Pacific Graphics 2017
J. Barbic and O. Sorkine-Hornung
(Guest Editors)

Volume 36 (2017), Number 7

Interactive Point Set Joint Registration and Co-segmentation

ID: paper1049

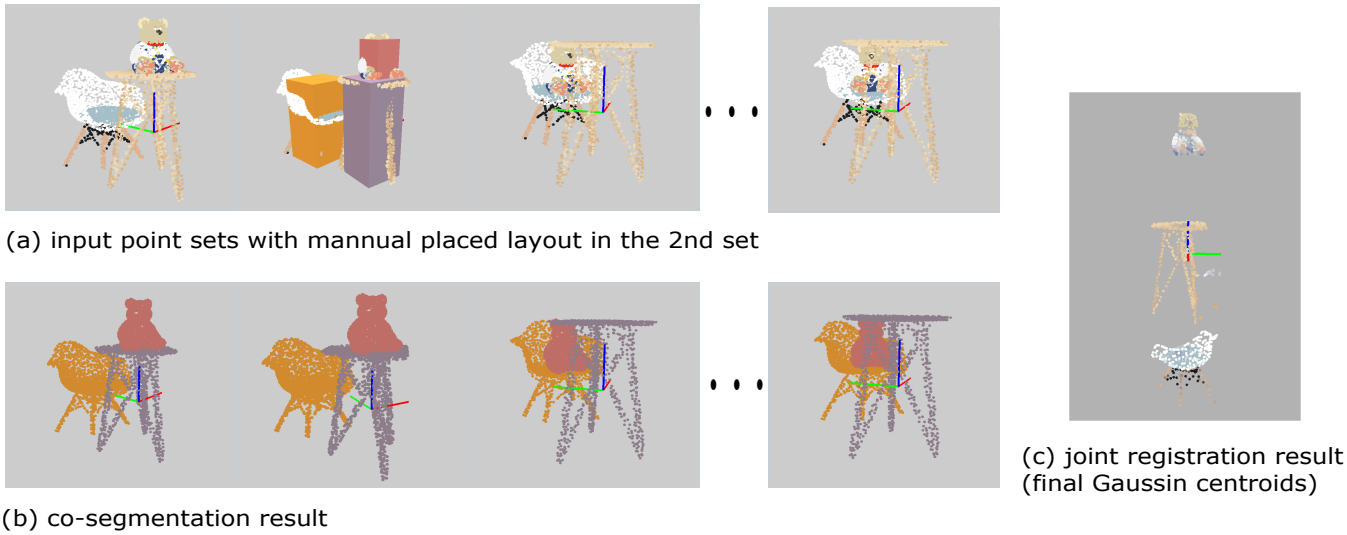

Figure 1: *(a) are input point sets and user have initialized layout for the 2^{nd} set by interactively placing boxes in it. (b) are results of co-segmentation. (c) are the Gaussian centroids in three groups (from top to down) corresponding to three objects*

Abstract

We present a method of joint registration and co-segmentation for point sets of objects. We view the joint registration and co-segmentation as two problems heavily entangled with each other. To model such entangled problems, we treat the input point sets as samples from a generative model and bring up with a novel formulation based on Gaussian mixture model. By maximizing the posterior probability of the samples, we gradually recover the latent object model and object level segmentation and align the objects to the latent model(solve the registration). Alone with the formulation, we design a procedure of interaction that can help users to intuitively initialize the optimization. Our evaluation shows that our novel method is helpful and effective to do the joint registration and co-segmentation on point sets of multiple objects.

Categories and Subject Descriptors (according to ACM CCS): I.3.8 [COMPUTER GRAPHICS]: Applications—I.4.8 [IMAGE PROCESSING AND COMPUTER VISION]: Scene Analysis—Range data

1. Introduction

In many researches and applications of indoor scenes the data of segmented and even annotated 3D indoor scenes are required as either data base or training data (e.g. [NXS12] [DSS12] [FRS*12] [CLW*14] [FSL*15]).
One way to build such database is to interactively compose scenes from 3D shape models resulting in scenes with object segmentation and annotation naturally available, or to mannually segment and annotate existing scenes. This procedure can be tedious and time consuming, despite the efforts to improve the interaction experience(e.g. [MSL*11] [XCF*13]).
Another way is to automatically generate scenes from 3D shape models according to the input RGB or RGB-D images(e.g.

