Understanding the 3D Layout of a Cluttered Room From Multiple Images

Sid Yingze Bao, Axel Furlan, Li Fei-Fei, Silvio Savarese
Department of Computer Science, Stanford University
{bao,furlan,feifeili,ssilvio}@stanford.edu

Abstract

We present a novel framework for robustly understanding the geometrical and semantic structure of a cluttered room from a small number of images captured from different viewpoints. The tasks we seek to address include: i) estimating the 3D layout of the room – that is, the 3D configuration of floor, walls and ceiling; ii) identifying and localizing all the foreground objects in the room. We jointly use multiview geometry constraints and image appearance to identify the best room layout configuration. Extensive experimental evaluation demonstrates that our estimation results are more complete and accurate in estimating 3D room structure and recognizing objects than alternative state-of-the-art algorithms. In addition, we show an augmented reality mobile application to highlight the high accuracy of our method, which may be beneficial to many computer vision applications.

1. Introduction

In this paper we will present a new framework for understanding an indoor environment from multiple images. The goal of indoor room understanding involves estimating 3D layout (e.g. floor, walls, ceiling) of the indoor environment as well as identifying the objects within it. Using images to understand the layout of a cluttered room is a great challenge in computer vision research. A room may be occupied by objects that are not necessarily observed in a training set. The room walls may be occluded and cannot be observed directly (Fig. 1). Solving the room layout understanding problem is beneficial in many applications such as augmented reality.

In the past few decades, researchers proposed numerous remarkable methods[8, 21, 7] focusing on obtaining metric reconstructions of an unknown environment. These methods can accurately recover the 3D geometry of an environment given enough quantity of images. However, they cannot identify the key semantic phenomena inside the environment (Fig. 1a). Meanwhile, researchers [2, 26, 20] also looked at estimating scene semantics from 3D points. Nevertheless, these methods usually require very dense and accurate reconstructions obtained using 3D scanners or from a very large number of images. Such requirement limits the scope of their applications. Moreover, [23, 4, 6] leverage the Manhattan world assumption to estimate room walls from a sequence of images, but they cannot handle objects in a scene (Fig. 1b).

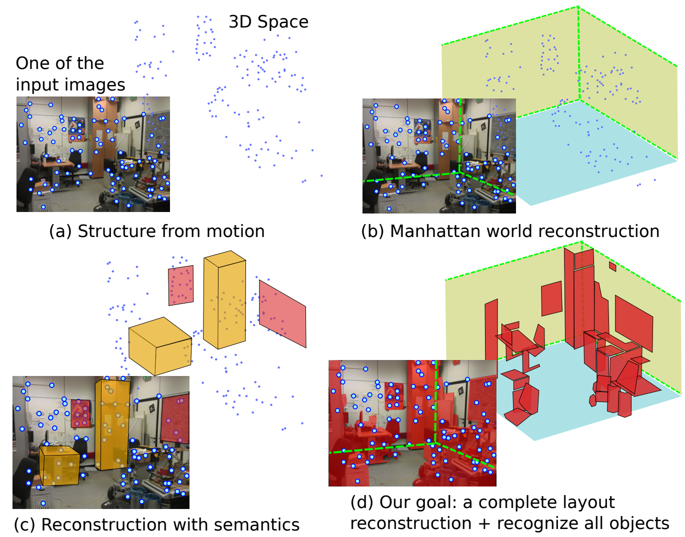


Figure 1: Understanding a cluttered room from a few images. (a) Structure from motion techniques (e.g. [21]) can only understand the geometry of the room as a sparse set of 3D points. (b) Layout estimation methods (e.g. [23, 4]) may recover the wall structure without reasoning about objects. (c) Joint geometric and semantic reconstruction methods (e.g. [1]) can recognize a few objects (the yellow boxes), estimate their positions in 3D, as well as estimate the 3D layout as a sparse set of elements (the red regions). (d) Our goal is to estimate the complete 3D layout of the room (floor, walls, ceiling) and identify all the foreground objects. Notice that we aim at differentiating objects v.s. walls, rather than distinguishing different object entities / categories.

Recently, [1] proposed an approach to jointly estimating the geometric and semantic properties of a scene. Using a small set of images, [1] shows better 3D geometry estimation and object recognition results than the geometry estimation methods or the semantic reasoning methods that work in isolation. Unfortunately, one of its shortcomings is that it can only produce a very sparse reconstruction of a scene (Fig. 1c), which is not desirable for the aforementioned applications.

Another noticeable series of works concentrate on parsing the room layout from a single image [9, 10, 11, 16, 15, 18,

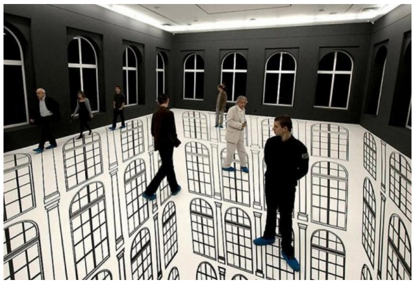


Figure 2: Using a single image to understand room layout may suffer from the intrinsic ambiguity of a single image. This photo may be interpreted in two ways: 1) the floor is painted artificially to create the illusion; 2) the room is hollow and the people are floating. If we are given another photo from a different view point, this ambiguity will naturally dissolve.

19, 25, 13, 5, 3]. However, their accuracy in estimating the 3D scene layout is limited mostly due to the fact that 3D perception from a single view is essentially an ill-posed problem, and the room structure may not be uniquely inferred from a single image. An illustrative example is shown in Fig. 2.

