Separating Objects and Clutter in Indoor Scenes

S. H. Khan* Xuming He† M. Bannamoun* F. Sohel* R. Togneri‡
*School of CSSE UWA †NICTA and ANU ‡School of EECE UWA
{salman.khan,mohammed.bennamoun,ferdous.sohel,roberto.togneri}@uwa.edu.au, xuming.he@nicta.com.au

Abstract

Objects' spatial layout estimation and clutter identification are two important tasks to understand indoor scenes. We propose to solve both of these problems in a joint framework using RGBD images of indoor scenes. In contrast to recent approaches which focus on either one of these two problems, we perform 'fine grained structure categorization' by predicting all the major objects and simultaneously labeling the cluttered regions. A conditional random field model is proposed to incorporate a rich set of local appearance, geometric features and interactions between the scene elements. We take a structural learning approach with a loss of 3D localisation to estimate the model parameters from a large annotated RGBD dataset, and a mixed integer linear programming formulation for inference. We demonstrate that our approach is able to detect cuboids and estimate cluttered regions across many different object and scene categories in the presence of occlusion, illumination and appearance variations.

1. Introduction

We live in a three dimensional world where objects interact with each other according to a rich set of physical and geometrical constraints. Therefore, merely recognizing objects or segmenting an image into a set of semantic classes does not always provide a meaningful interpretation of the scene and its properties. A better understanding of real-world scenes requires a holistic perspective, exploring both semantic and 3D structures of objects as well as the rich relationship among them [12, 29, 19, 33]. To this end, one fundamental task is that of the volumetric reasoning about generic 3D objects and their 3D spatial layout.

Among different approaches to tackle the generic 3D object reasoning problem, much progress has been made based on representing objects as 3D geometric primitives, such as cuboids. Some of the first efforts focus on the 3D spatial layout and cuboid-like objects in indoor scenes from monocular imagery [22, 14, 31]. Owing to the complex structure of the scenes, additional depth information has recently been introduced to obtain more robust estimation [23, 16, 13, 25]. However, real-world scenes are composed of not only large regular-shaped structures and objects (such as walls, floor, furniture), but also irregular shaped objects and cluttered regions which cannot be represented well by object-level primitives. The overlay of different types of scene elements makes the procedure of localizing 3D objects fragile and prone to misalignment.

Figure 1: With a given RGBD image (*left column*), our method explores the 3D structures in an indoor scene and estimates their geometry using cuboids (*right image*). It also identifies cluttered/unorganized regions in a scene (*shown in orange*) which can be of interest for tasks such as robot grasping.

Most previous work has focused on clutter reasoning in the scene layout estimation problem [14, 29, 32]. Such object clutter is usually defined at a coarse-level, including everything other than the global layout, which is insufficient for object-level parsing. To tackle the problem of 3D object cuboid estimation, we attempt to use *clutter* in a more fine-grained sense, referring to any unordered region other than the main structures *and* major cuboid-like objects in the scene, as shown in Fig. 1.

We aim to address the problem of 3D object cuboid detection in a cluttered scene. In this work, we propose to jointly localize generic 3D objects (represented by cuboids) and label cluttered regions from an RGBD image. Unlike the recent cuboid detection techniques, which consider such regions as background, our method explicitly models the appearance and geometric property of the fine-grained cluttered regions. We incorporate scene context (in the form of object and clutter)

to better model the regular-shaped objects and their interaction with other types of regions in a scene.

We adopt the approach in [16] for representing an indoor scene, which models a room as a set of hypothesized cuboids and local surfaces defined by superpixels. To cope with clutters, we formulate the joint detection task using a higher-order Conditional Random Field model (CRF) on superpixels and cuboid hypotheses generated by a bottom-up grouping process. Our CRF approach extends the linear model of [16] in several aspects. First, we introduce a random field of local surfaces (superpixels) that captures the local appearance and spatial smoothness of cluttered and noncluttered regions. In addition, we improve the cuboid representation by generating two types of cuboid hypotheses, one of which corresponds to regular objects inside a scene and the other is for the main structures of a scene, such as floor and walls. Furthermore, we incorporate both the consistency between superpixel labels and cuboid hypotheses and the occlusion relation between cluttered regions and cuboid objects.

More importantly, we take a structural learning approach to estimate the CRF parameters from an annotated indoor dataset, which enables us to systematically incorporate more features into our model and to avoid tedious manual tuning. We use a max-margin based objective function that minimizes a loss defined on cuboid detection. Similar to [16], the (loss-augmented) MAP inference of our CRF model can be formulated as a mixed integer linear programming (MILP) formulation. We empirically show that the MILP can be globally optimized with the Branch-and-Bound method within a time of seconds to find a solution in most cases. During testing, the MAP estimate of our CRF not only detects cuboid objects but also identifies the cluttered regions. We evaluate our method on the NYU Kinect v2 dataset with augmented cuboid and clutter annotations, and demonstrate that the proposed approach achieves superior performance to the state of the art.

2. Related Work

Localizing and predicting the geometry of generic objects using cuboids is a challenging problem in highly cluttered indoor scenes. A number of approaches extend 2D appearance-based methods to the task of predicting the 3D cuboids. Variants of the Deformable Parts based Model (DPM) [8] have been used for 3D cuboid prediction [24, 25, 30]. However, they do not consider clutter and heavy occlusion in the scene. In [23], the Constrained Parametric Min-cut (CPMC) [6] was extended from 2D to RGBD to generate a cuboid hypotheses set. In contrast, we directly generate two types of cuboid proposals in a bottom-up fashion [16], thus providing a simpler and efficient procedure which is better suited for indoor RGBD data.