[LZW*15] [CLW*14]). In such methods, a retrievial procedure is usually needed and inevitablely limit the result to a certain set of 3D models despite the actual 3D model in the input images.
We prefer an approach that helps us build such dataset directly from the captured data. One of the major gap between the required data set and available scene capturing framework (e.g. [IKH*11]) is the general object level segmentation. We want to stress that a general object level segmentation problem should not be treated as an equivalence of multilabel classification problem since it is not limited to a certain set of objects. For 3D data, [JGSC15] used some simplified physical prior knowledge (i.e. the block-based stability) to help acheiving the general objectness segmentation, while the work of [XHS*15] proposes a practical and rather complete framework to close the gap between the required data set and available scene capturing method. One of the observation in [XHS*15] is that the motion consistency of rigid object can serve as a strong evidence of general objectness. To exploit this fact, they employ a robot to do proactive push and use the movement tracking to verify and iteratively improve their object level segmentation result. Our work presented in this paper is trying to exploit the same observation from a different approach.
We intend to use the motion consistency in objects that is naturally revealed by human activities along the time. Down to this approach, we are facing the choice of scanning scheme. One way is to record the change of the scene along with the human activities, another is to schedule a daily or even a once every half day sweep to only record the result of human activities but avoid the instant of human motion. The main challenge brought in by the second scheme is that we may not be able to solve the object correspondence by a local search due to the sparse sampling over time, but the very same challenge exists in the first scheme due to the exclusion caused by human bodies not to mention other additional process(e.g. tracking with severe oclussion) needed for human bodies. With the second scanning scheme, our original intention of building 3D scene data set from capturing naturally leads us to the problem of coupled joint registration and co-segmentation.
In this problem, registration and segmentaion are entangled in each other. On one hand the segmentation depends on the registration to connect the point clouds into series of rigid movement so that the objectness segmentation can be done based on the motion consistency, on the other hand, the registration depends on the segmentation to break the problem into a series of rigid joint registration instead of a joint registration with non-coherent point drift(A pair of points is close to each other in one point set but their correspondent pair of points in another point set is far from each other, in other words, the point drift of this pair is non-coherent. (This happens when this pair of points actually belong to different objects.)
To model the problem, we employ a group of Gaussian mixture models and each of these Gaussian mixture models represents a potential object. This model unentangle the registration and segmentation in the way that the segmentation can be done by evaluate the probability of points belongs to the Gaussian mixture models and the registration can be done by evaluate rigid registration against each gaussain mixture models.
In summary our work makes following contributions:
Firstly, as far as we know we are the first work that bring up with the problem of point set joint registration and co-segmentation for indoor scenes.
Secondly, we come up with a Gaussian mixture model based formulation to simultaneously model both the joint registration and co-segmentation problem.
Thirdly, targeting the disadvantages of our formulation, we design a procedure of interaction and provide a practical tool for point set joint registration and co-segmentation based on it. We will release our tool with acceptance of this paper.

2. Related Work

In this section we explain how our work is related to the previous works and how we draw experience from these previous works.

2.1. Point Set Registration with GMM Representation

There are a series of work that uses gaussion mixture model as representation for point set to formulate the registration problem. [MS10] consider the registration of two point sets as a probability density estimation problem. They force the Gaussian mixture model centroids to move coherently as a group to preserve the topological structure of the point sets. Their method is appliable to both rigid registration and non-rigid registration. As we highlighted in Section 1, our problem is different from the non-rigid registration considered in [MS10], the point drift could be non-coherent in our probelm. [JV11] summarized the works for point set registration using Gaussian mixture models and present a unified framework for the rigid and nonrigid point set registration problem. These works select one of the point set as the "model". Unlike these works, [EKBHP14] treats all the point sets as data: they are all realizations of a Gaussian mixture and the registration is cast into a clustering problem. The recent work of [CP16] pushes the idea to the application on a large scale of data. Comparing to these works, our work is most related to [EKBHP14]. Our formulation can be seen as an extention of the formulation of [EKBHP14] to simultaneously handle joint registration and co-segmentation.

