate the joint distribution $p_{c,c'}^{\mathrm{p}}(\mathrm{dist}_{ij}, \mathrm{rot}_{ij})$ from training data using kernel density estimation (KDE). The relative pose potentials between a pair of objects are then defined by the negative log-likelihood $\psi_{ij}^{\mathrm{p}}(x_i, x_j) =$

$-\,x_i\,x_j\,\log p^{\mathrm{p}}_{c_i,c_j}(\mathrm{dist}_{ij}(z_i,z_j),\mathrm{rot}_{ij}(z_i,z_j))$. Similarly, we consider scale by the negative logarithm of the relative scale distribution between semantic classes $c_i$ and $c_j$ as $\psi^{\mathrm{s}}_{ij}(x_i,x_j) = -\,x_i\,x_j\,\log p^{\mathrm{s}}_{c_i,c_j}(\mathrm{s}_{ij})$. Here, the relative scale $\mathrm{s}_{ij}$ is defined as the difference of the logarithm in scale and $p^{\mathrm{s}}_{c_i,c_j}(s_{ij})$ is learned from training data using KDE. To avoid objects intersecting each other, we further penalize the overlapping volume of two objects

$$\psi^{\mathrm{ovlp}}_{ij}(x_i,x_j) = x_i\,x_j\left(\frac{V(\rho_i)\cap V(\rho_j)}{V(\rho_i)} + \frac{V(\rho_i)\cap V(\rho_j)}{V(\rho_j)}\right)$$

where $V(\rho)$ denotes the space occupied by the 3D bounding box of proposal $\rho$.
**Layout-Object Potentials:** Regarding the pairwise potential between layout $i$ and object $j$, we consider the relative pose and volume exclusion constraints in analogy to those for the object-object potentials specified above:

$$\psi^{\mathcal{L},\mathcal{O}}_{ij}(x_i,x_j) = w^{\mathrm{p}}\psi^{\mathrm{p}}_{ij}(x_i,x_j) + w^{\mathrm{ovlp}}\psi^{\mathrm{ovlp}}_{ij}(x_i,x_j) \tag{8}$$

Here, $\psi^{\mathrm{p}}_{ij}$ denotes the log-likelihood of the object-to-plane distance and $\psi^{\mathrm{ovlp}}_{ij}$ penalizes the truncation of an object volume by a scene layout element.

### 3.4 Layout and Object Proposals

As discussed in the previous sections, our discrete CRF takes as input a set of layout and object proposals $\{\rho_i\}$. We obtain these proposals by first generating a set of foreground candidate regions $\{r_i\}$ using [3, 12] and then solving the "inverse graphics problem" by drawing samples from the unary distributions specified in Eq. 2 and Eq. 3 for each candidate region $r_i$.
**Foreground Candidate Regions:** For generating foreground candidate regions, we leverage the CPMC framework [3] extended to RGB-D images [25]. Furthermore, we use the output of the semantic segmentation algorithm of [12] as additional candidate regions. While [3] only provides object regions, [12] additionally provides information about the background classes *wall*, *floor* and *ceiling*. In contrast to existing works on RGB-D scene understanding which often rely on simple 3D cuboid representations [17, 25], we explicitly represent the shape of objects using 3D models. For indoor objects such data becomes increasingly available, e.g., searching for "chair", "sofa" or "cabinet" in Google's 3D Warehouse returns more than $10,000$ hits per keyword. In our case, we make use of a compact set of 66 models to represent object classes with non-cuboid shapes.
**Proposals from Unary Distributions:** Unlike [17, 25], we do not fit the tightest 3D cuboid to each candidate region for estimating the proposal's pose parameters as this leads to an undesirable shrinking bias. Instead, we sample proposals directly from the unary distributions specified in Section 3.1 using Metropolis-Hastings [9, 26], leveraging the power of our 3D models in a generative manner. More specifically, for each layout candidate region, we draw samples from $p_{\mathcal{L}}(z_i,m_i) \propto \exp\left(-\phi^{\mathcal{L}}(z_i,m_i|\mathbf{I})\right)$ and for each object candidate region we draw samples from $p_{\mathcal{O}}(z_i,m_i) \propto \exp\left(-\phi^{\mathcal{O}}(z_i,m_i|\mathbf{I})\right)$. Here, the potentials $\phi^{\mathcal{L}}$

