Behavior Recovery/Analysis in Cluttered and Dynamic Indoor Scene

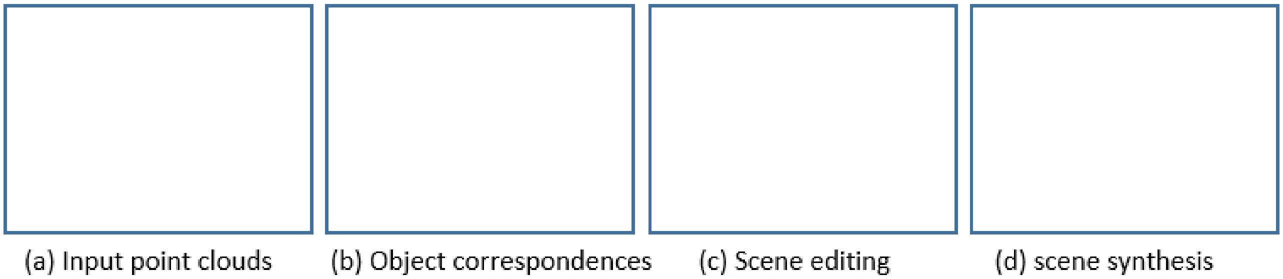

Figure 1: *Behavior recovery and behavior-based editing of dynamic cluttered indoor scenes. From a set of dense scans at different times (a), our system first extract the object correspondences (b) and behavior model ((xuejin: optionally show a graph)). The recovered behavior model can be applied to many applications, such as scene editing (c) and indoor scene synthesis (d).*

Abstract

While modeling static indoor scenes using RGBD cameras has been extensively studied in recent years, we introduce a *behavior recovery* system to investigate the behavior of objects in an cluttered indoor scene. In a daily indoor environment, because of object functions and human behaviors, the spatial placements of objects presents non-unique but statistically regular displacements. We take the *inverse problem* to recover object behaviors from a collection of point clouds captured at different times in a dynamic indoor scene. Our system consists of two key parts, *extraction of object correspondence* and *behavior model*. Given a collection of dense point clouds of an indoor scene in daily use at different times, the correspondence between objects are extracted using an iterative segmentation-and-registration process. Our algorithm is robust to noise and incomplete parts in imperfect scans. In the second part is to recover object behaviors, which represent the spatial arrangement of objects and interrelations between objects. Using our method, the correlation between concrete geometry and semantic behaviors of an indoor scene can be established. Therefore, the recovered behavior model can be adopted to many applications that requires labeled 3D database, such as scene synthesis, scene arrangement and so on. We evaluate our algorithm on a number of indoor scenes including office, bedroom and so on. The results demonstrate that our algorithm build accurate object correspondence from imperfect scans of cluttered indoor scenes, based on which, the recovered behavior provides natural principles for many other applications oriented to indoor scenes.

Keywords: Indoor scene, behavior analysis, dynamic, object correspondence

1 Introduction

Modeling indoor scenes has attracted a large amount of attentions for decades in computer graphics. Recently, many techniques have been presented to generate static 3D models for indoor scenes, including dense modeling from RGBD data [**?**; **?**; **?**; **?**], combing object classification and modeling [**?**; **?**; **?**], and synthesizing of 3D indoor scenes from large collection of examples [**?**; **?**]. While visually appealing models are obtained for rendering, it remains challenging to extract the semantics that the geometric representation essentially encodes. On the other hand, the semantics are required in many applications such as indoor scene understanding, scene editing, and etc.

Comparing with static scenes, dynamic scene analysis has significant value in interior design, animation making, etc. The manners of how furniture objects interact with each other and how furniture objects interact with users play a very important rule in interior design. Typically, the geometric representation including object model and spatial placements of objects at different times implicitly encode the object functions and human behavior in that environment. However, the dynamic indoor scene analysis has not been investigated much in computer graphics.

Though the geometry data collection carries the behavior information, it is non-trivial to extract behavior from imperfect scans by consumer-level RGBD cameras. The challenges are in two-fold because the scanned point clouds are noisy, incomplete and with errors. First, segmentation of objects with accurate boundary is tedious because the objects in a cluttered indoor environment are in a large variety of scales. Moreover, the massive occlusions and self-occlusions in a cluttered scene makes the segmentation more challenging. Second, it is an arduous task to figure out exact object correspondences given imperfect segmentations. Furthermore, there are a great of deal of similar structures in man-made objects in indoor scenes, which leads to large ambiguities of shape correspondences.