2.2. Image segmentation and co-segmentation

[RKB04] is an influential work for interactive image segmentation. It uses two Gaussian mixture model, one for foreground and one for background. To initialize these two Gaussian mixture models, [RKB04] let users place a rectangle that contain the foreground. Our design of interaction draw on the experientce from [RKB04]. The difference is that our interaction is designed for 3D space and can handle multiple objects segmentation rather than foreground-background segmentation. [TSS16] jointly recover cosegmentation and dense per-pixel correspondence in two images. Its co-segmentation is limited to foreground-background segmentation. Our work solve a similar problem for multiple 3D point sets.

2.3. Segmentation from Motion

The idea that motion can be strong hint for segmentation is used in many works. [XHS*15] employs a robot to do proactive push and track the motion to learn object segmentation. [LPR*16] exploit the motion in video and use the motion edge as training data to learn an edge detector for image. These methods lean on the motion that is continuous in time and can be tracked. Our method can handle motion that is not continuous in time.

2.4. 3D Object Recognition based on Correspondence Grouping

By allowing interactively input layout, the joint registration and co-segmentation problem can be treated as a series of 3D object recognition problem in point sets. Our method should be seen as one of the correspondence grouping method. Comparing to the previous methods of [TS10] and [CB07], our method simutaneously solve the problem for multiple target models in multiple scenes.

3. Method Overview

3.1. Problem Statement

Given M point sets with I_m points $V_m = \{\boldsymbol{v}_{mi}\}$ for the m^{th} point set (Figure 1 (a) shows an example of input point sets.), we intend to simutaneously partition the point sets into N objects and find the rigid transformations $\{\phi_{mn}(\boldsymbol{x}) = R_{mn}\boldsymbol{x} + \boldsymbol{t}_{mn}\}$ that transform each object model to its observed position in each input point set. For partition, we output point-wise label vectors $\{\boldsymbol{y}_m\}$ for each input point set to indicate its object partition. (Figure 1 (b) illustrates result label vectors by assigning same color to points with same label.) To establish our formulation, we first summarize the symbols that will be used as in Table 1.

Symbol	Meaning
V	The input point sets.
V_m	The m^{th} input point set.
$\boldsymbol{v}_{mi}$	The i^{th} point of V_m.
$\boldsymbol{f}_{mi}$	The point-wise feature vector of $\boldsymbol{v}_{mi}$.
z_{mi}	The latent parameter for $\boldsymbol{v}_{mi}$.
	$z_{mi} = k$ means v_{mi} is generated by k^{th} Gaussian
Z	$Z = \{z_{mi} \mid m = 1...M, i = 1...I_m\}$
K_{all}	The total number of Gaussian models.
K_n	The number of Gaussian for n^{th} object. $\sum_n^N K_n = K_{all}$
p_k	The weight of k^{th} Gaussian. $\sum_k^{K_{all}} p_k = 1$
$\boldsymbol{x}_k$	The centroid of k^{th} Gaussian.
$\boldsymbol{xv}_k$	The centroid of k^{th} Gaussian for point position.
$\boldsymbol{xf}_k$	The centroid of k^{th} Gaussian for point feature.
Σ_k	The covariance matrix of k^{th} Gaussian.
σ_k	$\Sigma_k = \sigma_k^2 I$. (I is identity matrix here.)
σv_k	Gaussian covariance parameter for point position
σf_k	Gaussian covariance parameter for point feature
ϕ_{mn}	Transformation from n^{th} object to m^{th} input set.
R_{mn}	The rotation matrix for ϕ_{mn}.
$\boldsymbol{t}_{mn}$	The translation vector for ϕ_{mn}.