and $\phi^{\mathcal{O}}$ are defined as the right hand sides of Eq. 2 and Eq. 3, fixing $x_i = 1$. Note that for proposal generation $\phi^{\mathcal{L}}$ and $\phi^{\mathcal{O}}$ depend on the pose and model parameters while those arguments are fixed during subsequent CRF inference. By restricting $z_i$ to rotations around the up-axis we obtain an 8-dimensional sampling space for objects. For layout elements the only unknowns are the normal direction and the signed distances from the camera coordinate origin.

We randomly choose between global and local moves. Our global moves sample new pose parameters directly from the respective prior distributions which we have learned from annotated objects in the NYUv2 training set [11]. Modes of the target distribution are explored by local Student's t distributed moves which slightly modify the pose, scale and shape parameters. For each candidate region $r_i$ we draw $10,000$ samples using the OpenGL-based 3D rendering engine `librender` presented in [14] and select the 3 most dominant modes.

### 3.5 Inference

Despite the great promise of high-order discrete CRFs for solving computer vision problems [2,43], MAP inference in such models remains very challenging. Existing work either aims at accelerating message passing for special types of potentials [7,20,27,31,40] or exploits sparsity of the factors [19,21,34]. Here, we explore the sparsity in our high-order potential functions (cf., Eq. 5) and recursively split the state space into sets depending on whether they do or do not contain any special state as detailed in the supplementary material. The class of sparse high-order potentials which can be handled by our recursive space-partitioning is a generalization of the pattern-based potentials proposed in [21,34]. In contrast to [21, 34], our algorithm does not make the common assumption that energy values corresponding to "pattern" states are lower than those assigned to all other states as this assumption is violated by the high-order potential in Eq. 5. For algorithmic details, we refer the reader to the supplementary material.

## 4 Experimental Results

We evaluate our method in terms of 3D object detection performance on the challenging NYUv2 RGB-D dataset [38] which comprises 795 training and 654 test images of indoor scenes including semantic annotations. For evaluation, we use the 25 object and layout (super-)categories illustrated in Fig. 1 and leverage the manually annotated 3D object ground truth of [11]. We extract 400 superpixels from each RGB-D image using the StereoSLIC algorithm [44], adapted to RGB-D information and generate about about 100 object proposals per scene. The parameters in our model ($w^{\mathcal{L}} = 1$, $b^{\mathcal{L}} = 0$, $w^{\mathcal{O}} = 1.45$, $b^{\mathcal{O}} = 1.3$, $w^{\mathcal{S}} = 1.3$, $w^{\text{p}} = w^{\text{s}} = 0.001$, and $w^{\text{ovlp}} = 100$) are obtained by coordinate descent on the NYUv2 training set and kept fixed during all our experiments.

**Evaluation Criterion:** We evaluate 3D object detection performance by computing the F1 measure for each object class and taking the average over all