Understanding the room layout from multiple images is far from being trivial. We need an effective and efficient algorithm to jointly reason about the content in multiple images. On the other hand, although we can infer certain 3D geometry information, e.g. structure-from-motion (SFM) points, to help room layout estimation, the 3D cues inferred from a few input images are usually very sparse and noisy. Experiment results proved that simply relying on the SFM points from a small set of images (~10) will yield very unstable and inaccurate layout estimation results. In order to address these challenges, we propose a new room understanding framework bearing the following contributions.

Accuracy. We can achieve higher accuracy in layout estimation and object recognition tasks than pure geometry-based methods or single-image methods. We estimate 3D room layout (walls, floor, ceiling) jointly using geometric and semantic cues, which play complementary roles in helping recover the geometry of the scene. When a room is very cluttered, there will usually exist a large set of characteristic feature points, which can yield a SFM point cloud with reasonable density (geometric cue). SFM points can help us reason about the extent of the room and thereby tackle the adversary that wall boundaries are occluded by the foreground objects. As the opposite, when a room is comparatively clean, we can exploit image line segments and region segmentation results (semantic cue) to obtain a good estimation of the room's walls. Meanwhile, by jointly using multiple images to reason about the existence of objects, our object recognition accuracy can be demonstrated to be significantly higher than single-image methods.

Completeness. We seek for a complete reconstruction of the room layout in 3D including objects. In contrast, many aforementioned methods can only reconstruct the room layout as a set of points [8, 21] or a sparse set of regions [1]. Moreover, different from many previous works [16, 10, 18] that only consider box-like objects, our model can accommodate objects with more complex shapes. We propose a surface-based object representation (Fig. 5), which greatly expands the types of recognizable objects compared to a box-based representation. Notice that, our goal is to recognize objects apart from room layouts, rather than recognizing object categories. Compared to recognizing an object as a whole (as in [16, 10, 18]), our surface-based representation also enhances the chance of recognizing an unknown object (an object appears in testing but not in the training set) by using parts (surfaces) that are shared by other objects in the training set. For example, a wooden desk may share similar texture and legs as a wooden chair. Hence, even if our training set does not contain the desk category, the desk may still be successfully recognized (as an object) provided that the training set contains a chair with similar parts and texture. Notice that, we use a generic object segmentation algorithm (Sec. 2.1) to decompose objects into surfaces, rather than using a pre-trained model for each object category.

We conducted numerous experiments using a novel dataset containing 50 various room scenes with 10 images in each scene. Various experiments demonstrate that our framework can achieve better estimation accuracy and higher reconstruction completeness than alternative state-of-the-art approaches. At last, we will show an Android application which leverages our layout estimation method to achieve pleasant augmented reality results.

2. Problem Definition

2.1. Inputs and Measurements

We are provided a total number of N unordered images $I^1 \cdots I^N$ (Fig. 3a). In each image I^i we can detect a set of feature points (e.g. [17]) $\mathbf{p}^i$, as well as a set of segmented regions $\mathbf{b}^i$ (Fig. 3c and 3d). In the following text we will also refer to line segments in images. Line segments are essentially the boundaries of regions. For the sake of simplicity, we do not introduce additional symbols for line segments.

The feature points play a number of different roles in our framework. One role is to create the 3D reconstruction of the points in rooms and help estimate camera parameters. Since the target scenario of our algorithm is a cluttered room, we can assume that the input images contain a sufficient amount of feature points to be matched across each other, and therefore a structure-from-motion (SFM) pipeline can be used to estimate a set of 3D points $\mathbf{P}$ in the scene, as well as the camera parameters $\mathbf{C}$ (Fig. 3b). Let $\mathbf{C} = \{C^i\}$ be the camera parameters where C^i indicates the rotation, translation, and intrinsics of image I^i. The extrinsics are estimated using a SFM pipeline (e.g. [21]), while the intrinsics may be provided as input or estimated using auto-calibration [8].

