PROPOSAL: Form and Function Exploration in Interior Environments from Dynamic RGBD Data

Abstract

RGBD cameras is becoming more and more popular for common users to capture the environment where they live. In this paper, we present a *form and function* exploration system to reconstruct the 3D geometry and mine furniture functions in interior environments from dynamic RGBD data. We capture a sequence of RGBD images in a long period of time, in which objects may change their positions and poses frequently. To reconstruct the geometry of the cluttered indoor scene, we first cluster the objects by their motions: static objects or dynamic objects. Object motion is computed from correspondence detection from both appearance and geometry. During the motion clustering, the interrelation between different objects can be discovered from the spatial relationship at different times.

Keywords: Interior, behavior analysis, dynamic exploration, function

1 Introduction

3D indoor scenes are popular in many applications, such as games, robotics, virtual reality, etc. Modeling indoor scenes has been attracted large amount of attentions for decades in computer graphics. Recently, many techniques have been presented to generate static indoor environments, including dense modeling from RGBD data [Henry et al. 2012; Izadi et al. 2011; Xiao and Furukawa 2012; Yan et al. 2014], combing object classification and modeling [Shao et al. 2012; Nan et al. 2012; Kim et al. 2012], and synthesizing of 3D indoor scenes from large collection of examples [Fisher et al. 2012a; Xu et al. 2013].

Comparing with static scenes, dynamic scene analysis has significant value for artists 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 function (behavior) of an object is interrelated with its form (geometry). In a dynamic scene, the function of an object is reflected its different forms. However, the dynamic indoor scene analysis has not been investigated much in computer graphics.

A few techniques have been proposed to analyze the spatial relationships between objects in a cluttered scene either from a boundary model [Sharf et al. 2013] or an RGBD image [Nathan Silberman and Fergus 2012]. The former technique learns features to classify the support relationship between RGBD image patches from a training dataset and then segments and classifies the support relationship of a new RGBD image. The later technique do not use any training data. It builds a support tree to describe the supporting-supported relationship between objects. Furthermore, it learns the mobility of each object/part from various poses of the repetitive instances of the same object. However, perfectly matched meshes and the large variety are required, which is very challenging for raw RGBD images.

In this paper, we present a framework to automatically explore the object behaviour in a cluttered indoor environment from a set of RGBD images without any training data. Because the raw RGBD data is noisy and incomplete due to occlusions in a cluttered environment, it is very challenging to reconstruct a perfect 3D model for the scene. Ambiguous boundaries between objects make it nontrivial to generate clear motion representation.

Robust PCA [Candès et al. 2011] is first used to segment the dynamic objects first. Then a behavior map is built to represent the dynamic spatial interactions between objects in the scene.

The contributions of our system are four-fold.

1. To the best of our knowledge, our system, for the first time, performs behavior analysis in a dynamic indoor scene from raw RGBD data taken during a long time without any database.

2. We present a global optimization framework to combine object segmentation, background completion, behavior analysis in an iterative scheme.

3. Our object segmentation method using RPCA simultaneously segment objects and complete the occluded objects in a cluttered environment.

4. We present a novel graph representation and optimization technique for behavior analysis in a dynamic scene.

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 Data For static scenes, KinectFusion [Izadi et al. 2011] 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 [Xiao and Furukawa 2012; Henry et al. 2012]. 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. 2011] 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 [Shao et al. 2012; Nan et al. 2012]. [Kim et al. 2012] 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. [Shao et al. 2012] 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 [Liu et al. 2014]. Different with these techniques, we pay more attentions on analyzing the object behaviors from the dynamic range data.

Reconstruction from Dynamic Point clouds Many techniques have been proposed to reconstruct the object surfaces from the range data sequences. [Wand et al. 2007] 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. [Chang and Zwicker 2011] 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.)

[Bouaziz et al. 2013] propose a new formulation of the ICP algorithm using sparse inducing norms. While it achieves superior registration result on the data with outliers and missing region, only rigid alignment is handled. [Yan et al. 2014] employ a proactive capturing 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. (xuejin: However, the motion information worths more than just helping registration. It can be used for analysis of object movements and functions.)