| | mantel | counter | toilet | sink | bathtub | bed | headboard | table | shelf | cabinet | sofa | chair | chest | refrigerator | oven | microwave | blinds | curtain | board | monitor | printer | overall |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| #obj | 10 | 126 | 30 | 36 | 25 | 169 | 23 | 455 | 242 | 534 | 228 | 703 | 137 | 42 | 29 | 40 | 111 | 91 | 50 | 81 | 25 | 3187 |
| [25] - 8 Proposals | 0 | 4 | 27 | **12** | 0 | 13 | 0 | 8 | **13** | 3 | 16 | 8 | 5 | 0 | 0 | 0 | **13** | 5 | 3 | **8** | 0 | 7.90 |
| [25] - 15 Proposals | 0 | 3 | 27 | 10 | 0 | 11 | 0 | 7 | 11 | 3 | 19 | 8 | 4 | 0 | 0 | 11 | 11 | 6 | 3 | 6 | 0 | 7.71 |
| [25] - 30 Proposals | 0 | 3 | 24 | 11 | 12 | 10 | 0 | 7 | 10 | 3 | 18 | 9 | 5 | 0 | 0 | **11** | 11 | 5 | 3 | 6 | 0 | 7.61 |
| Base-Det-Cuboid | 0 | 8 | 3 | 2 | 13 | 12 | 5 | 8 | 3 | 6 | 6 | 4 | 14 | 14 | **7** | 3 | 3 | 2 | 1 | 4 | 2 | 5.80 |
| Base-NMS-Cuboid | 0 | 3 | 16 | 0 | 0 | 51 | 6 | 11 | 8 | 14 | 12 | 7 | 24 | 10 | 6 | 0 | 10 | **7** | 2 | 7 | 4 | 11.93 |
| NoOcclusion-Cuboid | 0 | 5 | 8 | 3 | 22 | 51 | 7 | 15 | 9 | 17 | 17 | 10 | 21 | 17 | 0 | 0 | 6 | 6 | 2 | 1 | 5 | 13.68 |
| NoContext-Cuboid | 0 | **9** | 7 | 2 | 27 | 51 | 6 | 17 | 7 | 18 | 16 | 6 | 21 | 23 | 5 | 0 | 4 | 2 | 1 | 5 | **6** | 13.38 |
| FullModel-Cuboid | 0 | 6 | 8 | 3 | 23 | 51 | 7 | 15 | 8 | 18 | 17 | 7 | 24 | 21 | 0 | 0 | 6 | 6 | 2 | 6 | 5 | 13.45 |
| Base-Det-CAD | 0 | 8 | 13 | 2 | 11 | 10 | 5 | 10 | 4 | 6 | 8 | 9 | 14 | 14 | **7** | 4 | 5 | 3 | 4 | 4 | 1 | 7.66 |
| Base-NMS-CAD | 0 | 2 | 43 | 3 | 0 | 48 | 6 | 16 | 9 | 14 | 21 | 15 | 23 | 14 | 5 | 6 | 6 | 5 | 2 | 5 | 4 | 15.05 |
| NoOcclusion-CAD | 0 | 4 | 52 | 4 | 25 | 49 | 0 | 21 | 9 | 17 | 30 | 18 | 24 | 24 | 0 | 0 | 0 | 6 | 4 | 3 | 0 | 17.57 |
| NoContext-CAD | 0 | 8 | 47 | 4 | 28 | 45 | 7 | 23 | 8 | **20** | 28 | **20** | 25 | 22 | 0 | 4 | 2 | 4 | **5** | 4 | 0 | 18.61 |
| FullModel-CAD | 0 | 4 | **61** | 4 | **31** | **55** | **7** | **24** | 10 | 19 | **33** | 18 | **27** | **24** | 0 | 0 | 1 | 6 | 3 | 5 | 0 | **19.22** |
| [25] - 8 Proposals | 0 | 4 | 27 | **12** | 0 | 13 | 0 | 8 | **13** | 3 | 16 | 8 | 5 | 0 | 0 | 0 | **13** | 5 | 3 | **8** | 0 | 7.90 |
| [25] - 15 Proposals (vis) | 0 | 6 | 33 | 10 | 0 | 12 | 0 | 10 | **13** | 6 | 23 | 10 | 8 | 0 | 0 | 16 | **14** | 10 | **5** | **10** | 0 | 10.12 |
| [25] - 30 Proposals (vis) | 0 | 5 | 30 | 11 | 12 | 11 | 0 | 9 | 12 | 6 | 22 | 10 | 9 | 0 | 0 | **16** | 13 | 9 | **5** | **10** | 0 | 9.96 |
| FullModel-CAD (vis) | 0 | **7** | **61** | 8 | **31** | **56** | **7** | **25** | 13 | **21** | **31** | **18** | **26** | **16** | 0 | 0 | 2 | **11** | 5 | 6 | 0 | **20.47** |