Table 1: *Table of Symbols Used in the Paper*

3.2. Basic Formulation

To simutaneously model the joint registration and co-segmentation, we come up with a generative model as follows:

$$P(\boldsymbol{v}_{mi}) = \sum_{k=1}^{K_n} p_k \mathcal{N}(\boldsymbol{v}_{mi} \mid \phi_{mn}(\boldsymbol{x}_k), \Sigma_k) \tag{1}$$

which treat the i^{th} observed point v_{mi} from the m^{th} point set as a sample point generated by one of N object models. Each object model is represented by a group of K_n gaussian models.
The model parameters are:

$$\Theta = \{\{p_k, \boldsymbol{x}_k, \Sigma_k\}_{k=1}^{\sum K_n}, \{\phi_{mn}\}_{m=1,n=1}^{MN}\}$$

The problem is how to estimate the parameters for the model to fit all the input point sets. The problem can be solved within the framework of expectation maximization. In particular, we bring in a latent parameter as:

$$Z = \{z_{mi} \mid m = 1...M, n = 1...I_m\}$$

such that $z_{mi} = k (k = 1, 2..., K_{all})$ assigns the observed point v_{mi} to the k^{th} component of Gaussian mixture model. We aim to maximize the expected complete-data log-likelihood:

$$\varepsilon(\Theta|V,Z) = \mathbb{E}_Z[\ln P(V,Z;\Theta)|V] = \sum_Z P(Z|V,\Theta) \ln P(V,Z;\Theta) \tag{2}$$

Such formulation can be seen as an adaption of joint registration formulation in [EKBHP14], upon which we seperate Gaussian models into groups to express multiple objects and the latent parameter Z that assign observed points to gaussian models can naturally indicate the object level segmentation.
By the asssumption of independent and identically distributed of input points, we can rewrite the objective (2) into:

$$\Theta = \arg\max \sum_{mik} \alpha_{mik} (\ln p_k + \ln P(\boldsymbol{v}_{mi} | z_{mi} = k; \Theta)) \tag{3}$$

where $\alpha_{mik} = P(z_{mi} = k | \boldsymbol{v}_{mi}; \Theta)$
By bringing in equation 1 and ingnoring constant terms, we can rewrite the objective as:

$$\Theta = \arg\max \sum_{mik} \alpha_{mik} (||\boldsymbol{v}_{mi} - \phi_{mn}(\boldsymbol{x}_k)||_{\Sigma_k}^2 + \ln|\Sigma_k| - 2\ln p_k) \tag{4}$$

where the $|\cdot|$ denotes the determinant and $||x||_A^2 = x^T A^{-1} x$. It is predefined that x_k is one of the gaussian centroid used to represent n^{th} object, which is why we apply transformation ϕ_{mn} on to the x_k. For the convenience of computation, we restrict the model to isotropic covariances, i.e.,$\Sigma_k = \sigma^2 I$ and I is the identity matrix.
Now, we can optimize this through iterating between estimating α_{mik} (Expectation-step) and maximizing $f(\Theta|V,Z)$ sequentially with respect to each parameters in Θ (Maximization-steps). These steps are:
E-step: this step estimates the posterior probability α_{mik} of v_{mi} to be a point generated by the k^{th} Gaussian model.

$$\alpha_{mik} = \frac{p_k \sigma_k^{-3} exp(-\frac{1}{2\sigma_k^2} ||v_{mi} - \phi_{mn}(\boldsymbol{x}_k)||^2)}{\sum_s^{K_{all}} p_s \sigma_s^{-3} exp(-\frac{1}{2\sigma_s^2} ||v_{mi} - \phi_{mn}(\boldsymbol{x}_s)||^2)} \tag{5}$$