The region segments are critical to our framework for eval-

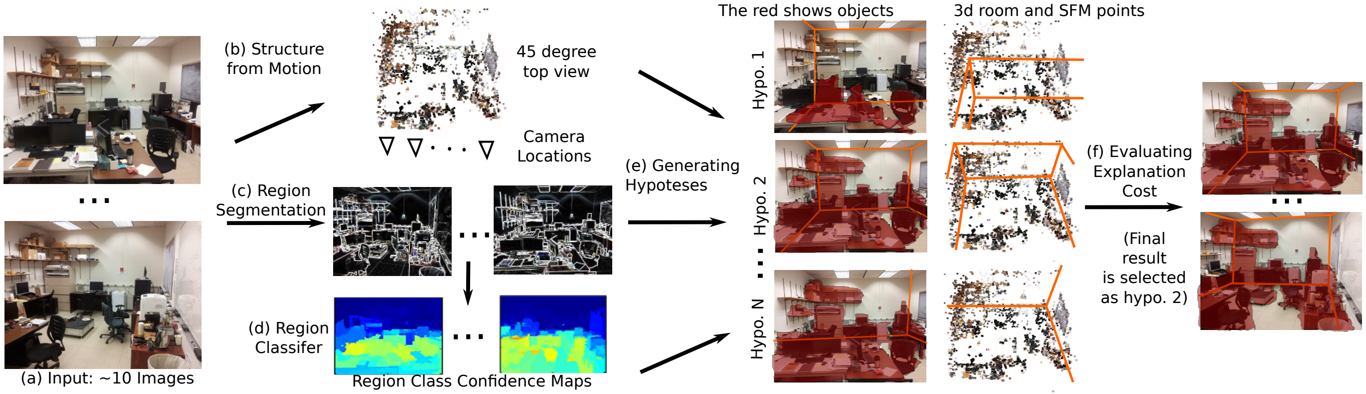
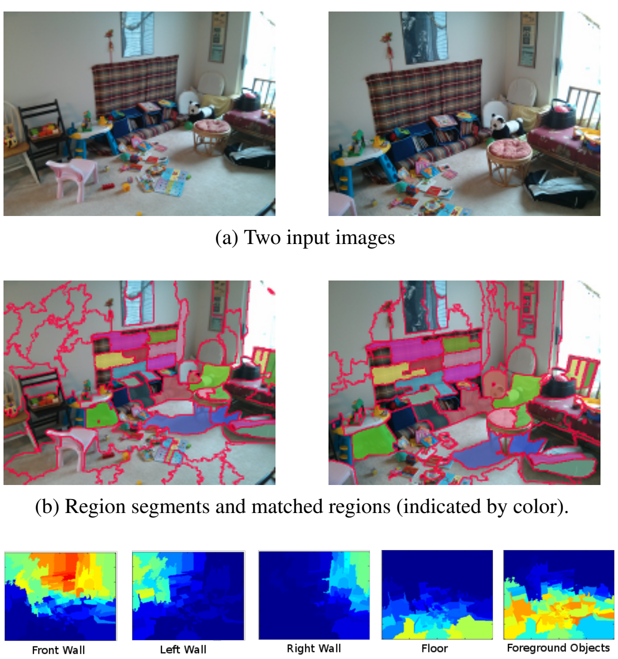


Figure 3: Multi-image room layout understanding framework.

uating the possibility of a room hypothesis (layout + objects). Since our framework is designed to use multiple input images, the region segments should be matched across images. We apply a multi-image segmentation algorithm (e.g. [22]), which not only automatically matches across-image regions covering the same objects (see colored regions in Fig. 4b), but also simultaneously guarantees that the matched regions are similar in shape and appearance (see region shapes in Fig. 4b). The k^{th} region segment in image I^i is denoted as b_k^i, whose appearance can be described by a vector concatenating multiple cues (e.g. cues proposed by [12]). Given the appearance vector and a pre-trained region classifier, a confidence can be calculated that b_k^i belongs to class label l (e.g. walls, floors, ceilings, or other objects). See Fig. 4c for examples.



(a) Two input images

(b) Region segments and matched regions (indicated by color).

(c) Response map of region classifiers. (The left image in the image pair)

Figure 4: Co-segmentation. This example shows the result of using two images. In our experiment the co-segmentation is applied to ~10 images of the same room.

2.2. Unknown Parameters

The unknowns are the 3D layout and the configuration of objects in the room.

The 3D layout can be described by a set of room surfaces (walls, floors, ceilings) $\mathbf{S} = \{S_1 \cdots S_{N_S}\}$. A surface S_i is parametrized by its centroid, orientation, and extent in 3D. In our experiments, we follow previous works [9, 15, 18] which hold the assumption that the room layout is a 3D box. See orange lines in Fig. 3e and 3f.

We model 3D objects as a set of 3D planar surfaces (we also refer to as regions). See Fig. 5 for examples. In our framework, we do not model objects as single entities. Instead, we assign to each surface a single class label which is *object* v.s. *non-object*. Non-object means that a surface belongs to the room layout which can be further classified into floor, wall, or ceiling. Object means that a surface belongs to one of the foreground objects (though we do not distinguish which one). Let $\mathbf{O} = \{O_1 \cdots\cdots O_{N_o}\}$ represent the collection of all objects in a room environment, where O_i is a planar 3D surface which belongs to an object in the scene. A surface O_i captures the location, orientation, and extent of a component of an object in 3D. Although such modeling approximates every surface as flat, it allows to accommodate arbitrarily complicated object configurations.

3. Model Formulation

Our goal is to estimate a room layout $\mathbf{R} = \{\mathbf{S}, \mathbf{O}\}$ from measurements by minimizing a cost function E:

$$\mathbf{R} = \arg\min_{\mathbf{R}} E(\mathbf{R}; \mathbf{P}, \mathbf{C}, \mathbf{b}) \quad (1)$$

where E evaluates the likelihood of $\mathbf{R}$ given SFM points $\mathbf{P}$, estimated camera parameters $\mathbf{C}$, and region measurements in every image $\mathbf{b} = \{\mathbf{b}^i\}$. In order to compute the cost of a given hypothesis with respect to the measurements, we consider the cost of their geometric compatibility in 3D space (E_G) and the cost of semantic interpretation in images (E_M):

$$E(\mathbf{R}; \mathbf{P}, \mathbf{C}, \mathbf{b}) = E_G(\mathbf{R}; \mathbf{P}) + E_M(\mathbf{R}; \mathbf{C}, \mathbf{b}) \quad (2)$$

(a) Objects in a scene (b) Surface Decomposition

Figure 5: Object Representation. (a) Objects in images. These two objects cannot be effectively represented using bounding cubes as proposed by [10, 16, 18] (b) One possible region decomposition for the objects. In our experiments, the decomposition is the result generated from region segmentation algorithm (e.g. [22]), not from a pre-trained model. The 3D locations and orientations of object surfaces are estimated using the SFM points attached to the surfaces.

Notice that, if only one image is given, we cannot evaluate 3D geometry cost, the overall cost degenerates into evaluating semantic cost in a single image, which is related to the energy function proposed in most single-image methods [12, 9].