**Table 1. 3D Detection Performance on 21 Object Classes of NYUv2.** The first part of the table shows results for [25], our baselines and our full model (FullModel-CAD) when evaluating the full extent of all 3D objects (i.e., including the occluded parts) in terms of the weighted F1 score (%). The second part of the table shows F1 scores when evaluating only the visible parts. See text for details.

classes, weighted by the number of instances. An object is counted as true positive if the intersection-over-union of its 3D bounding box with respect to the associated ground truth 3D bounding box is larger than 0.3. This threshold is chosen smaller than the 0.5 threshold typically chosen for evaluating 2D detection [5] as the 3D volume intersection-over-union criterion is much more sensitive compared to its 2D counterpart.

**Ablation Study:** In this section, we evaluate the importance of the individual components in our model. First, we compare our method when using *CAD* models vs. using only simple *Cuboid* models as object representation. As illustrated in Table 1, we obtain a relative improvement in F1 score of 42.2% when using CAD object models in our full graphical model (*FullModel-CAD vs. FullModel-Cuboid*), highlighting the importance of accurate 3D geometry modeling for this task. Next, we compare our full model with versions which exclude the occlusion (*NoOcclusion*) or context (*NoContext*) terms in our model. From Table 1, it becomes evident that the occlusion term is more important than context, improving the F1 score by 9.1%. Adding the contextual relationship improves performance by 3.2%. Finally, Fig. 2 displays the 3D detection performance of our model with respect to the number of proposals (first subfigure) and superpixels (second subfigure) evaluating objects to their full extent (blue) or only the visible part (red) by clipping all bounding boxes accordingly.

**Baselines:** In this section, we quantitatively compare our method against a recently published state-of-the-art algorithm [25] and two simpler baselines derived from our full model: For our first baseline (*Base-Det*), we simply threshold our unary detections at their maximal F1 score calculated over the training set. Our second baseline (*Base-NMS*) additionally performs greedy non-maximum-

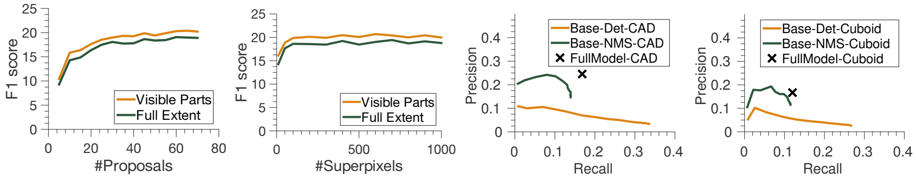

**Fig. 2. 3D Object Detection.** From left-to-right: Performance of full model wrt. number of proposals and wrt. number of superpixels. Precision-recall curves of the baselines wrt. the full model when using 3D CAD models and cuboid primitives.

suppression, selecting only non-overlapping objects from the proposal set. As our results in Table 1 show, our method yields relative improvements in F1 score of 149.4% and 27.2% wrt. *Base-Det-CAD* and *Base-NMS-CAD*, respectively. Furthermore, the third and fourth plot of Fig. 2 show the performance of the baselines in terms of precision and recall when varying the detection threshold.