In this paper, we present a novel algorithm to explore object behaviors in cluttered indoor scenes from a set of point clouds scanned using consumer-level RGBD cameras without any training data. The consistency and difference between frames simultaneously provide valuable hints for recovering object correspondences. First, each frame is roughly segmented into patches, which are then clustered into objects as initial correspondences hypothesis. While we project the object models back to each frame with the corresponding transformation, the consistency between captured data and the project data strengthens the initial hypothesis while the difference indicates wrong correspondences. Based on this validation, each frame is resegmented into patches. By iteratively perform the segmentation and registration steps, our work converges to coherent segmentations and correct correspondences between a bunch of objects at different scales in a large collection of points clouds.

In summary, the contributions of our system are three-fold:

1. To the best of our knowledge, our system, for the first time, performs behavior analysis in a dynamic indoor scene from point clouds scanned using consumer-level RGBD cameras without any database.

2. We present a global optimization framework to combine ob-

ject segmentation, correspondence extraction and behavior analysis in an iterative scheme.

3. We present a novel behavior model in dynamic indoor scenes, which can be applied directly to many appealing applications.

2 Related Work

Many techniques have been proposed to generate static 3D indoor scenes in computer graphics. Though none of them focus on dynamic scene analysis like our system, they provide valuable reference on the underlying techniques.

Reconstruction from RGBD Images. For static scenes, KinectFusion [?] enables the real-time reconstruction by holding and moving a depth camera. For large-scale indoor scenes with multiple rooms, reconstructing a dense 3D model from the noisy and incomplete scanned range data typically involves registration of point clouds in different views and a global optimization to reduce gaps in a large scene [?; ?]. Their goal is mainly to generate high-quality point clouds but without semantic analysis of the objects appear in the scene. Recently, object classification is employed to assist modeling for massive indoor scenes that containing many instances of chairs, desks, etc. Koppula et al. [?] first introduce the learning algorithm to understand the RGBD data of an indoor scene. To further reconstruct the 3D model for a cluttered indoor scene, 3D model databases can be used as template by searching for similar 3D model and then fitting the template to the scanned data [?; ?]. [?] do not manually collect 3D models to build the database. The template model is reconstructed by scanning the same object in different configuration. Each model has an additional presentation by geometric primitives. [?] trains the class model based on geometry and appearance features to segment and label the RGBD data captured under sparse views. By learned an initial model for each class of object in indoor environments from a pre-labelled database, the model are refined progressively with user-refined segmentation results. The 3D model can be generated by placing the most similar model in the database according to the RGBD data. If objects move in a scene, they can be detected and reposed by segmented and classified based on the learned model from previously reconstructed model [?]. Different with these techniques, we pay more attentions on analyzing the object behaviors from the dynamic range data.

Reconstruction from Sequential Point clouds. Many techniques have been proposed to reconstruct the object surfaces from the range data sequences. [?] uses a *statistical framework* to reconstruct the geometry from real-time range scanning. Each frame is divided into 3D pieces. A statistical model is used to iteratively merge adjacent frames by aligning pieces and optimizing their shapes. However, some geometric artifacts remain due to structured outliers and in some boundary regions. [?] presents a global registration algorithm to reconstruct *articulated 3D models* from dynamic range scan sequences. The surface motion is modeled by a reduced deformable model. Joints and skinning weights are solved in the system to register point clouds in different poses. (xuejin: We may also consider the furniture objects in indoor environments as articulated models, whose shapes under different poses can be deformed through connectors like hinge, slide, and so on.) A new formulation of the ICP algorithm is proposed using sparse inducing norms [?]. While it achieves superior registration result on the data with outliers and missing region, only rigid alignment is handled. A proactive capturing is employed by asking the user to move the objects to capture both interior and exterior of a scene [?]. The correspondence between adjacent frames is built first then segmentation. Xu et al. [?] employ a robot to move objects during scene reconstruction so that the ambiguities in object structures can be solved from the dynamic data. All these techniques take the advantage of the differences caused by motion to extract valuable and semantic information of object structure. We do not only take the motion information in object modeling, but also take its advantage of implicitly encoding object behavior in a scene.

Functionality Analysis using Context. Many techniques of functional analysis of one category of objects using 3D model collections have been proposed [?; ?]. Besides of functionality analysis of a single object, interaction between objects has drawn more and more attentions. The Icon descriptor is proposed to represent the functionality of 3D objects with its context [?]. Human action is involved to establish the correlations between the geometry and functionality of a region [?]. With the association of object arrangements and human activities, novel 3D scenes can be synthesized towards specific functions [?]. Inspired by these methods, we put our effort on behavior recovery, but from more challenging data. The point clouds can be easily to capture using RGBD cameras. However, the noisy and incompleteness brings tremendous challenges for extracting accurate object correspondences.