M-step-a:this step update the transformations ϕ_{mn} that maximize $f(\Theta)$, given instant values for α_{mik}, x_k, σ_k. We only consider rigid transformations, making $\phi_{mn}(x) = R_{mn}x + t_{mn}$. The maximizer R_{mn}^*, t_{mn}^* of $f(\Theta)$ is the same with the minimizers of the following constrained optimization problems:

$$\begin{cases} \min_{R_{mn}, t_{mn}} & ||(W_{mn} - R_{mn}X_n - \boldsymbol{t}_{mn}\mathbf{e}^T)\Lambda_{mn}||_F^2 \\ s.t. & R_{mn}^T R_{mn} = I, |R_{mn}| = 1 \end{cases} \tag{6}$$

where Λ_{mn} is $K_n \times K_n$ diagonal matrix with elements $\lambda_{mnk} = \frac{1}{\sigma_k}\sqrt{\sum_i^{I_m} \alpha_{mik}}$,$I_m$ is the number of point for the m^{th} input point set, $X_n = [\boldsymbol{x}_1,\boldsymbol{x}_2,....,\boldsymbol{x}_{K_n}]$ is the matrix stacked by the centroids of gaussian models that are predefined to represent the n^{th} object. $\mathbf{e}^T$ is a vector of ones, $||\cdot||_F$ denotes the Frobenius norm, and $W_{mn} = [w_{m1}, w_{m2}, ..., w_{mk}, ..., w_{mK_n}]$, in which w_{mk} is a weighted point as:

$$w_{mk} = \frac{\sum_{i=1}^{I_m} \alpha_{mik} \boldsymbol{v}_{mi}}{\sum_{i=1}^{I_m} \alpha_{mik}} \tag{7}$$

This problem have a similar solution of in [EKBHP14]. The only difference is that we are estimating the transformation from Gaussian models to the input point sets instead of the transformation from input point sets to Gaussian models, since there are multiple group of $\boldsymbol{x}_k$ corresponding to multiple objects in our Gaussian models. The optimal can be given by:

$$R^*_{mn} = U_{mn} C_{mn} V^T_{mn} \tag{8}$$

$$t^*_{mn} = \frac{1}{tr(\Lambda^2_{mn})}(W_{mn} - R_{mn} X_n)\Lambda^2_{mn}\mathbf{e} \tag{9}$$

where $[U_{mn}, S, V_{mn}] = svd(W_{mn}\Lambda_{mn} P_{mn} \Lambda_{mn} X^T_{mn})$ and $P_{mn} = I - \frac{\Lambda_{mn}\mathbf{e}(\Lambda_{mn}\mathbf{e})^T}{(\Lambda_{mn}\mathbf{e})^T \Lambda_{mn}\mathbf{e}}$,$I$ is identity matrix. $C_{mn} = diag(1, 1, |U_{mn}||V_{mn}|)$.

M-step-b: this step we update the parameters related to the Gaussian mixture model and the indicating vector for object segmentation

$$\boldsymbol{x}^*_k = \frac{\sum_{m=1}^{M}\sum_{i=1}^{I_m} \alpha_{mik}(R^{-1}_{mn} v_{mi} - \boldsymbol{t}_{mn})}{\sum_{m=1}^{M}\sum_{i=1}^{I_m} \alpha_{mik}} \tag{10}$$

where $\boldsymbol{x}_k$ is one of the Gaussian centroids that is predefined to represent n^{th} object.

$$\sigma^{*2}_k = \frac{\sum_{m=1}^{M}\sum_{i=1}^{I_m} \alpha_{mik} ||(v_{mi} - \boldsymbol{t}_{mn} - R^*_{mn}\boldsymbol{x}^*_k)||^2_2}{3\sum_{m=1}^{M}\sum_{i=1}^{I_m} \alpha_{mik}} \tag{11}$$

$$p^*_k = \frac{\sum_{m,i} \alpha_{mik}}{M} \tag{12}$$

$$y_m(i)^* = \arg\max_n \sum_{k=K_{n-1}+1}^{K_n} \alpha_{mik} \tag{13}$$

where $y_m(i)$ is the i^{th} entry of the indicate vector y_m.