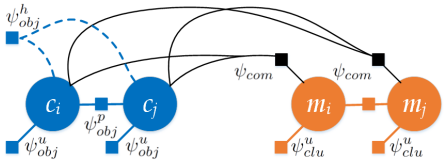


Figure 2: Graph structure representation for the potentials defined on the object cuboids and the cluttered/non-cluttered regions. *(Best viewed in color)*

Based on the physical and geometrical constraints, a number of approaches have been proposed for 3D object and scene parsing, e.g., [33, 18, 3]. The basic idea is to incorporate contextual relationships at a higher level to avoid false detection. Silberman et al. [26] predict the support surfaces and semantic object classes in an indoor scene. Geometric and semantic relationships between different object classes are modeled in works such as [20, 26, 10]. Gupta et al. [12] use a parse graph to consider mechanical and geometric relationships amongst objects represented by 3D boxes. For indoor scenes, volumetric reasoning is performed for 2D [22] and RGBD images [16] to detect cuboids. However, none of these works estimate cuboids and clutter jointly using relevant constraints.

The joint estimation of clutter along with the room layouts has previously been shown to enhance performance. Wang et al. [29] predict clutter and layouts in a discriminative setting where clutter is modeled using hidden variables. Recently, Zhang et al. [32] employed RGBD data for joint layout and clutter estimation and efficiently perform inference by potential decomposition. However, these works are limited to only scene layout estimation and label everything else as clutter. Recently, Schwing et al. [25] used monocular imagery to jointly estimate room layout along with one major object present in a bedroom scene. In this work, we estimate the scene bounding structures as well as 'all' of the major objects using 3D cuboids.

3. Our Approach

Indoor scenes contain material structures (e.g., ceiling, walls) and the regular-shaped objects which we term as *non-cluttered* regions. In contrast, *cluttered* regions consist of small, indistinguishable objects (e.g., stationery on an office table) or jumbled regions in a scene (e.g., clothes piled on a bed). We represent an indoor scene as an overlay of the cluttered regions (modeled as local surfaces) and the non-cluttered regions (modeled using 3D cuboids). Our goal is to describe an RGBD image with an optimal set of cuboids and pixel-level labeling of cluttered regions.

Our approach first generates a set of cuboid hy-

potheses based on image and depth cues, which aims to cover the majority of true object locations. Taking them as the potential object candidates, we can significantly reduce the search space of 3D cuboids and construct a CRF on the image/depth superpixels and these candidates. We will first introduce our CRF formulation assuming the cuboid hypotheses are given, and refer the reader to Sec. 4 for details on the cuboid extraction procedure.

3.1. CRF Formulation

Given an RGBD image, denoted by $\mathcal{I}$, we decompose it into a number of contiguous partitions, i.e., superpixels: $\mathcal{S} = \{s_1, \cdots, s_J\}$, where J is the total number of superpixels. We associate a binary membership variable m_j with each superpixel s_j to indicate whether it belongs to the cluttered or non-cluttered regions, and denote $\mathbf{m} = \{m_1, \cdots, m_J\}$. The set of cuboid hypotheses is denoted by $\mathcal{O} = \{o_1, \cdots, o_K\}$, where K is the total number of cuboid hypotheses. For each cuboid, we introduce a binary variable c_k to indicate whether the k^{th} cuboid hypothesis is active or not, and denote $\mathbf{c} = \{c_1, \cdots, c_K\}$.

Note that for indoor scenes, the room structures such as walls and floor bound the scene and therefore appear as planar regions, which have different geometric properties from the ordinary object cuboids. To encode such different constraints, we define two types of cuboids in the hypotheses set, namely the *scene bounding cuboids* ($\mathcal{O}_{sbc}$) and the *object cuboids* ($\mathcal{O}_{oc}$). The cuboid extraction procedure for both types of cuboids is described in Sec. 4.

We build a CRF model on the superpixel clutter variables $\mathbf{m}$ and the object variables $\mathbf{c}$ to describe the properties of clutter, objects and their relationship in the scene. Formally, we define the Gibbs energy of the CRF as follows,

$$E(\mathbf{m}, \mathbf{c}|\mathcal{I}) = E_{obj}(\mathbf{c}) + E_{sp}(\mathbf{m}) + E_{com}(\mathbf{m}, \mathbf{c}), \quad (1)$$

where $E_{obj}(\mathbf{c})$, $E_{sp}(\mathbf{m})$ captures the object level and the superpixel level properties respectively, and $E_{com}(\mathbf{m}, \mathbf{c})$ models the interactions between them.

More specifically, the **first** term, $E_{obj}(\mathbf{c})$, is defined as a combination of three potential functions:

$$E_{obj}(\mathbf{c}) = \sum_{k=1}^{K} \left[\psi^u_{obj}(c_k) + \psi^h_{obj}(c_k)\right] + \sum_{i<j} \psi^p_{obj}(c_i, c_j), \quad (2)$$

where the unary potential $\psi^u_{obj}(c_k)$ expresses the data likelihood of k^{th} object hypothesis, $\psi^h_{obj}(c_k)$ encodes a MDL prior on the number of active cuboids, and the pairwise potential $\psi^p_{obj}(c_i, c_j)$ models the physical and geometrical relationships between cuboids.

Similarly at the superpixel level, the **second** term, E_{sp}, consists of two potential functions:

$$E_{sp}(\mathbf{m}) = \sum_{j=1}^{J} \psi^u_{sp}(m_j) + \sum_{(i,j)\in N_s} \psi^p_{sp}(m_i, m_j), \quad (3)$$

where the unary potential $\psi^u_{sp}(m_j)$ is the data likelihood of a superpixel's label, and the pairwise potential $\psi^p_{sp}(m_i, m_j)$ encodes the spatial smoothness between neighboring superpixels, denoted by N_s.

The **third** term in Eq. (1), is the compatibility constraint which enforces the consistency of the cuboid activations and the superpixel labeling:

$$E_{com}(\mathbf{m}, \mathbf{c}) = \sum_{j=1}^{J} \psi_{com}(m_j, \mathbf{c}). \quad (4)$$

In the following discussion, we will explain the different costs which constitute the energies defined in Eqs. (2), (3) and (4).