(xuejin: 1. In comparison, the range data used in our system is captured with a large time spacing and the motion of the objects in the scene varies in a wide range. 2. Can we use image/texture data for more reliable correspondence?)

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. [Merrell et al. 2011] 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 [Yu et al. 2011]. 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. [2012b] 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 [Xu et al. 2014]. 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.

Co-analysis of shape, functions in a large database of 3D object models With the growth of 3D shape databases on the Internet, many techniques have been proposed for co-analysis in a large shape collection of the same object category. A series of geometry processing tasks such as model segmentation, shape retrieval, and shape synthesis. Point-to-point networks are used to represent the shape correspondence between shapes [Rustamov et al. 2013]. To better explore the shape space, [Huang et al. 2014] propose a framework for computing consistent functional maps within heterogeneous shape collections. Cycle-consistency of the functional map network largely reduce the noise correspondences. Based on the continuous nature of functional maps, the proposed framework outperforms point-based representation in shape interpolation, shape retrieval and classifications for both man-made and organic shapes. Large 3d model collection can also help recovering the depth map for a single image [Su et al. 2014]. With a non-rigid registration formulation, the image is popped-up to minimize the distance between corresponding points in the image and similar 3d shapes in the database. However, all these techniques focus on the geometry characteristics within one object category. In a cluttered environment, the inter-connection between different types of objects has not been investigated yet.

RGBD image understanding Though many techniques have been proposed to model a cluttered indoor environment based on databases, little work has been done for the physical interactions between objects. [**?**] introduce an framework to segment the RGBD image and infer the support relationship between objects in a cluttered indoor scene. A dataset is also provided for various tasks, such as recognition, segmentation and relationship inference.

3 Overview

Our system consists of two main steps, *object identification* and *behavior analysis* as Figure 1 shows.

1. *Registration/Labeling:* Given a sequence of RGBD data captured in different times and views, we first register *all the RGBD images* to segment objects and detect motions according to their low-rank characteristics using Robust PCA [Candès et al. 2011]. Given the assumption that there are typically static objects, such as wall, floor and so on, moving objects can be treated as sparse noise and the static backgrounds can be separated and completed as low-rank part by the robust PCA. The point clouds are then divided into two categories: static objects and dynamic objects. Combing the separated sparse part and the low-rank part, *object segmentation* is performed by combing the motion, depth, and appearance features in the RGBD images.

2. *Function Analysis/Behavior Map:* With the segmented regions, objects should be identified according to their appearance and geometry features among the entire image sets. We build a dynamic behavior map including all of the objects in the scene. Each node is an object/a part of an object. The edges in the graph describe the spatial relationship between objects/parts. The object behavior can be explored from its surrounding objects in the dynamic structure graph. With a dynamic behavior map, the RGBD data is re-segmented and re-labeled to obtain semantical consistency of the scene.