3.3. Bilateral Formulation

When considering point-wise features, we can add bilateral terms into the generative model.

$$P(v_{mi}, f_{mi}) = \sum_{k=1}^{K_n} p_k \mathcal{N}(v_{mi}|\phi_{mn}(\boldsymbol{x}\boldsymbol{v}_k), \sigma v_k)\mathcal{N}(f_{mi}|\boldsymbol{x}\boldsymbol{f}_k, \sigma f_k) \tag{14}$$

where f_{mi} is the feature vector for point v_{mi} and xf_k is the feature vector for k^{th} point in latent model. As shown in the formulation, there is no transformation applyed onto xf_k, which means that this formulation is only suitable to the features that is rotation and translation invariant. For example, the point color vector(for all the result in this paper we use RGB color as feature vector) $[red_{mi}, green_{mi}, blue_{mi}]$ is a suitable feature for this formulation.In this formulation $N(v_{mi}|\phi_{mn}(xv_k), \sigma v_k)$ is the spatial term and $N(f_{mi}|xf_k, \sigma f_k)$ is the feature term. For the bilateral formulation, iteration steps will be as follows:

E-step:in this step the calculation of posterior probability need to consider both the spatial term and the feature term.

$$\alpha_{mik} = \frac{p_k P_v(v_{mi}, \phi_{mn}(xv_k), \sigma v_k) P_f(f_{mi}, xf_k, \sigma f_k)}{\sum_s^{K_{all}} p_s P_v(v_{mi}, \phi_{mn}(xv_s), \sigma v_s) P_f(f_{mi}, xf_k, \sigma f_s)} \tag{15}$$

where $P_v(x, y, \sigma) = \sigma^{-3} exp(-\frac{1}{2\sigma^2}||x-y||^2)$ and $P_f(x, y, \sigma) = \sigma^{-D(x)} exp(-\frac{1}{2\sigma^2}||x-y||^2)$ and $D(x)$ means the dimension of the vector x.

M-step-a:for bilateral formulation, this step is the same with the basic formulation and the update can be done as (8) and (9). **M-step-b**:for bilateral formulation, this step need not only update model centroids and variance for spatial term as (10) and (11). but also update the centroids and variance for feature term as in (16) and (17)

$$xf^*_k = \frac{\sum_{m=1}^{M}\sum_{i=1}^{I_m} \alpha_{mik} f_{mi}}{\sum_{m=1}^{M}\sum_{i=1}^{I_m} \alpha_{mik}} \tag{16}$$

$$\sigma f^{*2}_k = \frac{\sum_{m=1}^{M}\sum_{i=1}^{I_m} \alpha_{mik} ||(f_{mi} - xf^*_k)||^2_2}{D(f)\sum_{m=1}^{M}\sum_{i=1}^{I_m} \alpha_{mik}} \tag{17}$$

where $D(f)$ is the dimenssion of feature vectors.

The update of p_k for bialateral formulation is the same as the basic formulation in (12).

3.4. Interaction Design

Unfortunately, there are a large number of parameters that can not be easily initialized in our formulation . In this subsection we first introduce our design of interaction, which is intuitive for users to input the semantic prior this way. We then explain how we can easily initlialize those parameters for our optimization based on the manual input.

As demonstrated in Figure 2, we let user choose one of the point sets and placing and editing boxes in it to indicate the layout for this point set. From this, we can easily initialize the total number of objects N and determine $\{K_n\}$ which is the numbers of Gaussian mixture models used to represent each object. These two paremeters are difficult to be initialized without semantic prior, but with the input of the users we can naturally initialize the N as the number of different color label and the K_n as

$$K_n = \frac{V_n}{\sum V_n} K_{all} \tag{18}$$

in which the V_n represent the total volume of the boxes in the n^{th} color and the K_{all} is initialized as $K_{all} = \frac{median(I_m)}{2}$ and $\{I_m\}$ are point numbers of M input point sets. This is an emperical choice borrowed from [EKBHP14].