3.2. Potentials on Cuboids

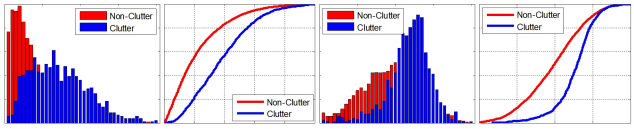
3.2.1 Unary Potential on Cuboids

The unary potential of a cuboid hypothesis ψ^u_{obj} measures the likelihood of a cuboid hypothesis being active based on its appearance, physical and geometrical properties. Instead of specifying local matching costs manually, we extract a set of informative multi-modal features from image/depth and each cuboid, and take a learning approach to predict the local matching quality. Specifically, we generate seven different types of cuboid features ($\mathbf{f}^{obj}_k$) as follows.

Volumetric occupancy feature f^{occ}_k measures the portion of the k^{th} cuboid occupied by the 3D point data. We define f^{occ}_k as the ratio between the empty volume inside a cuboid (v^k_e) to the total volume of a cuboid (v^k_b): $f^{occ}_k = v^k_e / v^k_b$. The volumes are estimated by discretizing the 3D space into voxels and counting the number of voxels that are occupied by 3D points or not. All invisible voxels behind occupied voxels are also treated as occupied.

Color consistency feature f^{col}_k encodes the color variation of the k^{th} cuboid. Object instances normally have consistent appearance while cluttered regions tend to have a skewed color distribution (Fig. 3). We fit a GMM with three components on the color distribution of pixels enclosed in a cuboid and measure the average deviation. Specifically, the feature is defined as: $f^{col}_k = \sum_{\forall p \in o_k} \omega_u \|v_p - \sigma_u\|$, where v_p denotes the color of a pixel p, σ_u is the mean of the closest component (u) and ω_u is the mixture proportion.

Normal consistency feature f^{nor}_k measures the normal variation of the k^{th} cuboid. The distribution of 3D point normals inside the cluttered regions has a larger variance (Fig. 3). In contrast, the normal directions of regular objects are usually aligned with the three perpendicular faces of the cuboid. Similar to the color feature, we calculate the variation of 3D point normals with respect to the closest dominant direction.



(a) (b) (c) (d)

Figure 3: The distribution of variation in color for cluttered and non-cluttered regions in the RMRC training set is compared in (a), (b). Comparison for variation in normals is shown in (c), (d). (b) and (d) are the cumulative distributions.

Tightness feature f_k^{tig} describes how loosely the 3D points fit the cuboid proposals. For each visible face of a cuboid, we calculate the ratio between the area of minimum bounding rectangle tightly enclosing all points (A_{rec}^f) to the area of the face (A_f). We take the weighted average of the tightness ratios of the cuboid faces to define $f_k^{tig} = \frac{1}{\sum_f [\![A_{rec}^f \neq 0]\!]} \sum_{\forall f \in Faces} \frac{A_{rec}^f}{A_f}$.

Support feature f_k^{sup} measures how likely each cuboid is supported either by another cuboid or clutter. We estimate the support by calculating the number of 3D points that fall in the space surrounding the cuboid ($\tau\%$[1] additional space along each dimension). The feature is defined as: $f_k^{sup} = \frac{e_{o'_k} - e_{o_k}}{e_{o_k}}$, where, $e_{o'_k}$ and e_{o_k} denote the number of points enclosed by the extended cuboid and the original cuboid respectively.

Geometric plausibility feature f_k^{geo} measures the likelihood that a cuboid has a plausible 3D object shape. Using 3D geometrical features (sizes and aspect ratios), we train a Random Forest (RF) classifier to score the geometric plausibility. The score is used to define f_k^{geo}, which filters out the less likely cuboid candidates e.g., very thin cuboids or those with irregular aspect ratio.

Cuboid size feature f_k^{och} measures the relative size of a cuboid w.r.t the average object size in the dataset. Let ℓ_{ldl} denote the maximum diagonal length of a cuboid and $\bar{\ell}_{ldl}$ is the mean length of objects. We define $f_k^{och} = \ell_{ldl}/\bar{\ell}_{ldl}$, which helps control the number of valid cuboids by removing small ones.

Given the feature descriptor $\mathbf{f}_k^{obj}$, we train a RF classifier on $\mathbf{f}^{obj}$ and define the unary potential based on the output of the RF, $P(c_k = 1|\mathbf{f}_k^{obj})$:

$$\psi_{obj}^u(c_k) = \lambda_{bbu} \mu_k^{bbu} c_k, \tag{5}$$

where λ_{bbu} is the weighting coefficient and $\mu_k^{bbu} = -\log \frac{P(c_k=1|\mathbf{f}_k^{obj})}{1-P(c_k=1|\mathbf{f}_k^{obj})}$. Note that those features are automatically weighted and combined by the RF for predicting the local matching cost.

[1]Based on empirical tests, τ is set to 2.5% in this work.

3.2.2 Cuboid MDL Potential

The MDL principle prefers to explain a given image compactly in terms of a small number of cuboids, instead of a complex representation consisting of an unnecessarily large number of cuboids [3, 16]. We define the MDL potential ψ_{obj}^h in Eq. (2) as: $\psi_{obj}^h(c_k) = \lambda_{mdl} c_k$, where $\lambda_{mdl} > 0$ is the weighting parameter.

3.2.3 Pairwise Potentials on Cuboids

We follow [16] and the pairwise energy in Eq. (2) decomposes in to view obstruction and box intersection potentials:

$$\psi_{obj}^p(c_i, c_j) = \psi_{obs}^p(c_i, c_j) + \psi_{int}^p(c_i, c_j). \tag{6}$$

As we have two types of cuboids, our pairwise potentials on cuboids are parametrized according to the configuration of each cuboid pair.