3.1. Geometric Cost

A good layout estimation should be compatible with the estimated SFM points. However, the criteria of evaluating such compatibility should be carefully selected since SFM points may contain many outliers and only sparsely represent the 3D layout of a room. We use the following criteria to calculate E_G:

- The inner space enclosed by $\mathbf{S}$ should contain all scene points $\mathbf{P}$. Let $\Omega(\mathbf{P};\mathbf{S})$ be the function computing the percentage of points in $\mathbf{P}$ not enclosed by $\mathbf{S}$. The cost of the points excluded from a room structure can be computed as $E_G^I = \Omega^2(\mathbf{P};\mathbf{S})/\sigma_\Omega^2$.

- The 3D walls / floors/ ceilings defined by $\mathbf{S}$ should be supported by points in $\mathbf{P}$. S_i is the i^{th} 3d surface in $\mathbf{S}$. Let $\tau_i^S \subset \mathbf{P}$ be the indices of the 3D points whose image projections fall into the image projection of S_i excluding the part occluded by object surfaces in $\mathbf{O}$. Denote by $\Lambda(S_i, p_j)$ the function computing the 3D distance from S_i to 3D point p_j. The cost of unsupported 3D walls can be computed as $E_G^S = \sum_i \sum_{j \in \tau_i^S} \Lambda^2(S_i, p_j)/\sigma_S^2$.

- Similarly, the 3D objects (i.e. a set of 3D regions) should also be supported by points in $\mathbf{P}$. Let $\tau_i^O \subset \mathbf{P}$ be the indices of the 3D points whose image projections fall into the image projection of O_i. The cost of unsupported 3D objects can be written as $E_G^O = \sum_i \sum_{j \in \tau_i^O} \Lambda^2(O_i, p_j)/\sigma_O^2$.

The overall geometric cost is a summation:$E_G = E_G^I + E_G^S + E_G^O$. The variance terms $\sigma_\Omega, \sigma_S, \sigma_O$ are learned using a max-likelihood approach.

3.2. Semantic Cost

The sophisticated content carried by images can be used to verify the possibility of a room hypothesis. We project $\mathbf{O}$ and $\mathbf{S}$ into each image using the estimated camera parameters $\mathbf{C}$. Denote by s_i^k / o_j^k the image projection of $S_i \in \mathbf{S}$ / $O_j \in \mathbf{O}$ in the k^{th} image considering their occlusion relationships. Once S_i or O_j is projected into an image, we can transfer their labels to corresponding image regions. The possible labels include left wall, front wall, right wall, ceiling, floor, objects. Correct 3D layout will lead to labels reinforced by image evidence. We use segmented regions to check the likelihood that a projection with certain inferred label is correct. A projected 3D region (s_i^k / o_j^k) may overlap with a number of image regions (object region only overlaps with one). Denote by θ_i^k the indices of the elements in $\mathbf{b}^k$ (regions in the k^{th} image) which overlap with s_i^k. The semantic cost for one projected wall region in image I^k can be computed as:

$$E_M^k(S_i) = -\frac{1}{|\theta_i^k|} \sum_{j \in \theta_i^k} c(b_j^k \in l_i^k)$$

where $c(\cdot)$ is the label confidence function defined by a classifier learned from a training set. The semantic cost for object regions $E_M^k(O_i)$ can be easily written in a similar fashion. Given multiple images and all the elements in $\mathbf{S}$ and $\mathbf{O}$, the semantic cost can be written as:

$$E_M(\mathbf{R};\mathbf{C},\mathbf{b}) = \sum_k (\sum_i E_M^k(S_i) + \sum_j E_M^k(O_j))$$

4. Solving the Estimation Problem

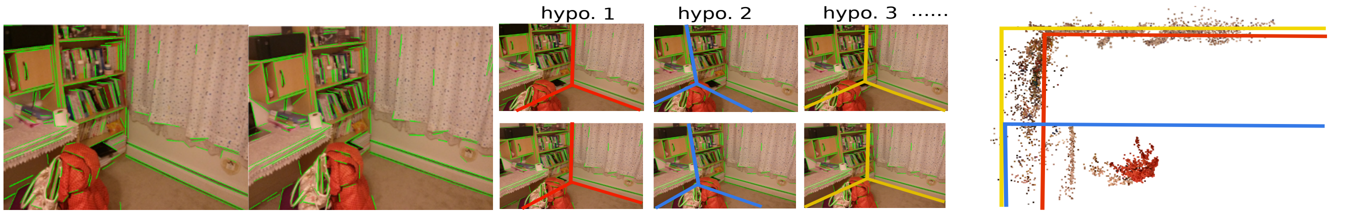
We solve the room estimation problem by identifying the room layout $\mathbf{S}$ and objects $\mathbf{O}$ minimizing Eq. 1. Due to the high dimensionality of the unknown parameter space, we adopt an approach that is based on proposing hypotheses and evaluating them using the cost function E (Eq. 2). We first propose a set of hypotheses $\{\mathbf{R}_n\}$ (Sec. 4.1), and next identify among these proposals the best layout configuration which yields the minimum cost (Sec. 4.2). Our framework can be summarized as the flowchart shown in Fig. 3.

4.1. Generating Hypotheses

Effectively proposing room hypotheses is the key to this estimation process. The room proposal process consists of four steps.