Data-Driven Furniture Layout. The general way producing the layout of furniture objects is to model a set of design rules and then to optimize an energy function given constraints by individuals. [?] formulates a group of layout guidelines in a density function according to professional manuals on furniture layout. When the user specifies the room shape and an initial arrangement of the set of furniture to be placed in the room, this system generates a number of layout suggestions by a hardware-accelerated Monte Carlo sampler. Instead of manually define the layout guidelines, the hierarchical and spatial relationships of the furniture objects can be learned from a set of examples [?]. Assembling these relationships and other ergonomic factors into a cost function, multiple arrangements can be yielded quickly by simulated annealing using a Metropolis-Hastings state search step. In these methods, manual labours are required in modeling the design rules and providing an initial layout. Fisher et al. [?] trains a probabilistic model for indoor scenes from a small number of examples. A variety of indoor scenes can be automatically synthesized from a few of user specified examples. Indoor scenes bring more difficulties for scene analysis because there are always many cluttered objects in different scales, shapes, and functions. A focal-driven analysis and organization framework is presented for heterogeneous collections of indoor scenes [?]. They develop a co-analysis algorithm which interleaves frequent patten mining and subspace clustering. The interrelations between objects play important role during furniture arrangement in these systems. However, the 3D scene models takes many efforts to collect for training. In comparison, our system provides an efficient framework to generate 3d model examples for many further applications.

3 Overview

The objective of our algorithm is to recover object behaviors from a set of point clouds scanned at different times for an indoor scene. To achieve this goal, objects and correspondences between objects and point clouds must be extracted. More specifically, our problem is to transfer the noisy and incomplete point-level data into object-level models and correspondences to understand the object behaviors in the scene.

Our system consists of two main steps, *point cloud segmentation* and *object registration*, as Figure 2 shows. The **input** is a set of point clouds scanned using a RGBD camera (a). We scan each scene at different times during a month using a Microsoft Kinect

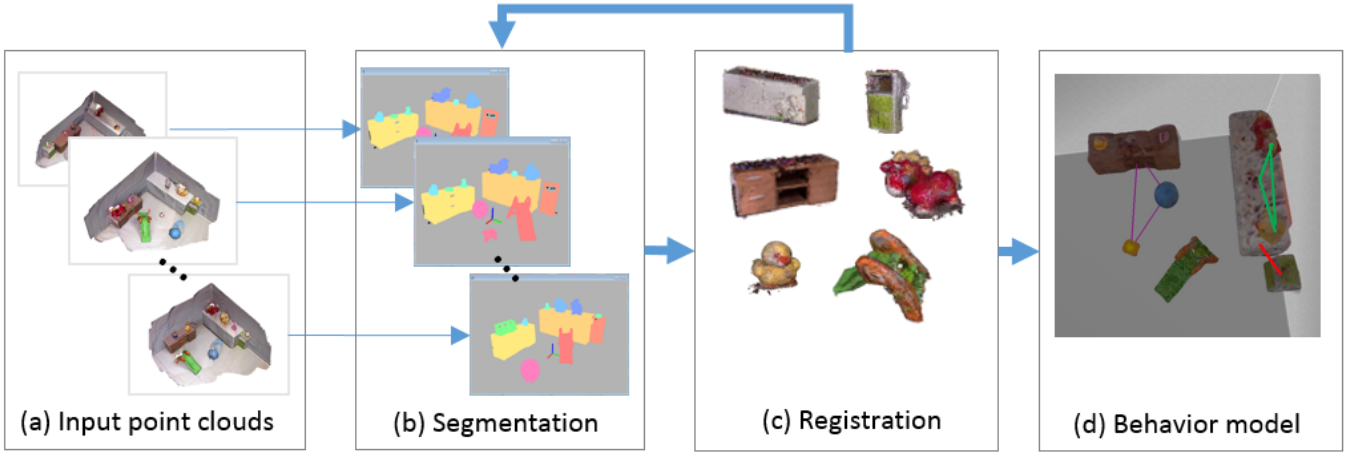

Figure 2: *System overview.*

V1. A set of dense point clouds are generated using a real-time fusion system [?]. It is burdensome to scan every detailed structure in a cluttered indoor scene. As a result, each point cloud is incomplete and noisy due to object occlusions.