View obstruction potential (ψ_{obs}^p) encodes the visibility constraint between a pair of cuboids, and is expressed as follows:

$$\psi_{obs}^p(c_i, c_j) = \lambda_{obs} \tilde{\mu}_{i,j}^{obs} c_i c_j = \lambda_{obs} \tilde{\mu}_{i,j}^{obs} y_{i,j} \tag{7}$$

where, $\tilde{\mu}_{i,j}^{obs}$ is the view obstruction cost, λ_{obs} is a weighting parameter and $y_{i,j}$ is an auxiliary boolean variable introduced to linearize the pairwise term [16]. The view obstruction cost $\tilde{\mu}_{i,j}^{obs}$ computes the intersection of 2D projections of two cuboids and induces a penalty when a larger cuboid lies in front of a smaller but farther cuboid. Let $\mu_{i,j}^{obs} = (A_{c_i} \cap A_{c_j})/A_{c_i}$ where, A_{c_i} and A_{c_j} are the areas of the 2D projections of cuboid hypotheses c_i and c_j on the image plane respectively and c_i is the farther cuboid w.r.t the viewer. The cost $\tilde{\mu}_{i,j}^{obs} = \mu_{i,j}^{obs}$ if $\mu_{i,j}^{obs} < \alpha_{obs}$ and infinity otherwise. This allows partial occlusion with a penalty but avoids heavy occlusion. We use $\alpha_{obs} = 60\%$ for object cuboids ($\mathcal{O}_{oc}$). For the case of scene bounding cuboids ($\mathcal{O}_{sbc}$), we relax the obstruction cost by a factor of 0.1 in Eq. (7) and set $\alpha'_{obs} = 80\%$.

Cuboid intersection potential (ψ_{int}^p) penalizes volumetric overlaps between cuboid pairs as two objects cannot penetrate each other, and is defined as:

$$\psi_{int}^p(c_i, c_j) = \lambda_{int} \tilde{\mu}_{i,j}^{int} c_i c_j = \lambda_{int} \tilde{\mu}_{i,j}^{int} x_{i,j} \tag{8}$$

where, $\tilde{\mu}_{i,j}^{int}$ is the cuboid intersection cost, λ_{int} is a weighting parameter and $x_{i,j}$ is an auxiliary boolean variable introduced to linearize the pairwise cost. The cuboid intersection cost induces a soft penalty as long as the intersection is smaller than a threshold. Let $\mu_{i,j}^{int}$ be the normalized intersection, and we define $\tilde{\mu}_{i,j}^{int} = \mu_{i,j}^{int}$ if $0 \leq \mu_{i,j}^{int} < \alpha_{int}$ and infinity otherwise. We set $\alpha_{int} = 10\%$ for the case of object cuboids and $\alpha'_{obs} = 50\%$ for all scene bounding cuboids.

3.3. Potentials on Superpixels

We decompose an input image into superpixels based on the hierarchical image segmentation [2]. The unary potential on each superpixel captures the appearance and texture properties of cluttered and non-cluttered regions. We employ the kernel descriptor framework of [4, 5] to convert pixel attributes to rich patch level feature representations. We extract several cues including image and depth gradient, color, surface normal, LBP and self similarity. A RF classifier is trained on these dense features, which predicts the probability of a region being a clutter or non-clutter. We use the negative log odds ratio as a cost μ_j^{app}, weighted by the parameter λ_{app} and define the unary in Eq. (3) as $\psi_{sp}^{u}(m_j) = \lambda_{app}\mu_j^{app} m_j$.

For the superpixel pairwise term, we define a contrast-sensitive Potts model on spatially neighboring superpixels, which encourages the smoothness of the clutter and non-clutter regions:

$$\psi_{sp}^{p}(m_i, m_j) = \lambda_{smo}\mu_{i,j}^{smo}(m_i + m_j - m_i \cdot m_j), \quad (9)$$

where, $\mu_{i,j}^{smo} = \exp(-\|\bar{v}_i - \bar{v}_j\|^2/\sigma_c^2)$, $\bar{v}_i, \bar{v}_j$ are the mean color of superpixel s_i and s_j. We use $w_{i,j}$ as an auxiliary boolean variable to linearize the quadratic term $m_i \cdot m_j$ (see Sec. 5).

3.4. Superpixel-Cuboid Compatibility

The compatibility term links the superpixels labeling to the cuboid selection task, which enforces consistency between the lower level and the higher level of the scene representation. Our compatibility potential consists of two terms, one for superpixel membership ψ_{mem} and the other for occlusion relation ψ_{occ}:

$$\psi_{com}(m_j, \mathbf{c}) = \psi_{mem}(m_j, \mathbf{c}) + \sum_k \psi_{occ}(m_j, c_k), \quad (10)$$

Superpixel membership potential (ψ_{mem}) defines a constraint that a superpixel is associated with at least one active cuboid if it is not a cluttered region: $m_j \leq \sum_{k:s_j \in o_k} c_k$. Equivalently, the corresponding potential function is a higher-order term (Fig. 2):

$$\psi_{mem}(m_j, \mathbf{c}) = \lambda_\infty [\![m_j \neq \max_{k:s_j \in o_k} c_k]\!], \quad (11)$$

where λ_∞ is an infinite (very large) penalty cost.

Superpixel-cuboid occlusion potential (ψ_{occ}) encodes that a cuboid should not appear in front of a superpixel which is classified as clutter, i.e., a detected cuboid cannot completely occlude a superpixel on the 2D plane which takes a clutter label.

$$\psi_{occ}(m_j, c_k) = \lambda_{occ}\mu_{jk}^{occ}\overline{m}_j c_k = \lambda_{occ}\mu_{jk}^{occ} z_{jk} \quad (12)$$

where, $\overline{m}_j = 1 - m_j$, and z_{jk} is the auxiliary variable for linearization. The cost $\mu_{jk}^{occ} = \frac{(A_{m_j} \cap A_{c_k})}{A}$ and A is the area of the further element (either cuboid or superpixel). The cost $\tilde{\mu}_{jk}^{occ}$ and parameter α_{occ} are defined similar to the view obstruction potential in Sec. 3.2.3.

4. Cuboid Hypothesis Generation

Our method for initial cuboid hypothesis generation is based on a bottom-up clustering-and-fitting procedure, which generates both *object cuboids* and *scene bounding cuboids*. Specifically, we first extract homogeneous regions from a normal image using SLIC [1]. Gaussian smoothing is performed to remove isolated regions and similar regions are merged using the DBSCAN clustering algorithm [7]. The neighborhood of each resulting region is found and the inlier points in each region are estimated using the RANSAC algorithm. We then estimate three major perpendicular directions of a room as in [26], denoted as x, z (horizontal) and y (vertical).