4.1.1 Estimating Dominant Directions

We adopt the Manhattan world assumption that the walls of a room must be perpendicular to one of three mutually perpendicular directions (dominant directions). We adopt [15] to estimate dominant directions from the line segments (e.g. boundaries of regions or detected using methods such as [24]) in each input image. The dominant direction in the world



(a) Input images and detected line segments (b) Three wall hypotheses among many proposals. (c) SFM points and hypotheses in 3D (top view)

Figure 6: Proposing wall layout candidates. (a) Detected line segments. Line segments allow to estimate the dominant direction of the room. (b) Wall layout candidates are generated by enumerating pairs of line segments. (c) Triangulation of 2D wall layouts provides their configurations in 3D. By comparing with the SFM points, it is easy to see only hypothesis 3 (the yellow one) is compatible with the SFM points. Notice that most single-image methods suffer from accurately choosing the best layout among the three candidates shown in (b).

coordinate system can be calculated by averaging the dominant directions in all images considering their relative camera poses.

4.1.2 Triangulating Room Corners

In order to generate room hypotheses, we first estimate a set of possible 3d locations of room corners. A room corner is a 3D point where three walls intersect. A room's layout can be defined by its corners. In order to locate room corners in 3D space, we first identify them in each image (Fig. 6b), and use estimated camera poses to triangulate their 3D position (Fig. 6c). This is not a trivial task since wall corners may not be directly observable, due to occlusion or weak corner detector response. We leverage on line segments to infer the existence and locations of room corners in an image (Fig. 6a). Given the estimated dominant directions, each line segment can be labeled as "bottom-up", "left-right", "back-front", or "random". Two different types (except random lines) of line segments may intersect and form a corner. In one image, by pairing line segments and inferring their 2D intersections, we can obtain a (large) set of image points among which a few represent true room corners. We can obtain the 3D location of a room corner candidates q_i by triangulating a pair of image corner candidates that satisfy epipolar constraint. Triangulating every pair of 2D corner candidates may generate a very large set of 3D points $\mathbf{Q} = \{q_i\}$, among which only a few are true room corners.

4.1.3 Generating Room Hypotheses

Room hypotheses are generated from the corner candidate set $\mathbf{Q}$. In our experiments, we assume a room layout is a cuboid, hence a layout hypothesis can be uniquely proposed using a number of corners. We randomly sample points in $\mathbf{Q}$ and obtain the set of room layout hypotheses. In order to confine the total number of layout hypotheses within a tractable range (at most 300 in our experiments), we use K-means algorithm to cluster similar room layout and only keep significantly different room layout hypotheses as $\{\mathbf{S}_l\}$. Please see supplementary materials for more details.

4.1.4 Generating Object Hypotheses

After a layout hypothesis $\mathbf{S}_l$ is generated, we next generate its compatible object configuration $\mathbf{O}_l$. In order to minimize the overall cost, the object hypotheses are generated from two clues: 1) 3D SFM points that are not close to the room walls (to minimize E_G^O), 2) image regions that are assigned with a high score by object classifier (to minimize E_M^k). Please see Fig. 7 and its caption for the details regarding generating object hypotheses. In our experiments, we find these two types of clues complimentary. An object (e.g. a book with unique cover) may not share similar appearance with other objects in a training set, and therefore an appearance-based classifier may fail to detect it. However, its triangulated 3D location can help infer that it does not belong to rooms walls (hence it must be an object). On the other hand, an object (e.g. a table surface) may have simple and clean appearance which does not carry sufficient features for SFM, but it's simple appearance pattern may be easily recognized by a classifier. A room layout hypothesis $\mathbf{S}_l$ and its corresponding object hypotheses $\mathbf{O}_l$ constitute a room hypothesis $\mathbf{R}_l$.

4.2. Evaluating Hypotheses

Given a layout hypothesis $\mathbf{R}_l$, we can evaluate its cost as $e_l = E(\mathbf{R}_l; \mathbf{P}, \mathbf{C}, \mathbf{b})$. The final estimation of the room layout is obtained by selecting the hypothesis with the lowest cost. In our experiments, we exploit parallel computing technique to efficiently evaluate all layout hypotheses. As our future work, we will adopt faster inference algorithm such as branch-and-bound [19] to accelerate the hypotheses generation and evaluation process.

5. Evaluation

We conduct experiments in a novel dataset which contains 50 different room scenes each of which include 10 images. We would like to release this new kind of multi-image dataset to the community for future research. Example figures and results are shown in Fig. 9. Using this dataset, we compare our method against other state-of-the-art methods. Since our proposed method requires multiple images, we cannot evaluate on single image datasets such as the one proposed in [9]. However, we use the labeled data in [9] to train region clas-