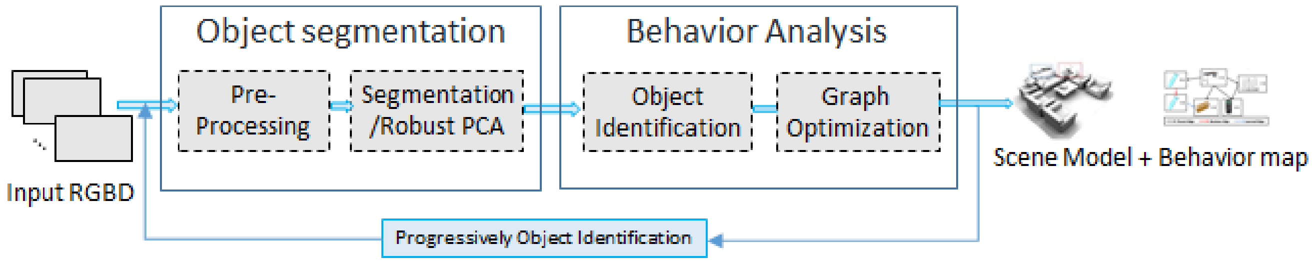

Figure 1: *System overview.*

4 Problem

Given a set of RGBD images $\{I_i\}_{i=1,\ldots,T}$ of an indoor scene taken under different times, where each pixel in I_i is defined by $\mathbf{x} = [r,g,b,d]^T$. Our behavior analysis algorithm aims to recover all the objects $\{O_i\}_{i=1,\ldots,N_o}$ appear in the scene and the behavior map G of the scene. Each object O_i is represented by its image regions $M_i, \{R_{i,t}\}_{t=1,ldots,T}$, where M_i is its registered point cloud, and $R_{i,t}$ denotes its corresponding image region in I_t. $R_{i,t} = \Phi$ if this object does not appear in the image I_t at time t.

Generally, this is a labeling problem. Each pixel at each time should be assigned with an object label. The challenging is:

1. The number of objects N_o is unknown.
2. Discrete labeling problem is NP-hard. Considering all the pixels, the node number would be $Width \times Height \times T$. The computational complexity is extremely huge.

In order to reduce the computational complexity, we need a hierarchical/progressive method to build object correspondences.

1. From the discrete RGBD pixels, we first decompose the input images into static background and moving objects using RPCA.
2. Segment each frame into multiple regions.
3. Build correspondence between regions of all the frames to identify objects in the scene.

5 Segmentation and Registration of RGBD images

Since the method used in [Shao et al. 2012] requires a certain amount of user interactions, we first register all the RGBD images taken in different views and different times using Robust PCA [Candès et al. 2011] without any pre-labeled or training data.

No work has been done for solving geometry problem using low-rank matrix. For rigid objects, small objects with large motion can be detected as noise. The challenging problem is how to detect outliers using the low-rank technique. Outlier means the objects which is not appear at all the frames. It probably appears at the begin and then disappears at the last frames. Some objects which do not show up at the begin but then appear at last frames are also outliers in low-rank problem. We have to think about how to build scene correspondences with outliers.

Segment static objects Using RPCA, we input a matrix $D_{WH\times T}$. Each column of D is a stacked image of $W \times H$ pixels. After RPCA, a low-rank matrix A and a sparse matrix E is generated. Therefore, each input image can be decomposed as $I_i = I_i^b + I_i^e$.

Figure 2 shows three groups of experiments on the RPCA algorithm. Three scenes with different range of object motions are tested.

1. For the dynamic scene with large object motions, the recovered low-rank matrix A represent the static background very well while the recovered sparse matrix E represent the moving objects.
2. For the dynamic scene with moderate object motions, the recovered low-rank matrix A contains both the static background and a part of moving objects due the their overlapping at different times. The recovered sparse matrix E only indicates the part of the moving object where the depth data changes.
3. For the dynamic scene with small object motions, the result is similar with the case of moderate object motions.
4. In the three cases, new object can be correctly detected.

From the extracted A, we hope to segment the background first. Then we segment the moved objects from I_i^e or what we see about an object in the input image $I_i^b + I_i^e$ at where $I_i^e \neq 0$.

Problem of this stage:

1. Input: $D = A + E$. For each input image, it can be decomposed to $I_i = I_i^b + I_i^e$.
2. Output: $\{R_{i,t}\}_{i=1,\ldots,N_o;t=1,\ldots,T}$. Determine the optimal number N_o of objects in the scene. Segment the object regions in each frame. If possible, build initial object correspondence at this stage.

We segment the images A and I_i^e separately.

Why not co-segmentation Most of the state-of-the-art co-segmentation techniques segment the instances of one object in a group of images by recovering the similar appearance features. In our cases, the furniture objects usually have similar appearances and there are more than one types of objects (like chairs, cups, books) in the scenes. Moreover, the co-segmentation techniques could not complete the background/static regions due to occlusion.