We further compare our method to [25] as their setup is most similar to ours and their code for training and evaluation is available. As [25] is only able to detect the visible part of objects and has been trained on a ground truth dataset biased towards cuboids, we re-train their method on the more recent and complete NYUv2 ground truth annotations by Guo et al. [11] clipped to the visible range and report results for different number of proposals (8, 15, 30). For a fair comparison, we evaluate only the visible parts of each object (*visible*, lower part of the table). On average, we double the F1 score wrt. [25]. The differences are especially pronounced for furniture categories such as *bathtub*, *bed*, *table*, *cabinet*, *sofa* and *chair*, showing the benefits of leveraging powerful 3D models during inference. Furthermore, we note that the performance of [25] drops with the number of proposals while the performance of our method keeps increasing (Fig. 2), which is a favorable property considering future work at larger scales. For completeness, we also show the performance of [25] on the unclipped bounding boxes (first rows of Table 1).

**Qualitative Results:** Fig. 3 visualizes our inference results on a number of representative NYUv2 test images. Each panel displays (left-to-right) the inferred object wireframe models, virtual 3D renderings and the corresponding semantic segmentation. Note how our approach is able to recover even complex shapes (e.g., chair in row 1, right column) and detects heavily occluded 3D objects (e.g., bathtub and toilet in row 5, right column). The two lower rows show some failure cases of our method. In the top-left case, the sink is detected correctly, but intersects the volume of the containing cabinet which is removed from the solution. For most other cases, either the semantic class predictions which we take as input are corrupt, or the objects in the scene do not belong to the considered categories (such as *person*, *piano* or *billiard table*). However, note that even in those cases, the retrieved explanations are *functionally* plausible. Furthermore, flat objects are often missed due to the low probability of their volume intersecting the ground truth in 3D. Thus (and for completeness) we

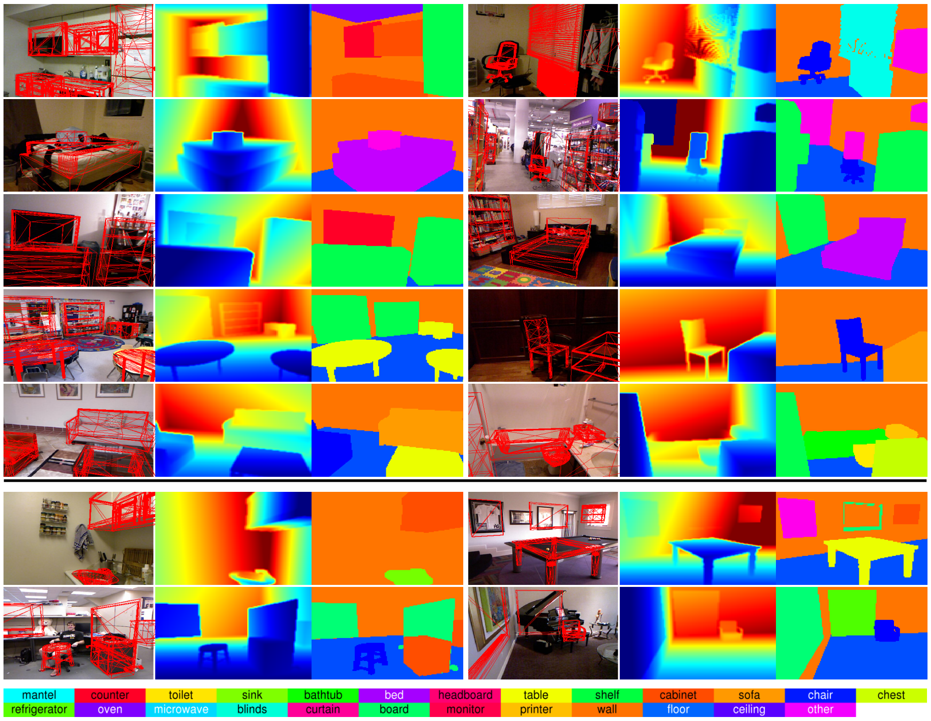

**Fig. 3. Inference Results.** Each subfigure shows: Object wireframes, rendered depth map and induced semantic segmentation.

also provide an evaluation of the objects projected onto the 2D image (similar to the one carried out in [25]) in our supplementary material.