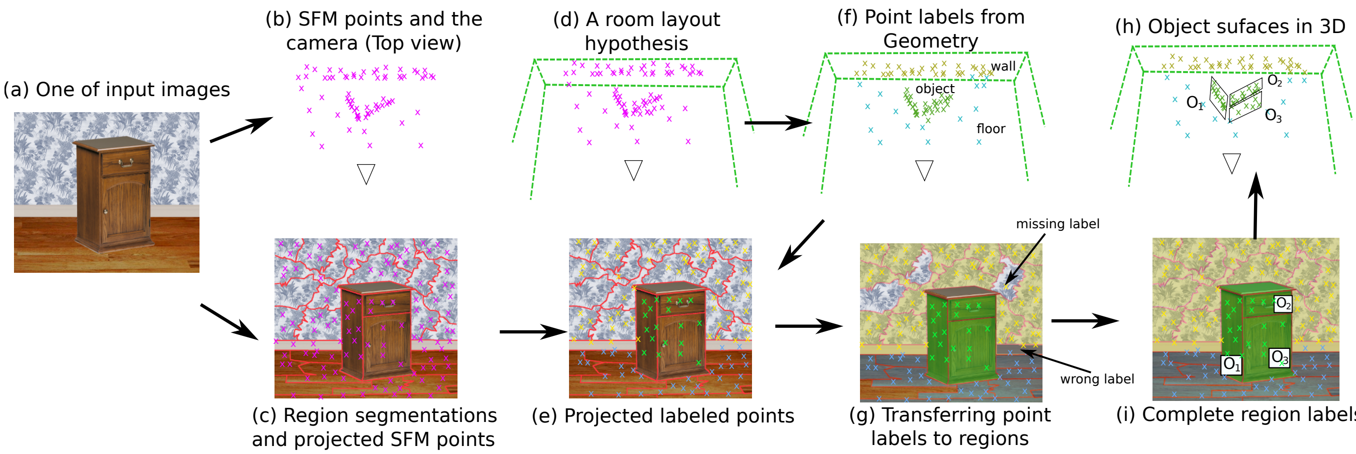


Figure 7: Generating object hypotheses. (a) an example image from a set of input images. (b) Top view of the SFM points and the camera (the triangle). (c) Region segments in this image and the location of projected SFM points. (d) A given room hypothesis (Sec. 4.1.3) overlaid with SFM points. (f) We can identify the points close to the walls and assign labels to SFM points (yellow and blue). The points that do not belong to any walls will be labeled as non-room (green) i.e. objects. (e)(g) the SFM point labels can be transferred to regions. Notice that there are missing (transparent) or wrong region labels, since regions may not carry sufficient SFM points and point labels may be noisy. (i) Based on the labels initialized from SFM points, we obtain a complete region classification by minimizing E_M. Notice that the missing labels are inferred and the wrong labels are corrected by enforcing appearance consistency. (h) The region labeled as objects can be back-projected into 3D space if they carry sufficient SFM points. In this case, we can generate an object configuration hypothesis containing O_1,O_2, and O_3. Notice that the top part of the cabinet does not correspond to an object surface in 3D since it does not carry SFM points. For such surfaces of objects, our framework can infer their existence in images but not in 3D.

	Feat.	SFM	Img. Seg.	Hypo.	Total
Sec.	8.2	2.5	9.3	36.8	56.6

Table 1: Average time consumption for estimating one scene from 10 images. **Feat.** includes sift [17] feature detection (CUDA) and feature matching(CUDA). **SFM** includes ransac-based essential matrix estimation (C), and bundle adjustment (C). **Img. Seg.** indicates image segmentation including superpixel generation (C) and classification (multi-thread Matlab). **Hypo.** indicates hypotheses generation (multi-thread Matlab) and evaluation (Matlab+C) process described in Sec. 4.1 and Sec. 3 .

	[21]	[1]	Ours
Objects	1.2%	77.5%	**86.0%**
All	0.69%	46.0%	**91.4%**

Table 2: 3D reconstruction completeness. The numbers are the percentage of image pixels whose 3D information can be estimated. Objects: only count the pixels belonging to non-wall objects. All: count every pixel. Notice that our completeness is not 100%, because we cannot recover the 3D location of the object surfaces that do not contain SFM points.

sifiers for our method and competing methods. At the end of this section, we will discuss an Android application that leverages our estimation results for indoor augmented reality.

5.1. Algorithm Speed

We report the time consumption of each step in our framework in Tab. 1. Our unpolished implementation mixes the usage of matlab, C, and CUDA. The implementation detail is listed in the table caption. The experiment is conducted on a 4-core 2.8GHz CPU.

5.2. 3D Reconstruction Completeness

We show the 3D reconstruction completeness in Tab. 2. Our model aims at estimating the 3D information of every pixel in an image. In contrast, many alternative room reconstruction methods can only recover the 3D information for a set of points (e.g. [21]) or a set of regions + points (e.g. [1]). [21, 1] both show higher completeness level in reconstructing non-wall objects than reconstructing both objects and walls. The reason is that [21, 1] rely on matched features (points / regions) to create 3D elements. Non-wall objects usually carry more features than walls, and therefore they are more likely to be reconstructed than walls. Notice that our method does not suffer from this condition in that we can infer the existence of walls even if they are not directly observable.

5.3. Layout Estimation Accuracy

In order to evaluate the accuracy for estimating room layout, we adopt the criterion commonly used in other works [9, 19]. We project the estimated room layout into each image, and label every pixel into wall, ceiling, or floor). The percentage of correctly labeled pixels is shown in Tab. 3. Due to the code unavailability of other works, we cannot evaluate them in our dataset. We also compare with a baseline geometry-based approach (Plane Fitting in Tab 3), which uses vanishing lines to estimate dominant directions and uses RANSAC to fit a box-like room based on SFM points. This approach is equivalent to a degenerated version of our method which only minimizes the geometry cost term.

	Home	Office	Other	Overall
Image#	300	110	90	500
[9]	79.8	79.0	81.5	79.9
[15]	73.5	67.7	71.7	71.9
Plane Fitting	71.6	76.0	68.4	72.0
Ours	**92.7**	**96.7**	**92.3**	**93.5**

Table 3: Room layout estimation accuracy. The number is percentage number averaged on 500 images in our dataset.