6 Behavior Analysis

Object behaviour in an indoor environment is defined as the relationship between objects, including spatial relationships, function interactions and so on. We construct a graph $G = V, E$, where $V = \{V_i\}_{i=1,\ldots,M}$, M is the total number of all the furniture objects in the dynamic indoor scene. $E = \{e_{i,j}\}$ describe all the pairwise interrelations between the furniture objects. (xuejin: TBD: the rep-

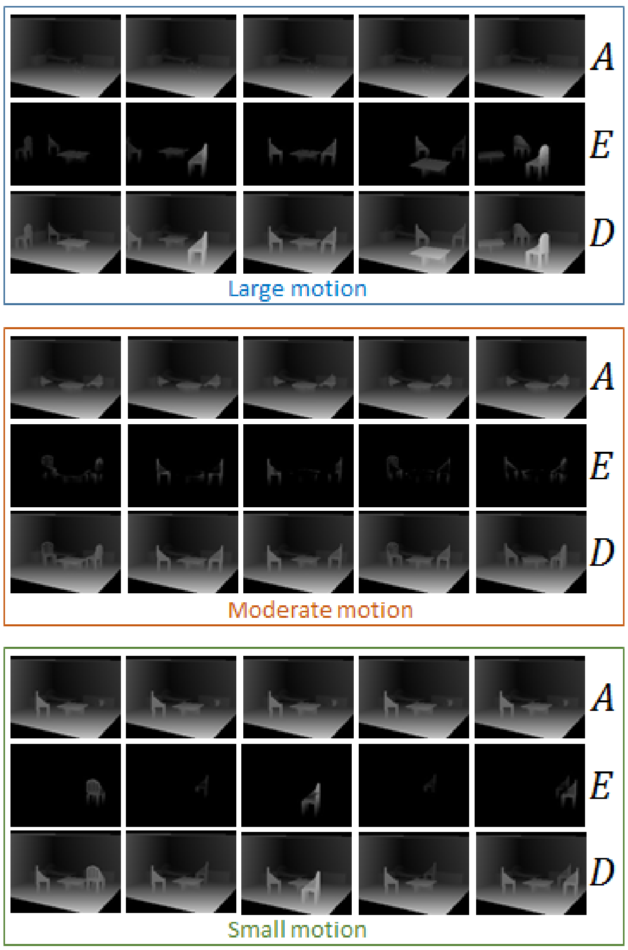

Figure 2: *Image decomposition by RPCA for three scenes.*

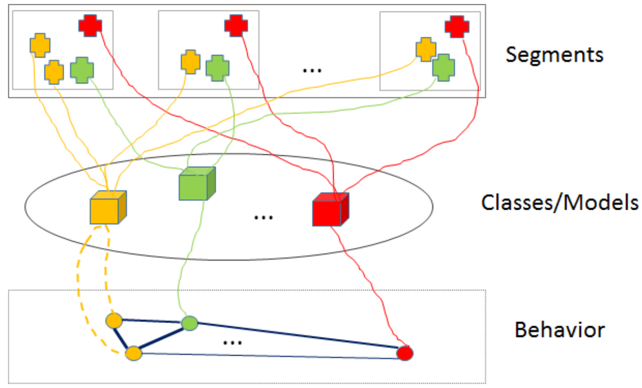

Figure 3: *The relationships between the terms used in our algorithm.*

resentation of behaviour can be groups, sub-graphs, and so on to describe the relationship between more than two objects.)

In order to generate the behavior map of all the furniture instances from the RGBD image set in the scene, we use a two-step framework, as shown in Figure 4. The first step is to recover all the object classes from all the RGBD segments. The second step is to analyze the behavior of all the object instances in the entire indoor environment.

Figure 3 illustrate the terms used in our system. Segments indicate the RGBD image regions in all frames. Ideally, each segment is from only one object, and each object will project to only one segment on each frame. The object class used in our system does not indicate the semantic label, such as door, table and so on. In comparison, each object class used in our system indicates a unique 3D model (point cloud) in the scenes. For example, the two yellow squares in the RGBD frames have the same geometric and appearance features, so that they are assigned to the same class. To build the behavior map, we should go back to the furniture objects appear in the scene. Objects mean that all the separable objects in the scene. For objects who have the same geometry and appearance, they belong to the same class/model, but they are different furniture objects. However, the correspondence between furniture objects and segments is ambiguous because of there are moving object with the same model. For example, there are two chairs at the beginning, and then one of them disappear in the other two frames in Figure 3. We can not recover the physical correspondences between the two chair objects and the segments in the RGBD frames.