For *object cuboids*, we adopt a fitting method similar to [16]. The cuboids identified using this procedure usually capture objects whose two or more sides are visible, but cannot capture the room structure. To propose *scene bounding cuboids*, we also generate cuboids which cover only one planar region. Among all the planar regions, we first remove the smaller ones ($< 5\%$ of the image size) and those not aligned with the three dominant directions. We then select the planar regions which are farthest from the camera view point. The cuboids enclosing these planar regions are included in the hypotheses set as the *scene bounding cuboids*. The detected cuboid proposals are ranked using the cuboid unary potential (Eq. (5)) and the top 60 cuboids are selected for our CRF inference.

5. Model Inference and Learning

5.1. Inference as MILP

Given an RGBD image $\mathcal{I}$, we parse the input into a set of cuboids and cluttered/noncluttered regions by inferring the most likely configuration of clutter label variables $\mathbf{m}$ and the cuboid hypotheses labels $\mathbf{c}$. Equivalently, we minimize the CRF energy:

$$\{\mathbf{m}^*, \mathbf{c}^*\} = \underset{\mathbf{m},\mathbf{c}}{\operatorname{argmin}} E(\mathbf{m}, \mathbf{c}|\mathcal{I}). \quad (13)$$

We adopt the relaxation method in [16, 11] and transform the minimization in Eq. (13) into a Mixed Integer Linear Program (MILP) with linear constraints. The MILP formulation can be solved much faster compared to the original ILP, using the branch and bound method.

	Small gap	Large gap	Cuts	LP relax.
Time (sec)	$1.84 \pm 31\%$	$1.31 \pm 24\%$	$0.45 \pm 13\%$	$0.001 \pm 0.4\%$
Det. Rate	26.8%	26.1%	24.4%	19.9%

Table 1: Inference running time comparisons for variants of MILP formulation.

Specifically, for the pairwise view obstruction cost in Eq. (7), we introduce $y_{i,j}$ for $c_i \cdot c_j$ with constraints: $c_i \geq y_{i,j}, c_j \geq y_{i,j}, y_{i,j} \geq c_i + c_j - 1$. Similarly, we introduce $x_{i,j}$ for the pairwise cuboid intersection cost. Also, we use an inequality $c_i + c_j \leq 1$ for the infinity cost constraint of $\tilde{\mu}_{i,j}^{obs}$ and $\tilde{\mu}_{i,j}^{int}$. These equivalent transforms can also be applied to $w_{i,j}$ for $m_i \cdot m_j$ in the superpixel pairwise potential, and $z_{j,k}$ for $\overline{m}_j c_k$ in the superpixel-cuboid potential. For clarity, we denote the complete set of linear inequality constraints for $\mathbf{c}$ and $\mathbf{m}$ as $\mathcal{LC}$ and include the details in the supplementary material. The complete MILP formulation with linear objective function and constraints is given by:

$$\min_{\mathbf{m},\mathbf{c},\mathbf{x},\mathbf{y},\mathbf{w},\mathbf{z}} E(\mathbf{m},\mathbf{c},\mathbf{x},\mathbf{y},\mathbf{w},\mathbf{z}|\mathcal{I}) \tag{14}$$

$$\text{s.t.} \quad \text{linear inequality constraints in } \mathcal{LC},$$

$$m_j,\, c_k \in \{0,1\}, \qquad \forall j,k \tag{15}$$

$$w_{i,j}, x_{i,j}, y_{i,j}, z_{j,k} \geq 0, \qquad \forall i,j,k \tag{16}$$

We solve the MILP problem in Eqs. (14) - (16) by the Branch and Bound method in the GLPK solver [21].

Algorithmic Efficiency: We empirically evaluate the efficiency of the Branch and Bound algorithm on the scene parsing problem introduced in Sec. 6. Tab. 1 lists the average time it takes to reach the optimal solution on a 3.4GHz machine. On average, $819 \pm 48\%$ variables are involved in each inference and the final MILP gap is zero for 98.5% of the cases on the whole dataset. In this work, we use a MILP gap tolerance of 0.001, however, it turns out that increasing the MILP gap by a factor of 100 causes a minute performance drop and a more efficient inference. Including cuts (cover cuts, Gomory mixed cuts, mixed integer rounding cuts, clique cuts) results in a much faster convergence at the expense of an average of 8% performance degradation and a 5% increase in memory requirements. When $\mathbf{c}$ and $\mathbf{m}$ are relaxed to get the corresponding LP which has a polynomial time convergence guarantee, the performance on the detection task decreases by 26% compared to the MILP formulation. These performance comparisons are computed at the 40% Jaccard Index (JI) threshold for cuboid detection.

5.2. Parameter Learning

We take a structural learning approach to estimate the model parameters from a fully annotated training dataset. We denote the model outputs $(\mathbf{m},\mathbf{c})$ as $\mathbf{t}$, and the model parameters $(\lambda_{bbu}, \lambda_{mdl}, \lambda_{obs}, \lambda_{int}, \lambda_{app}, \lambda_{smo}, \lambda_{occ})$ as $\boldsymbol{\lambda}$. The training set consists of a set of annotated images $\mathcal{T} = \{(\mathbf{t}^n, \mathcal{I}^n\}_{1\times N}$.

We apply the structured SVM framework with margin re-scaling [27], which uses the cutting plane algorithm [17] to search the optimal parameter setting (see the supplementary materials for details of the learning algorithm). We use the IOU loss function on cuboid matching as our loss function in learning, which is defined as $\Delta(\mathbf{t}^{(n)},\mathbf{t}) = \sum_i \left(1 - \frac{|o_i^{(n)} \cap o_i|}{|o_i^{(n)} \cup o_i|}\right)$ and o_i is the 3D cuboid associated with c_i. The algorithm efficiently adds low energy labelings to the active constraints set and updates the parameters such that the ground-truth has the lowest energy.

6. Experiments and Analysis

6.1. Dataset and Setup

We evaluate our method on the 3D detection dataset released as part of the Reconstruction Meets Recognition Challenge (RMRC), 2013. It contains 1074 RGBD images taken from the NYU Depth v2 dataset. Each image comes with 3D bounding box annotations. There are 7701 annotated 3D bounded boxes in total, which roughly equals to 7 labeled cuboids per image. We performed experiments on the complete dataset using 3-fold cross validation. Specifically, for each fold, training is done on 716 images and the testing is performed on the remaining 358 images.