	[12]	[9]	No Coseg.	No Geo.	Full
Precision	38.8%	52.2%	38.8%	42.1%	**58.1%**
Recall	50.0%	55.4%	50.0%	52.2%	**59.0%**

Table 4: Object Estimation Accuracy. We provide ground truth labels (objects / walls) to segments in images. The precision is the percentage of images pixels that can be correctly classified. The recall is the percentage of correctly-identified pixels that belong to objects. **No Coseg.** indicates the estimation only by maximizing semantic term E_M based on each image independently, which results to identical performance as [12]. **No Geo.** indicates maximizing E_M based on the result of co-segmentation. **Full** is our full model that maximizes $E_G + E_M$ based on the result of SFM and co-segmentation.

5.4. Object Estimation Accuracy

Our proposed framework can estimate non-wall objects in 3D space and in 2D images. We show example estimations in Fig. 9. We evaluate the accuracy of detecting object regions in images. The accuracy for estimating objects can by evaluated by examining every pixel label against ground truth. The result is shown in Tab. 4. Our proposed method shows significant advantage over rival methods or alternative designs of the pipeline, since it can effectively use multiple images which carry greater information than only a single image. Notice that, the same training set is provided to different testing methods.

5.5. Augmented Reality Application

Our method can robustly estimate the room layout from a small number of images, which clears the way toward building many new applications. We developed a cellphone application (the app) using a Nexus 4 android phone and a server. A user uses the app to take photos in a room (e.g. the room in Fig. 8). The app continuously captures new images as the user explores new view points. At the same time, the captured images will be uploaded onto the server via wireless. As more images are being uploaded, the server runs feature detection and matching. When enough features are found and matched, the server will inform the user to stop taking more images. Next, the server will run our layout estimation pipeline. After the layout estimation is finished (usually less than a minute), the user will be able to display new virtual objects (Fig. 8) in the already captured images. Given the estimated layout, the poses of virtual objects are precisely consistent with the actual room layout. Our augmented reality app is markerless, thus

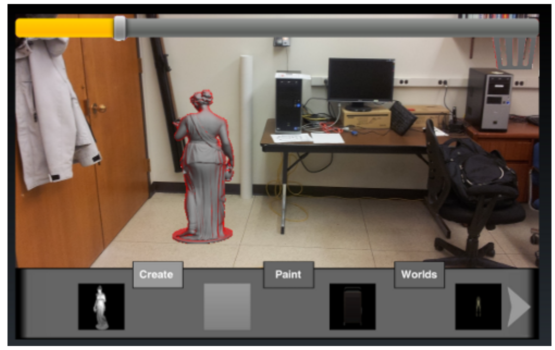


Figure 8: Interface of the mobile augmented reality application. The virtual object (female statue) is automatically placed with the same orientation as the real floor. At the bottom of the screen is a list of virtual objects to select. At the top of the screen is a scroll bar for switching between different images that are already captured.

applicable in more circumstances compared to marker-based augmented reality [14].

6. Conclusion

In this paper, we proposed a multiview framework to solve the cluttered room understanding problem. Our solution can be executed efficiently using a standard computer system. Experiment results demonstrate that our method produces more complete and accurate result in estimating room layout and foreground objects than alternative state-of-the-art methods. A mobile phone application is given to demonstrate our superior estimation results have great potential to enable new markerless augmented reality applications.

Acknowledgments

We acknowledge the support of a NSF CAREER award N.1054127 and a gift award from HTC.

References

[1] S. Y. Bao, M. Bagra, Y.-W. Chao, and S. Savarese. Semantic structure from motion with points, regions, and objects. In *CVPR*, 2012. 1, 2, 6

[2] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. In *ECCV*, 2008. 1

[3] W. Choi, Y. Chao, C. Pantofaru, and S. Savarese. Understanding indoor scenes using 3d geometric phrases. In *CVPR*, 2013. 2

[4] A. Flint, D. Murray, and I. Reid. Manhattan scene understanding using monocular, stereo, and 3d features. In *ICCV*, 2011. 1

[5] D. F. Fouhey, V. Delaitre, A. Gupta, A. A. Efros, I. Laptev, and J. Sivic. People watching: Human actions as a cue for single-view geometry. In *ECCV*, 2012. 2

[6] A. Furlan, D. Miller, D. G. Sorrenti, L. Fei-Fei, and S. Savarese. Free your camera: 3d indoor scene understanding from arbitrary camera motion. In *BMVC*, 2013. 1

[7] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski. Reconstructing building interiors from images. In *ICCV*, 2009. 1

[8] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000. 1, 2

[9] V. Hedau, D. Hoiem, and D. Forsyth. Recovering the spatial layout of cluttered rooms. In *CVPR*, 2009. 2, 3, 4, 5, 6, 7, 8

[10] V. Hedau, D. Hoiem, and D. Forsyth. Thinking inside the box: Using appearance models and context based on room geometry. In *ECCV*, 2010. 2, 4

[11] V. Hedau, D. Hoiem, and D. Forsyth. Recovering free space of indoor scenes from a single image. In *CVPR*, 2012. 2

[12] D. Hoiem, A. Efros, and M. Hebert. Recovering surface layout from an image. *IJCV*, 2007. 3, 4, 7

[13] Y. Jiang, M. Lim, and A. Saxena. Learning object arrangements in 3d scenes using human context. In *ICML*, 2012. 2