**Runtime:** On average, our implementation takes 119.2 s for generating proposals ($\sim 6,000$ samples/second via OpenGL), 7.9 s for factor graph construction and 0.7 s for inference on an i7 CPU running at 2.5 Ghz.

## 5 Conclusion

In this paper, we have proposed a model for 3D indoor scene understanding from RGB-D images which jointly considers the layout, objects and superpixels. Our experiments show improvements with respect to two custom baselines as well as a state-of-the-art scene understanding approach which can be mainly attributed to two facts: First, we sample more accurate 3D CAD proposals directly from the unary distribution and second, the proposed model properly accounts for occlusions and satisfies visibility constraints. In the future, we plan to address more complete scene reconstructions, e.g., obtained via volumetric fusion in order to increase object visibility and thus inference reliability. Furthermore, we plan to extend our model to object based understanding of dynamic scenes from RGB/RGB-D video sequences by reasoning about 3D scene flow [28, 29].

## References

1. Aubry, M., Maturana, D., Efros, A., Russell, B., Sivic, J.: Seeing 3D chairs: exemplar part-based 2D-3D alignment using a large dataset of CAD models. In: CVPR (2014)
2. Blake, A., Kohli, P., Rother, C.: Markov Random Fields for Vision and Image Processing. MIT Press (2011)
3. Carreira, J., Sminchisescu, C.: CPMC: automatic object segmentation using constrained parametric min-cuts. PAMI 34(7), 1312–1328 (2012)
4. Choi, W., Chao, Y.W., Pantofaru, C., Savarese, S.: Understanding indoor scenes using 3D geometric phrases. In: CVPR (2013)
5. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. IJCV 88(2), 303–338 (2010)
6. Felzenszwalb, P.F., Girshick, R.B., McAllester, D.A.: Cascade object detection with deformable part models. In: CVPR (2010)
7. Felzenszwalb, P.F., Mcauley, J.J.: Fast inference with min-sum matrix product. PAMI 33(12), 2549–2554 (2011)
8. Fouhey, D.F., Gupta, A., Hebert, M.: Data-driven 3D primitives for single image understanding. In: ICCV (2013)
9. Gilks, W., Richardson, S.: Markov Chain Monte Carlo in Practice. Chapman & Hall (1995)
10. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR (2014)
11. Guo, R., Hoiem, D.: Support surface prediction in indoor scenes. In: ICCV (2013)
12. Gupta, S., Arbelaez, P., Malik, J.: Perceptual organization and recognition of indoor scenes from RGB-D images. In: CVPR (2013)
13. Gupta, S., Girshick, R., Arbeláez, P., Malik, J.: Learning rich features from RGB-D images for object detection and segmentation. In: ECCV (2014)
14. Güney, F., Geiger, A.: Displets: Resolving stereo ambiguities using object knowledge. In: CVPR (2015)
15. Hedau, V., Hoiem, D., Forsyth, D.: Recovering the spatial layout of cluttered rooms. In: ICCV (2009)
16. Jia, Z., Gallagher, A., Saxena, A., Chen, T.: 3D-based reasoning with blocks, support, and stability. In: CVPR (2013)
17. Jiang, H., Xiao, J.: A linear approach to matching cuboids in RGB-D images. In: CVPR (2013)
18. Kim, B., Xu, S., Savarese, S.: Accurate localization of 3D objects from RGB-D data using segmentation hypotheses. In: CVPR (2013)
19. Kohli, P., Ladicky, L., Torr, P.H.S.: Robust higher order potentials for enforcing label consistency. IJCV 82(3), 302–324 (2009)
20. Kohli, P., Pawan Kumar, M.: Energy minimization for linear envelope MRFs. In: CVPR (2010)
21. Komodakis, N., Paragios, N.: Beyond pairwise energies: Efficient optimization for higher-order MRFs. In: CVPR (2009)
22. Lee, D., Gupta, A., Hebert, M., Kanade, T.: Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces. In: NIPS (2010)
23. Lim, J.J., Khosla, A., Torralba, A.: FPM: Fine pose parts-based model with 3D CAD models. In: ECCV (2014)
24. Lim, J.J., Pirsiavash, H., Torralba, A.: Parsing IKEA objects: Fine pose estimation. In: ICCV (2013)