We evaluate the performance on three tasks, including the cuboid detection, clutter/non-clutter estimation and the foreground/background segmentation. The weighting parameters involved in the energy function (Eq. (1)) are learned (details in Sec. 5.2). Other parameters which are involved in shaping the constraints (e.g., α_{obs}, α_{int}) are set to achieve the best performance on a small validation set. This validation set consists of 10 randomly sampled training images in each iteration of 3-fold cross validation.

6.2. Cuboid Detection Task

We first evaluate the cuboid detection task, in which we compute the intersection over union of volumes (Jaccard Index-JI) for the quantitative evaluation. Fig. 4 shows the cuboid detection rate as the threshold for JI is increased from 0 to 1. The overall low detection rate is partially due to the fact that many cuboids for scene structures and major objects (e.g., cupboard) are quite thin and the volumetric overlap measure can be sensitive in such cases. We compare our method with a baseline approach and the state of the art techniques by Jiang et al. [16], Huebner et al. [15] and Truax et al. [28]. The baseline method uses only the unary cuboid costs for detection. Random initializations are chosen for the parameters involved

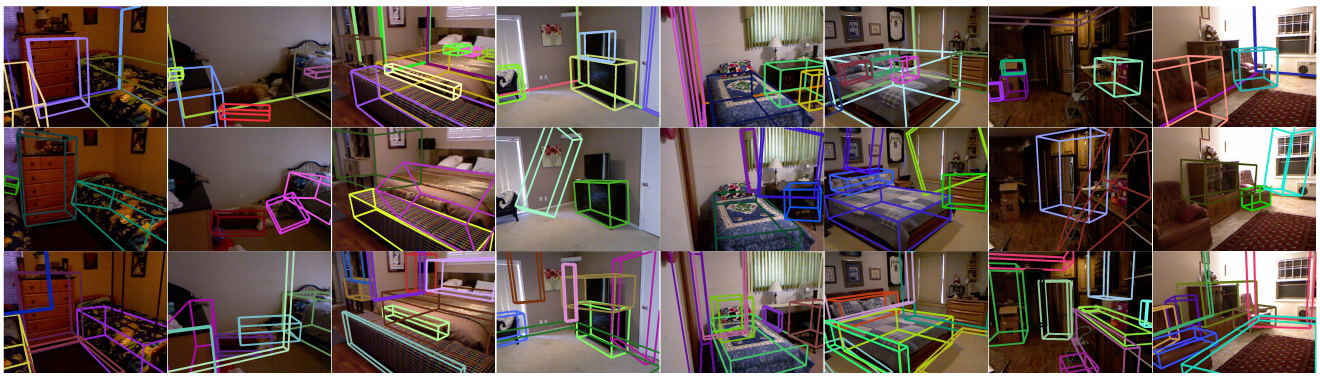


Figure 5: Comparison of our results (*3rd row*) with the state of the art technique[16] (*2nd row*) and Ground Truth (*1st row*). *(Best viewed in color and enlarged)*

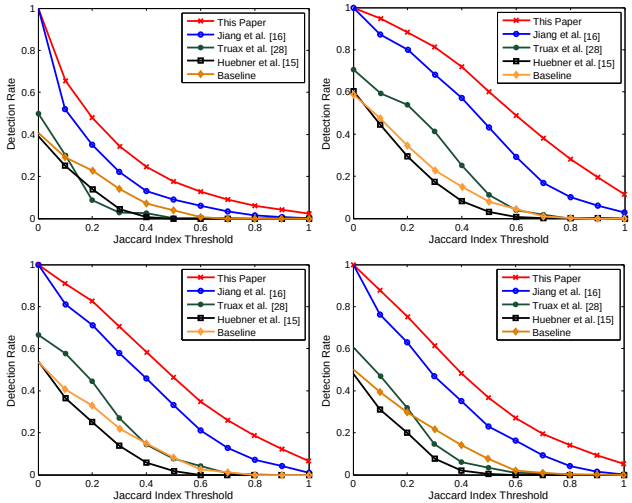


Figure 4: Jaccard Index comparisons for all annotated cuboids (*top left*), for the most salient cuboid (*top right*), for top two salient cuboids (*bottom left*) and top three salient cuboids (*bottom right*).

in [15, 28]. We use the projected area of a cuboid as its saliency measure to rank the ground-truth objects. The results (Fig. 4, Tab. 2) show that the global optimization performs better than the unary scores and the local search techniques [15, 28]. At the 40% JI threshold mark in Fig. 4, we have 31.1%, 26.8%, 38.0% and 89.4% better performances compared to [16] for top one, top two, top three and all cuboids detection tasks respectively. The ablative analysis in Tab. 2 indicates that both the newly introduced features and the joint modeling contribute to the overall improvement in detection accuracy.

Qualitative comparisons are shown in Fig. 5. Our method gives good results on many difficult indoor scenes involving clutter, partial occlusions, appearance and illumination variations. In some cases, ground-truth cuboids are not available for some major objects/structures in the scene, but our technique is able to detect them correctly. We also compare qualitatively with the Jiang et al's method [16], for which the results are generated using the code provided by the authors. It can be seen that our approach performs better in most of the cases.

Method	Accuracy
Unary cuboid cost of Jiang [16]	6.5%
Our unary cuboid cost only	8.8%
Our unary + pairwise cuboid cost only	19.4%
Our full model	26.1%

Table 2: An ablation study on the model potentials/features for the cuboid detection task at the 40% JI threshold.

Method	Precision	Recall	F-Score
Superpixel unary only	$0.43 \pm 13\%$	$0.45 \pm 11\%$	$0.44 \pm 16\%$
Unary + pairwise	$0.46 \pm 12\%$	$0.48 \pm 10\%$	$0.47 \pm 16\%$
Full model (all classes)	$0.65 \pm 9\%$	$0.68 \pm 8\%$	$0.66 \pm 12\%$
Full model (only object classes)	$0.75 \pm 6\%$	$0.71 \pm 8\%$	$0.73 \pm 10\%$

Table 3: Evaluation on Clutter/Non-Clutter Segmentation Task. Precision signifies the accuracy of clutter classification.