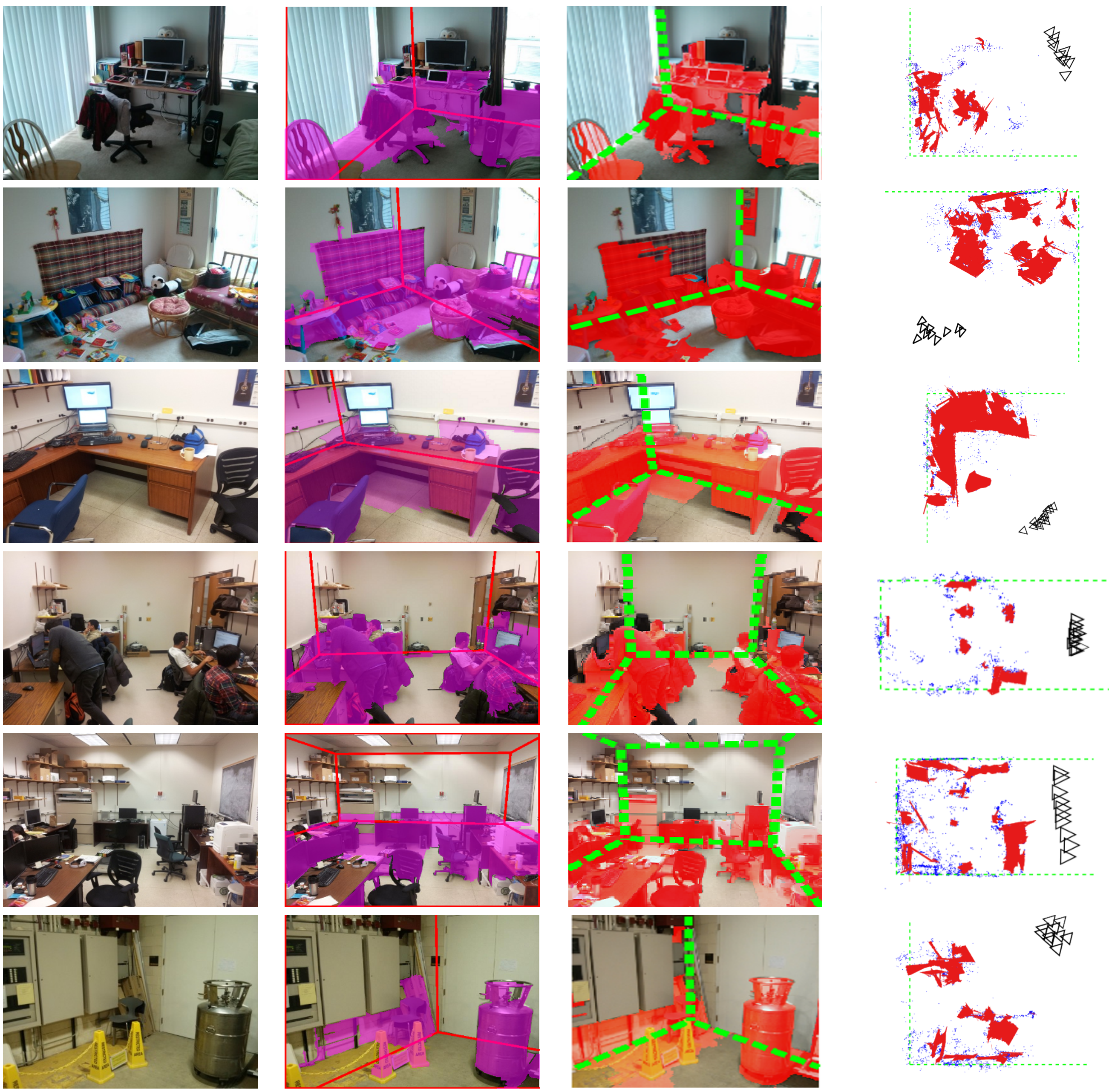


Figure 9: Example results. First column: one of the 10 input images of the scene. Second column: result of [9]. The pink region is recognized as objects. The red lines show the estimated room layout. Third column: our result. The red region is recognized as objects. The green lines indicate the estimated room. Fourth column: floor occupancy map shown from the top view of the scene. The green dashed lines show the extent of the room. The blue points are SFM points. The red show regions in 3D. The triangles visualize the camera locations.

[14] H. Kato and M. Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Int. Workshop on Augmented Reality*, 1999. 7

[15] D. Lee, M. Hebert, and T. Kanade. Geometric reasoning for single image structure recovery. In *CVPR*, 2009. 2, 3, 4, 7

[16] D. C. Lee, A. Gupta, M. Hebert, and T. Kanade. Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces. In *NIPS*, 2010. 2, 4

[17] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 2, 6

[18] L. D. Pero, J. Bowdish, D. Fried, B. Kermgard, E. L. Hartley, and K. Barnard. Bayesian geometric modeling of indoor scenes. In *CVPR*, 2012. 2, 3, 4

[19] A. Schwing and R. Urtasun. Efficient exact inference for 3d indoor scene understanding. In *ECCV*, 2012. 2, 5, 6

[20] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 1

[21] N. Snavely, S. M. Seitz, and R. S. Szeliski. Modeling the world from internet photo collections. *IJCV*, 2008. 1, 2, 6

[22] A. Toshev, J. Shi, and K. Daniilidis. Image matching via saliency region correspondences. In *CVPR*, 2007. 3, 4

[23] G. Tsai, C. Xu, J. Liu, and B. Kuipers. Real-time indoor scene understanding using bayesian filtering with motion cues. In *ICCV*, 2011. 1

[24] R. G. von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall. LSD: a Line Segment Detector. *Image Processing On Line*, 2012. 4

[25] H. Wang, S. Gould, and D. Koller. Discriminative learning with latent variables for cluttered indoor scene understanding. 2010. ECCV. 2

[26] J. Xiao and Y. Furukawa. Reconstructing the world's museums. In *ECCV*, 2012. 1