Definitions Before we explain the algorithm details, we first describe the definitions of each term used in our algorithm.

1. $\{I_k\}_{k=1}^{K}$: the input RGBD images;
2. K: the number of the input RGBD images;
3. $\mathbf{p} = (r, g, b, d, \mathbf{n})$: a pixel in an RGBD image, including its color, depth, normal;
4. $\{S_{i,k}\}_{i=1,\ldots,n_k;k=1,\ldots,K}$: the RGBD segments in I_k generated by our segmentation process.
5. n_k: the number of all the segments in I_k.
6. $\{C_l\}_{l=1}^{N}$: all the object classes appear in the scene.
7. N: the total number of all the object classes in the scene.
8. $\{O_m\}_{m=1,\ldots,M}$: all the objects appear in the scene.

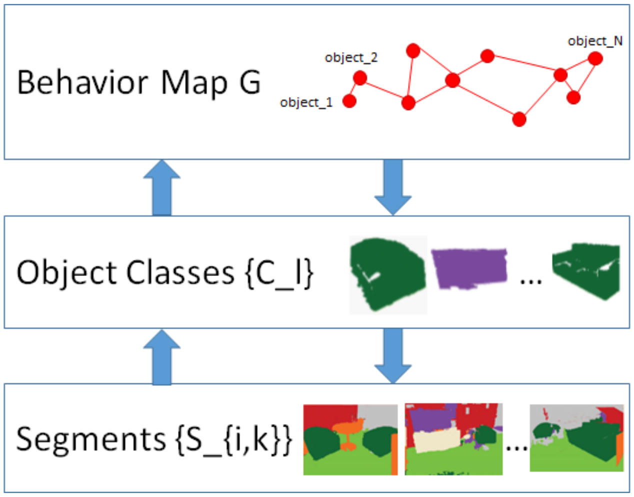

Figure 4: *The pipeline of behavior analysis.*

9. M: the total number of all the objects appear in the scene.

The difference between the object class C_l and the object O_m is that different objects in the scene can be the same object class. For example, two chairs in the scene belong to the same object class. Each object class has its unique geometric and appearance features. Different object classes must point to different objects.

6.1 Segments to Object Classes

From the segmentation stage described in Sec. 5, we get a set of segments/regions, $\{S_{i,k}\}_{i=1,\ldots,n_k;k=1,\ldots,K}$, in the RGBD images. To register them in 3D space, we should first to recover the correspondences between all the image segments. This is to compute a set of object classes $\{C_l\}_{l=1}^{N}$ from all the segments. However, the difficulty is the total number of all the object classes N is unknown, and the total number of segments $\sum_{k=1}^{K} n_k$ is very large so that it is non-trivial to compute the correspondences between all the RGBD images in a long time. On the other hand, objects appear or disappear in the entire time period so that the total numbers of object instances n_k in each frame are not the same. It brings more difficulty for our problem.

In order to solve the problem with unknown number of object classes and unknown number of instances in each frame, we use a non-parametric Bayesian modeling based on Dirichlet process. More specially, we formulate our problem in a framework of distance dependent Chinese Restaurant Processes (ddCRP) [Blei and Frazier 2011][Chiu and Fritz 2013].

For each RGBD segment S_i, we compute its features $\mathbf{x}_i$ including

1. The appearance features $\mathbf{h}_i^a$: include a typical Bag of Words (BoW) image representation. The length of the codebook is 100.
2. Geometric features $\mathbf{h}_i^d$: a typical BoW of the depth image;
3. the size $\mathbf{s}_i$ of the segment;
4. h_i: the height of the object where it stands;
5. $\mathbf{h}_i^n$: the normal histogram with 16 bins.