The expectation maximizaton framework is easily converged to a local optimal. To cope with this problem we further use this layout

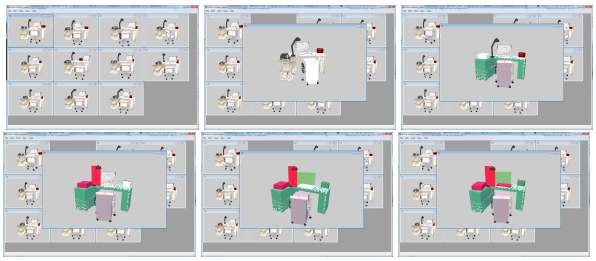

Figure 2: *From the first to the nineth, the nine images show the procedure of interaction: the user pick one point set and place boxes in it to indicate the layout for this point set. The box in white is the box currently under editing. The boxes in other colors are boxes placed to represent object layouts. One color represent one object. The interaction allows multiple boxes to represent same object.(e.g. the desk is represented by three boxes in same color)*

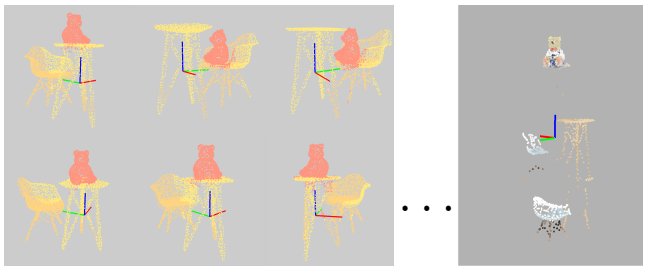

Figure 3: *This figure shows an example result when converges to a local optimal. (a) is the result of segmentation of this local optimal. (b) is the final centroids of latent model. It shows that from top to down the 2nd and 3rd object model both include part of the table and part of the chair.*

(boxes) from interaction as a soft constraint to guide the optimization and constrain the shape of generated object model. Such constraint is enforced by simply altering the posterior probability α_{mik} as

$$\alpha^*_{mik} = \frac{\alpha_{mik}\beta_{mik}}{\sum_{i,k}\alpha_{mik}\beta_{mik}} \tag{19}$$

where the β_{mik} is the prior probability according to the boxes. It is defined as:

$$\beta_{mik} = \begin{cases} 1 & \boldsymbol{v}_{mi} \in B_n \\ \exp(-\frac{\min_{\boldsymbol{v}_{mj}} ||\boldsymbol{v}_{mi}-\boldsymbol{v}_{mj}||_2^2)}{L}) & \boldsymbol{v}_{mi} \notin B_n \text{ and } \boldsymbol{v}_{mj} \in B_n \end{cases} \tag{20}$$

where the B_n is a point set that is enclosed by the boxes used to represent the layout of n^{th} object. The k^{th} Gaussian model is predefined to be one of the Gaussians used to represent n^{th} object. $\min_{v_{mj}} ||v_{mi}-v_{mj}||_2^2$ is actually the squared euclidean distance from point v_{mi} to the point set B_n, as we define the distance from a point to a point set as the minimum distance from the point to any point inside the point set. L here is a constant number with $L = 2r^2$, and r is the meadian of the radius of input point sets. The radius of a input point set is half of length of diagnonal line of its axis aligned bounding box. This alteration on posterior probability is only done with the probability related to the m^{th} point set that have the manual input layout (the boxes) in it. This alteration can help prevent the optimization from converging to a local optimal as in Figure 3. The result from the Figure 3 have the same input and initialization with the result from Figure 1, but it doesn't use the posterior alteration as a soft constraint.

4. Algorithms and Implementation Details

In this section, we summarize the entire algorithm and explain the implemented details.

4.1. Algorithm

Based on our formulation in section 3, our algorithm can be summarized as in Algorithm 1.