25. Lin, D., Fidler, S., Urtasun, R.: Holistic scene understanding for 3D object detection with RGB-D cameras. In: ICCV (2013)
26. Mansinghka, V., Kulkarni, T., Perov, Y., Tenenbaum, J.: Approximate bayesian image interpretation using generative probabilistic graphics programs. NIPS 2013 (2013)
27. Mcauley, J.J., Caetano, T.S.: Faster algorithms for max-product message-passing. JMLR 12, 1349–1388 (2011)
28. Menze, M., Geiger, A.: Object scene flow for autonomous vehicles. In: CVPR (2015)
29. Menze, M., Heipke, C., Geiger, A.: Joint 3d estimation of vehicles and scene flow. In: ISA (2015)
30. Newcombe, R.A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A.J., Kohli, P., Shotton, J., Hodges, S., Fitzgibbon, A.: Kinectfusion: Real-time dense surface mapping and tracking. In: ISMAR (2011)
31. Potetz, B., Lee, T.S.: Efficient belief propagation for higher-order cliques using linear constraint nodes. CVIU 112(1), 39–54 (2008)
32. Ren, X., Bo, L., Fox, D.: RGB-(D) scene labeling: Features and algorithms. In: CVPR (2012)
33. Roberts, L.G.: Machine perception of three-dimensional solids. Ph.D. thesis, Massachusetts Institute of Technology (1963)
34. Rother, C., Kohli, P., Feng, W., Jia, J.: Minimizing sparse higher order energy functions of discrete variables. In: CVPR (2009)
35. Satkin, S., Hebert, M.: 3DNN: viewpoint invariant 3D geometry matching for scene understanding. In: ICCV (2013)
36. Schwing, A., Urtasun, R.: Efficient exact inference for 3D indoor scene understanding. In: ECCV (2012)
37. Schwing, A.G., Fidler, S., Pollefeys, M., Urtasun, R.: Box in the box: Joint 3D layout and object reasoning from single images. In: ICCV (2013)
38. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from RGB-D images. In: ECCV (2012)
39. Song, S., Xiao, J.: Sliding shapes for 3D object detection in depth images. In: ECCV (2014)
40. Tarlow, D., Givoni, I.E., Zemel, R.S.: Hop-map: Efficient message passing with high order potentials. In: AISTATS (2010)
41. Tighe, J., Niethammer, M., Lazebnik, S.: Scene parsing with object instances and occlusion ordering. In: CVPR (2014)
42. Tsai, G., Xu, C., Liu, J., Kuipers, B.: Real-time indoor scene understanding using bayesian filtering with motion cues. In: ICCV (2011)
43. Wang, C., Komodakis, N., Paragios, N.: Markov random field modeling, inference & learning in computer vision & image understanding: A survey. CVIU 117(11), 1610–1627 (2013)
44. Yamaguchi, K., McAllester, D., Urtasun, R.: Robust monocular epipolar flow estimation. In: CVPR (2013)
45. Zhang, H., Geiger, A., Urtasun, R.: Understanding high-level semantics by modeling traffic patterns. In: ICCV (2013)
46. Zhang, Y., Song, S., Tan, P., Xiao, J.: PanoContext: A whole-room 3D context model for panoramic scene understanding. In: ECCV (2014)
47. Zheng, B., Zhao, Y., Yu, J.C., Ikeuchi, K., Zhu, S.C.: Beyond point clouds: Scene understanding by reasoning geometry and physics. In: CVPR (2013)
48. Zia, M., Stark, M., Schiele, B., Schindler, K.: Detailed 3D representations for object recognition and modeling. PAMI 35(11), 2608–2623 (November 2013)