6.1.1 ddCRP

We briey introduce the basic idea of CRP and its extension to ddCRP. CRP is an alternative representation of Dirichlet process model and it denes the following procedure. Imagine a restaurant with an innite number of tables. A sequence of customers come enter the restaurant and sit at randomly chosen tables. The i-th customer sits down at a table with a probability that is proportional to how many customers are already sitting at that table or opens up a new table with a probability proportional to a hyperparameter. Their seating conguration represents a random partition also called *table assignments*. Thus CRP provides a exible prior distribution over table assignments where the number of tables is potentially innite. Since the table assignment of each customer just depends on the number of people sitting at each table and is independent of the other ones, the ordering of customers does not affect the distribution over partitions and therefore exchangeability holds.

While in some cases there are spatial or temporal dependencies between customers, the exchangeability does not hold any more, the generalized process allowing nonexchangeable distribution over partitions is needed. The ddCRP was proposed to offer an intuitive way for modeling non-exchangeability and dependency. The main difference between the CRP and ddCRP is that rather than directly linking customers to tables with table assignments, in ddCRP customers sit down with other customers according to the dependencies between them, which leads to customer assignments. Groups of customers sit together at a table only implicitly if they can be connected by traversing the customer assignments. Therefore the i-th customer sits with customer j with a probability inversely proportional to the distance d_{ij} between them or sits alone with a probability proportional to the hyperparameter $\aleph$:

$$p(c_i = j|D, f, \alpha) \propto \begin{cases} f(d_{ij}) & j \neq i \\ \alpha & j = i \end{cases} \tag{1}$$

where c_i is the customer assignment for customer i, and $f(d)$ is the decay function, and D denotes the set of all distances between customers. The decay function f should be non-increasing, takes non-negative finite values, and satisfies $f(\infty) = 0$. It describes how distances between customers affect the probabilities of linking them together. Figure 5 demonstrates a ddCRP, from which we can see how to cluster the customer to different clusters/tables with unknown number of clusters.

6.1.2 Class assignment for segments

In this section, we describe how we cluster the RGBD segments S_i into object classes C_l using ddCRP. Customers correspond to segments $\{S_i\}$, and tables correspond to object classes $\{C_l\}$. The decay function used in Eq. 1 is now defined according to the appearance and geometric features of the segments:

$$f(d) = \frac{\min(h_i, h_j)}{\max(h_i, h_j)} \frac{\min(s_i, s_j)}{\max(s_i, s_j)} e^{-\left(\frac{d^a}{\sigma_a} + \frac{d^g}{\sigma^g}\right)}, \tag{2}$$

where d_a is the distance of appearance features, and d_g is the distance of geometric features, defined as following:

$$\begin{aligned} d_{ij}^a &= \|\mathbf{h}_i^a - \mathbf{h}_i^a\| \\ d_{ij}^g &= \|\mathbf{h}_i^d - \mathbf{h}_i^d\| + \|\mathbf{h}_i^n - \mathbf{h}_i^n\| \end{aligned} \tag{3}$$

(xuejin: Test the class assignments with the prior.)

After one pass of the class assignment for all the RGBD segments in all frames, we assign a unique class c^{max} to the segment if

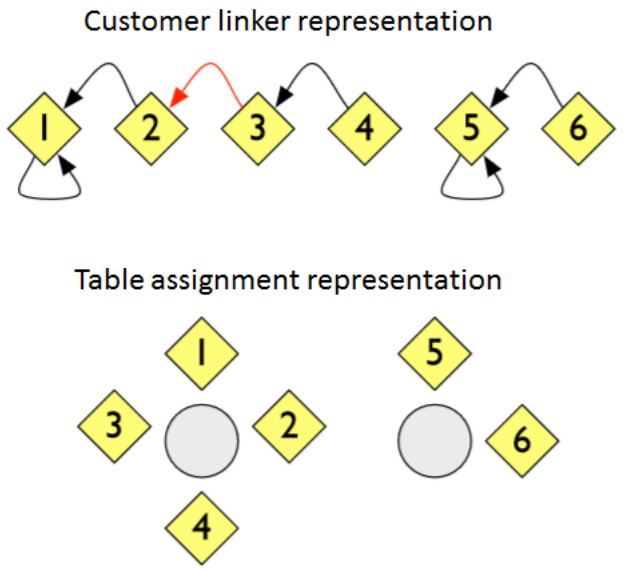

Figure 5: *Illustration of a ddCRP. For each customer i (yellow square), ddCRP assigns another customer j for him to sit with (upper). The table assignment (bottom) can be derived from the customer linker.*

$\frac{p(c^{sec}|D,f,\alpha)}{p(c^{max}|D,f,\alpha)} < T_{ass}$. Otherwise, this segment has ambiguity on its class.