Eval. Criterion	CPMC [6]		This Paper	
	Pre.	Rec.	Pre.	Rec.
Most salient obj.	$0.83 \pm 11\%$	0.79 ± 12	$0.85 \pm 15\%$	$0.82 \pm 15\%$
Top 2 salient obj.	$0.77 \pm 12\%$	0.73 ± 14	$0.81 \pm 16\%$	$0.79 \pm 16\%$
Top 3 salient obj.	$0.69 \pm 15\%$	0.66 ± 17	$0.79 \pm 21\%$	$0.76 \pm 19\%$
All objects	$0.54 \pm 17\%$	0.51 ± 20	$0.73 \pm 23\%$	$0.69 \pm 21\%$

Table 4: Evaluation on Foreground/Background Segmentation Task. Precision signifies the accuracy of foreground detection.

6.3. Clutter/Non-Clutter Segmentation Task

To evaluate the clutter segmentation task, we generate the ground-truth clutter labeling based on the cuboid annotation. Specifically, we project the 3D points inside the ground-truth cuboids onto the image plane, and label them as the non-clutter regions while the rest of the regions are clutter. As a baseline, we report the performance when only superpixel unary cost was used for segmentation. The addition of the pairwise cost and the joint modeling results in significant improvement (Tab. 3). We also consider only the object cuboids and compare the performance when scene structure cuboids are excluded from the evaluations.

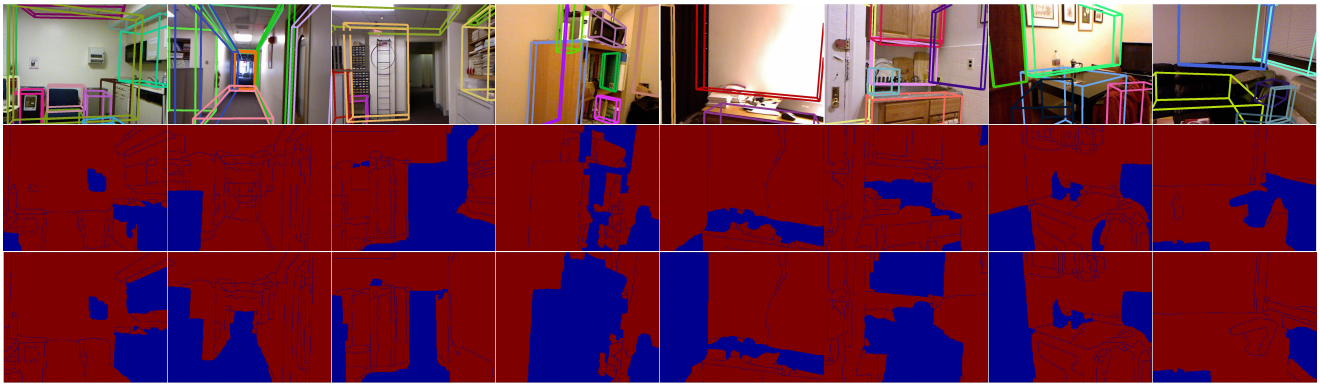


Figure 6: Qualitative Results: Our method is able to accurately detect cuboids in the case of cluttered indoor scenes (1^{st} row). The 2^{nd} and 3^{rd} rows show our clutter labelling and the ground-truth labelling on superpixels, respectively. In the *bottom* two rows, *red* color represents *non-clutter* while *blue* color represents *clutter*. *(Figure best viewed in color and enlarged)*

6.4. Foreground Segmentation Task

We compare our results with the CPMC framework [6] on the foreground/background segmentation task. The objects which are labeled in the dataset are treated as foreground, while the cuboids which model the structures and the unlabeled regions are treated as background. Tab. 4 shows the comparisons for the cases when top most, top two, top three and all object cuboids are detected as foreground. For the case of all detected object cuboids, the top ten foreground masks from the CPMC framework are considered.

6.5. Discussion

The proposed approach can find wide applications in personal robotics, especially for tasks such as indoor navigation and manipulation. A limitation of our approach is its reliance on the initial cuboid generation. Some of the imperfect cuboid detection examples are shown in Fig. 7. For example, our method is not able to propose cuboids for objects when only one side was visible. For the clutter estimation task, our method confuses specular surfaces with cluttered regions due to missing depth values. Also we did not explicitly use constraints such as Manhattan world [9], which may improve the quality of the cuboids aligned with room.

In order to confirm that the detected cluttered regions satisfy our definition (Sec. 3), we report some statistics on the RMRC dataset (Tab. 5). On each detected cluttered region, we fit a cuboid whose base is aligned with the room coordinates. It turns out that the mean volume occupancy and face coverage of all such cuboids is quite low (36% and 44% respectively).

We summarize the run-time statistics of each step involved in our approach. The cuboid hypothesis generation takes $21 \pm 18\%$ sec/img. The feature extraction on cuboids and superpixels take $8 \pm 25\%$ and $97 \pm 33\%$ sec/img respectively. The RF classifier training for terms f_k^{geo}, f_k^{obj} and ψ_u^{sp} take 6.5 sec, 11.2 sec and 2.8 min respectively. The parameter learning algorithm takes $\sim$ 7 hours. The proposed approach is also efficient at test time i.e., $\sim$ 1 sec/image (Tab. 1).

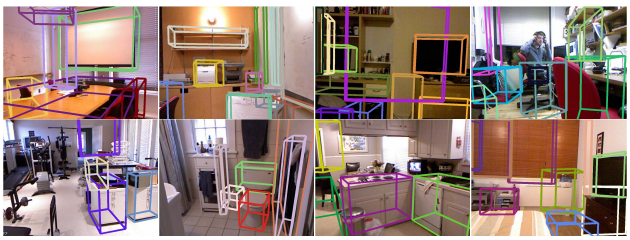


Figure 7: Ambiguous Cases: Examples of detection errors. *(Figure best viewed in color and enlarged)*

Evaluation Criterion	Statistics on RMRC Database
Mean Volume Occupied	$0.36 \pm 19\%$
Mean Coverage along Cuboid Faces	$0.44 \pm 20\%$

Table 5: Statistics for cuboids fitted on cluttered regions.