We first segment each frame simply using region growing (b). There are very likely many wrong boundaries in the generated patches. Then we cluster all the patches from all frames into k clusters using $k$means. [?] have demonstrated the power of features based on bounding box in the segmentation of indoor objects. Therefore, we design the descriptor of each patch as the cascades of length, width and height of its bounding box, mean and standard deviation of the distance of each point to its bounding box, percentage of closest points to the faces of its bounding box. The feature dimension is then reduced using PCA. (xuejin: Currently, this step is manually done.)

In each cluster, the patches are registered using a joint registration method [?] to produce the object model of this cluster (c). There are inevitably wrong registration due to wrong clustering. Therefore, we project the generated object models back into each frame using the estimated transformation. Re-segmentation is performed using the model consistency and neighborhood information in each frame, as described in Sec. 4. By iteratively register resegmented patches and re-segment frames, our system converges to a set of well-registered 3D object models and accurate correspondence between object model and all point clouds. Then we learn the behavior model (d) based the object correspondences from all input point clouds (Sec. 5), then apply the behavior model into many applications (Sec. 6).

4 Iterative Segmentation and Registration

4.1 In-Patch Graph Cut

In this step, we decompose each patch into two parts, one is matched to the correspondent object model and the other is identified as outliers. The outliers will join the unlabeled region and participate in the generation of new object models in the next iteration. The data term we use is formulated as follows:

$$data(P_{patch},L)=\begin{cases}1-match(T(P_{patch}),P_{object})*conf(P_{object}), L=1\\ match(T(P_{patch}),P_{object})*conf(P_{object}), L=0\end{cases} \tag{1}$$

where the match function is defined as:

$$T(P)= \tag{2}$$

the $T(P)$ is the rigid registration for pixel P. the confidence score is evaluated in the registration step and the P_{object} is retrieved by transform the patch according to the result of registration and find the closest point of P_{patch}

4.2 Global Consistent Graph Cut

In this step, two goals are achieved. Firstly, re-collect frequently occluded points for each patch from unlabeled region. Secondly, refine the border of segmentations.

Figure 3 shows the segmentation is progressively refined.

5 Behavior Model

Our behavior model in a dynamic scene consists of three parts. One is the behavior of single object, which is represented by the statistics of the object motion (rotations/translations) interpreted from its orientations and positions in frames. The second is the behavior of a group of object, which means the correlations of the statistics of the motions of objects. Their supporting/proximity relationship or arrangements of a group of objects keep consistent. c. The relationship variations caused by motion of the objects in the set of point clouds. In other words, the behavior of the object relationship. For example, the support relationship of vase can change from one desktop to another desktop or so.

5.1 Pairwise Object Relations

In a cluttered indoor scene, there are many pairs of objects that usually co-occur in the scene with some typical geometric relations in a long time period. In our system, we consider two types of pairwise relations, which are commonly used to describe object relations in many previous methods [?; ?; ?; ?].

Support $SR(\mathbf{O}_a,\mathbf{O}_b)$ indicates that $\mathbf{O}_a$ is supported by $\mathbf{O}_b$. We use a Gaussian function to indicate the distribution of $\mathbf{O}_a$ on the supporting surface of $\mathbf{O}_b$; In each frame, we check any pair of objects connected by a surface. Typically, smaller object is supported

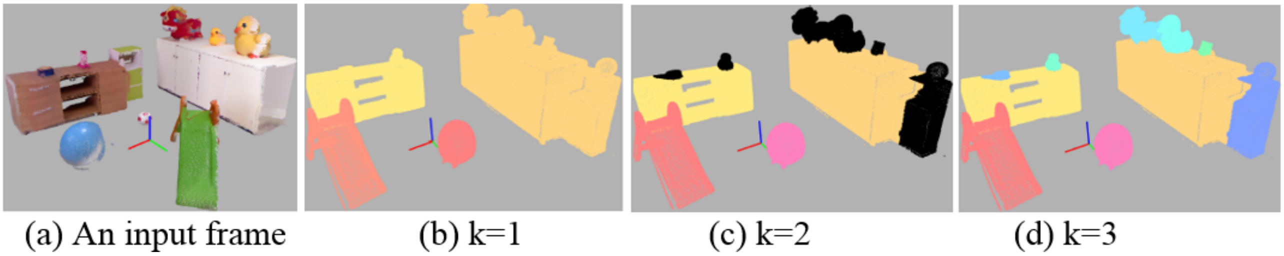
(a) An input frame (b) k=1 (c) k=2 (d) k=3