Iterative update model of each class Once we assign the object class for some segments, we can register the multiple segments assigned to the same class to build a 3D point cloud for the class. This can be done by rigid registration.

Because there are large variance of appearances and geometric features due to the multiple poses of the same object class in the scene, the ddCRP probably fail to assign the same class for the segments of the same object class from very different points of view. However, due to the rigidity of the furniture object, a post-merging process can be done to merge the table/class assignments if the registered object models are very similar.

Iteratively assign class for segments Once we build the point cloud for each class, we run a CRP to refine the class assignment for each segment according to the object class. The probability to assign class l to the segment S_i is:

$$p(c_i = l|\mathbf{x}_i, \alpha) = \begin{cases} p(\mathbf{x}_i|C_l) & l \leq N \\ \alpha & l = N+1 \end{cases} \quad (4)$$

We iterative update the model of each class C_l according to the current class assignment and then re-assign the class to all the segments according to the updated class model until convergence.

6.2 Classes to Behavior

7 Discussions

In our framework, no training data is required and the entire process is fully automatic. We do not claim that our system is able to detect all the small objects and accurately recover its motion behavior using the current automatic framework. However, using our method, we can point out which object probably appears or disappear suddenly or it probably has unusual behavior. With this kind of report, the user can easily interact with the large set of objects and rgbd images in a cluttered environment rather that manually check each frame and each object, which is non-trivial for common users.

8 Plan

1. Dec15 Segmentation on RGBD images (Cheng); Segments to Models for the synthetic data. (Xuejin).
2. Dec20 Behavior Analysis.
3. Dec30 Refine algorithms on real data.
4. Jan20 Paper writing and demo video.

References

BLEI, D. M., AND FRAZIER, P. I. 2011. Distance dependent chinese restaurant processes. *J. Mach. Learn. Res. 12* (Nov.), 2461–2488.

BOUAZIZ, S., TAGLIASACCHI, A., AND PAULY, M. 2013. Sparse iterative closest point. *Computer Graphics Forum (Symposium on Geometry Processing) 32*, 5, 1–11.

CANDÈS, E. J., LI, X., MA, Y., AND WRIGHT, J. 2011. Robust principal component analysis? *Journal of the ACM (JACM) 58*, 3, 11.

CHANG, W., AND ZWICKER, M. 2011. Global registration of dynamic range scans for articulated model reconstruction. *ACM Transactions on Graphics 30*, 3.

CHIU, W.-C., AND FRITZ, M. 2013. Multi-class video co-segmentation with a generative multi-video model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

FISHER, M., RITCHIE, D., SAVVA, M., FUNKHOUSER, T., AND HANRAHAN, P. 2012. Example-based synthesis of 3d object arrangements. *ACM Trans. Graph. 31*, 6 (Nov.), 135:1–135:11.

FISHER, M., RITCHIE, D., SAVVA, M., FUNKHOUSER, T., AND HANRAHAN, P. 2012. Example-based synthesis of 3d object arrangements. *ACM Trans. Graph. 31*, 6 (Nov.), 135:1–135:11.

HENRY, P., KRAININ, M., HERBST, E., REN, X., AND FOX, D. 2012. RGB-D mapping: Using kinect-style depth cameras for dense 3D modeling of indoor environments. *International Journal of Robotics Research (IJRR) 31*, 5 (April), 647–663.

HUANG, Q., WANG, F., AND GUIBAS, L. 2014. Functional map networks for analyzing and exploring large shape collections. *ACM Trans. Graph. 33*, 4 (July), 36:1–36:11.