7. Conclusion

We have studied the problem of cuboid detection and clutter estimation for developing a better holistic understanding of indoor scenes from RGBD images. Our approach jointly models 3D generic objects as cuboids and cluttered regions as local surfaces defined by superpixels. We build a CRF model for all the relevant scene elements, and learn the model parameters based on a structural learning framework. This enables us to incorporate a rich set of appearance and geometric features, as well as meaningful physical and spatial relationships between generic objects. We also derive an efficient inference based on the MILP formulation, and show superior results on cuboid detection and foreground segmentation. In future, we will extend the current work to incorporate useful relationships between semantic classes.

Acknowledgments

This research was supported by the IPRS scholarship from the UWA and the ARC grants DP150104251, DP110103336 and DE120102960. NICTA is funded by the Australian Government as represented by the Dept. of Communications and the ARC through the ICT Centre of Excellence program.

References

[1] R. Achanta, A. Shaji, et al. Slic superpixels compared to state-of-the-art superpixel methods. *TPAMI*, 2012.

[2] P. Arbelaez, M. Maire, et al. Contour detection and hierarchical image segmentation. *TPAMI*, 2011.

[3] M. Bleyer, C. Rhemann, and C. Rother. Extracting 3d scene-consistent object proposals and depth from stereo images. In *ECCV*. Springer, 2012.

[4] L. Bo, X. Ren, and D. Fox. Kernel descriptors for visual recognition. In *NIPS*, 2010.

[5] L. Bo, X. Ren, and D. Fox. Depth kernel descriptors for object recognition. In *IROS*. IEEE, 2011.

[6] J. Carreira and C. Sminchisescu. Cpmc: Automatic object segmentation using constrained parametric min-cuts. *TPAMI*, 2012.

[7] M. Ester, H.-P. Kriegel, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, 1996.

[8] P. F. Felzenszwalb, R. B. Girshick, et al. Object detection with discriminatively trained part-based models. *TPAMI*, 2010.

[9] D. F. Fouhey, A. Gupta, and M. Hebert. Unfolding an indoor origami world. In *ECCV*, pages 687–702. Springer, 2014.

[10] A. Geiger, C. Wojek, and R. Urtasun. Joint 3d estimation of objects and scene layout. In *NIPS*, 2011.

[11] F. Glover and E. Woolsey. Converting the 0-1 polynomial programming problem to a 0-1 linear program. *Operations research*, 22(1):180–182, 1974.

[12] A. Gupta, A. A. Efros, and M. Hebert. Blocks world revisited: Image understanding using qualitative geometry and mechanics. In *ECCV*. Springer, 2010.

[13] M. Hayat, M. Bennamoun, and S. An. Deep reconstruction models for image set classification. *TPAMI*, 37(4):713–727, April 2015.

[14] V. Hedau, D. Hoiem, and D. Forsyth. Recovering the spatial layout of cluttered rooms. In *ICCV*. IEEE, 2009.

[15] K. Huebner, S. Ruthotto, and D. Kragic. Minimum volume bounding box decomposition for shape approximation in robot grasping. In *ICRA*, pages 1628–1633. IEEE, 2008.

[16] H. Jiang and J. Xiao. A linear approach to matching cuboids in rgbd images. In *CVPR*. IEEE, 2013.

[17] T. Joachims, T. Finley, and C.-N. J. Yu. Cutting-plane training of structural svms. *Machine Learning*, 77(1):27–59, 2009.

[18] S. H. Khan, M. Bennamoun, F. Sohel, and R. Togneri. Automatic feature learning for robust shadow detection. In *CVPR*, pages 1939–1946. IEEE, 2014.

[19] S. H. Khan, M. Bennamoun, F. Sohel, and R. Togneri. Geometry driven semantic labeling of indoor scenes. In *ECCV*, pages 679–694. Springer, 2014.

[20] H. S. Koppula, A. Anand, et al. Semantic labeling of 3d point clouds for indoor scenes. In *NIPS*, 2011.

[21] A. H. Land and A. G. Doig. An automatic method of solving discrete programming problems. *Econometrica*, 1960.

[22] D. C. Lee, A. Gupta, et al. Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces. In *NIPS*, 2010.

[23] D. Lin, S. Fidler, and R. Urtasun. Holistic scene understanding for 3d object detection with rgbd cameras. *ICCV*, 2013.

[24] B. Pepik, M. Stark, et al. Teaching 3d geometry to deformable part models. In *CVPR*. IEEE, 2012.

[25] A. G. Schwing, S. Fidler, et al. Box in the box: Joint 3d layout and object reasoning from single images. 2013.

[26] N. Silberman, D. Hoiem, et al. Indoor segmentation and support inference from rgbd images. In *ECCV*. Springer, 2012.

[27] B. Taskar, V. Chatalbashev, and D. Koller. Learning associative markov networks. In *ICML*, page 102. ACM, 2004.

[28] R. Truax, R. Platt, and J. Leonard. Using prioritized relaxations to locate objects in points clouds for manipulation. In *ICRA*, pages 2091–2097. IEEE, 2011.

[29] H. Wang, S. Gould, and D. Roller. Discriminative learning with latent variables for cluttered indoor scene understanding. *Communications of the ACM*, 2013.

[30] Y. Xiang and S. Savarese. Estimating the aspect layout of object categories. In *CVPR*. IEEE, 2012.

[31] J. Xiao, B. C. Russell, and A. Torralba. Localizing 3d cuboids in single-view images. In *NIPS*, 2012.

[32] J. Zhang, C. Kan, et al. Estimating the 3d layout of indoor scenes and its clutter from depth sensors. 2013.

[33] B. Zheng, Y. Zhao, et al. Beyond point clouds: Scene understanding by reasoning geometry and physics. In *CVPR*. IEEE, 2013.