Figure 3: *The segmentation of each frame is progressively refined based on registered object models. From left to right: an input point cloud (a), segmentation updates at three iterations. (xuejin: show corresponding object model at each iteration.)*

by the larger one, the upper object is supported by the bottom object. We transform the object been supported $\mathbf{O}_a$ into the local coordinates of its support object $\mathbf{O}_b$ to get its location $\mathbf{x}_a$ and orientation θ_a. According to all the examples of this pair in the input frames, we compute a Gaussian function

$$SR(\mathbf{O}_a,\mathbf{O}_b) = \mathcal{N}_{position}(\mu_x,\Sigma_x)\mathcal{N}_{orientation}(\mu_o,\sigma_o) \tag{3}$$

Each relation has a confidence or frequency w in all the frames.

Proximity $PR(\mathbf{O}_a,\mathbf{O}_b)$ indicates $\mathbf{O}_a$ is close to $\mathbf{O}_b$ in with a Gaussian function $\mathcal{N}(\mu,\sigma)$ including position x,y in the local coordinate system of $\mathbf{O}_a$ and the orientation θ_b of $\mathbf{O}$ in the local coordinate system of $\mathbf{O}_b$, defined as

$$PR(\mathbf{O}_a,\mathbf{O}_b) = \mathcal{N}_{position}(\mu_x,\Sigma_x)\mathcal{N}_{orientation}(\mu_o,\sigma_o). \tag{4}$$

If the background floor and wall are taken into account for pairwise relations, the relation between a single object and the background is actually the behavior of this object itself.

5.2 Group Behavior

Based on all the extracted pairwise relation, a directed graph $G(V,E)$ is constructed to encode the mutual relations of all the objects in the scene, as shown in Figure 4. Each node is an object model in the scene. Each directed edge represents a pairwise geometric relation between two objects. The information carried in each edge includes:

1. **Relation type**: supporting or proximity. Each edge is directed from a small object to an larger object indicating that the small object is usually associated with a larger object.

2. **Edge weight**: is the frequency of that two objects co-occur in the scene. We define it

$$w(i \rightarrow j) = \frac{c(i \rightarrow j)}{K} \tag{5}$$

   where $c(i \rightarrow j)$ is the total counts of that two objects co-occur and follow the relation defined by the edge in all the frames. K is the number of all frames.

3. **Relation description**: Associated with each edge, the statistics of the spatial relations defined in Sec. 5.1 is also stored.

A **structural group** G_s is defined as a *complete graph* in which every pair of nodes in connected by a unique edge, to describe the mutual relations between nodes in a subset. From the directed graph $G(V,E)$, a set of structural groups can be extracted. There are many potential structural groups in G. The smallest structural group is an edge connecting two nodes. However, we only focus on reliable structural groups for further application, such as layout re-arrangement, unusual event detection, etc. The reliability of a structural group $G_s(V_s,E_s)$ with k objects is defined as defined as

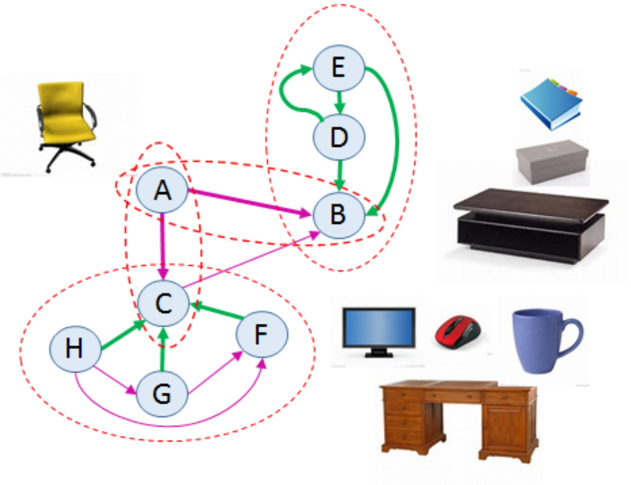

Figure 4: *Construction a directed geometric graph from pairwise relations. Green edges indicate supporting relations. Purple edges represent proximity relations. The edge width indicates the frequency of each relation. Each dashed ellipse represents a structural group. The pair (B,C) should be very weak because they are far away from other other, and can be removed. Otherwise, (A,B,C) is another structural group.*

$$\rho_{sg} = \Big(\prod_{e_i \in E_s} w(e_i)\Big)^{\frac{1}{k}} \tag{6}$$

6 Applications

7 Results and Discussions

8 Conclusions

In this paper, we present