IZADI, S., KIM, D., HILLIGES, O., MOLYNEAUX, D., NEWCOMBE, R., KOHLI, P., SHOTTON, J., HODGES, S., FREEMAN, D., DAVISON, A., AND FITZGIBBON, A. 2011. Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth camera. In *ACM Symposium on User Interface Software and Technology*.

KIM, Y. M., MITRA, N. J., YAN, D.-M., AND GUIBAS, L. 2012. Acquiring 3d indoor environments with variability and repetition. *ACM Transactions on Graphics 31*, 6, 138:1–138:11.

KOPPULA, H., ANAND, A., JOACHIMS, T., AND SAXENA, A. 2011. Semantic labeling of 3D point clouds for indoor scenes. In *Conference on Neural Information Processing Systems (NIPS)*.

LIU, Z., TANG, S., XU, W., BU, S., HAN, J., AND ZHOU, K. 2014. Automatic 3D indoor scene updating with rgbd cameras. *Computer Graphics Forum (Pacific Graphics) 33*, 7.

MERRELL, P., SCHKUFZA, E., LI, Z., AGRAWALA, M., AND KOLTUN, V. 2011. Interactive furniture layout using interior design guidelines. *ACM Trans. Graph. (Siggraph'11).*

NAN, L., XIE, K., AND SHARF, A. 2012. A search-classify approach for cluttered indoor scene understanding. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2012) 31*, 6.

NATHAN SILBERMAN, DEREK HOIEM, P. K., AND FERGUS, R. 2012. Indoor segmentation and support inference from rgbd images. In *ECCV*.

RUSTAMOV, R. M., OVSJANIKOV, M., AZENCOT, O., BEN-CHEN, M., CHAZAL, F., AND GUIBAS, L. 2013. Map-based exploration of intrinsic shape differences and variability. *ACM Trans. Graph. 32*, 4 (July), 72:1–72:12.

SHAO, T., XU, W., ZHOU, K., WANG, J., LI, D., AND GUO, B. 2012. An interactive approach to semantic modeling of indoor scenes with an rgbd camera. *ACM Trans. Graph.*, 136–136.

SHARF, A., HUANG, H., LIANG, C., ZHANG, J., CHEN, B., AND GONG, M. 2013. Mobility-trees for indoor scenes manipulation. *Computer Graphics Forums 32*, 1–13.

SU, H., HUANG, Q., MITRA, N. J., LI, Y., AND GUIBAS, L. 2014. Estimating image depth using shape collections. *ACM Trans. Graph. 33*, 4 (July), 37:1–37:11.

WAND, M., JENKE, P., HUANG, Q., BOKELOH, M., GUIBAS, L., AND SCHILLING, A. 2007. Reconstruction of deforming geometry from time-varying point clouds. In *Proceedings of the Fifth Eurographics Symposium on Geometry Processing*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, SGP '07, 49–58.

XIAO, J., AND FURUKAWA, Y. 2012. Reconstructing the world's museums. In *Proceedings of the 12th European Conference on Computer Vision*, ECCV '12.

XU, K., CHEN, K., FU, H., SUN, W.-L., AND HU, S.-M. 2013. Sketch2scene: Sketch-based co-retrieval and co-placement of 3d models. *ACM Transactions on Graphics 32*, 4, 123:1–123:12.

XU, K., MA, R., ZHANG, H., ZHU, C., SHAMIR, A., COHEN-OR, D., AND HUANG, H. 2014. Organizing heterogeneous scene collection through contextual focal points. *ACM Transactions on Graphics, (Proc. of SIGGRAPH 2014) 33*, 4, to appear.

YAN, F., SHARF, A., LIN, W., HUANG, H., AND CHEN, B. 2014. Proactive 3d scanning of inaccessible parts. *ACM Transactions on Graphics(Proc. of SIGGRAPH 2014) 33*, 4.

YU, L.-F., YEUNG, S. K., TANG, C.-K., TERZOPOULOS, D., CHAN, T. F., AND OSHER, S. 2011. Make it home: automatic optimization of furniture arrangement. *ACM Trans. Graph. 30*, 4, 86.