Algorithm 1 Joint Registration and Co-segmentation (JRCS)

Input:
$\{V_m\}$:M input 3D point sets
Θ^0:Initial parameters
$\{\beta_{ik}\}_m$:layout based prior
Output:
Θ^q:Final parameters
1. $q \leftarrow 1$
2. **repeat**
3. E-step: Use Θ^{q-1} to estimate α^q_{mik} according to (5) ((15) if use bilateral formulation)
4. alter α^q_{mik} with $\{\beta_{ik}\}_m$ according to (19)
5. M-step-a: Use $\alpha^q_{mik}, \boldsymbol{x}^{q-1}_k$ to estimate $\{R^q_{mn}\}$ and $\{t^q_{mn}\}$ according to (8)(9)
6. M-step-b: Use α^q_{mik}, $\{R^q_{mn}\}$ and $\{t^q_{mn}\}$ to other parameters for Gaussian models according to (10)(11)(12)(13)
7. $q \leftarrow q+1$
8. **until** Convergence
9. **return** Θ^q

4.2. Implementation Details

Initialization of Θ:
We first determine the total number of Gaussian model K_{all} as we explained in subsection 3.4. We set $p_k = \frac{1}{\sum K_n}$ which means each Gaussian model have the same weight at the beginning. We seperate the Gaussian models into N groups to represent N objects. Each group has K_n Gaussian models based on (18). $\{\boldsymbol{x}_k\}_n$ are Gaussian centroids of n^{th} group and they are initialized as a random positions uniformally distributed on the surface of a sphere. The radius of the sphere is chosen as the median of the radius of the input point sets r. The center of the n^{th} spheres is $c_n = (0,0,z_n)$, where

$z_n \in \{-(N-1)r, -(N-3)r, ..., (N-1)r\}$. This means that the object models are vertically arranged in latent space as shown in Figure 1(m)(n) and in Figure 3(b).We choose vertical arrangement for groups of object merely for the convenience of visualization. We choose the sphere as the initial shape so that we can initialize all the R_{mn} to identity matrix. For the $\boldsymbol{t}_{mn}$ we initialize them as $\boldsymbol{t}_{mn} = -\boldsymbol{c}_n$ so that all the object model starting with position at origin point when they are transformed to the space of each input set. However, if the m^{th} input point set has the manually placed layout, we treat the associated $\boldsymbol{t}_{mn}$ differently. For this case we have:

$$\boldsymbol{t}_{mn} = \frac{\sum_{\boldsymbol{v}_{mi} \in B_n} \boldsymbol{v}_{mi}}{N(B_n)} - \boldsymbol{c}_n \tag{21}$$

where $N(B_n)$ is the number of element in B_n and B_n is the point set that is enclosed by the mannual input layout (boxes).

4.3. Hot Intervention Machinism

Our current implementation of optimization is quite slow (fail to converge in interative time) epecially when the point numbers inside each input point set are large and it is possible for our optimization to stuck in a local optimal, requiring the guide from the mannual input. As a compensation for these drawbacks. We implement a hot intervention machinism, allowing the mannaul input take effect even after the optimization has started. Theoretically, this is possible due to the i.i.d assumption. This assumption makes it possible for the calculation of posterior probability being independent for each input point set. Even after the optimization is started, we can still allow the user to add more layouts for other point sets and the program can do the same alteration as (19) in the next iteration. The Figure 4 shows how the users can use the hot intervention machinism within our tool.

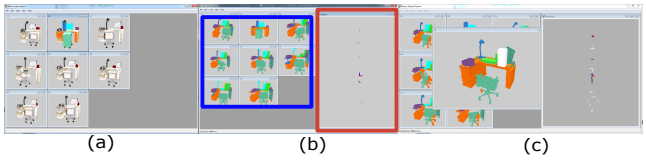

Figure 4: *This figure shows the hot intervention machinism. (a) is the input point sets with mannually placed layout in 2rd point set. (b) shows that from the region highlighted by blue rectangle the user can see the instant result of segmentation and from the region highlighted by the red rectangle the user can see the space of object model(the shape of the centroids of the Gaussian models). (c) shows that the user picks another input point set and add more boxes targeting the incorrect segmentation to further guide the optimization.*

5. Experiment and Discussion

5.1. Evaluation for Co-segmentation on Synthetic Data

From the perspective of co-segmentation, we evaluate our algorithm on synthetic data of indoor scenes. To estimate the power of the algorithm we only input layout for one point set in each group

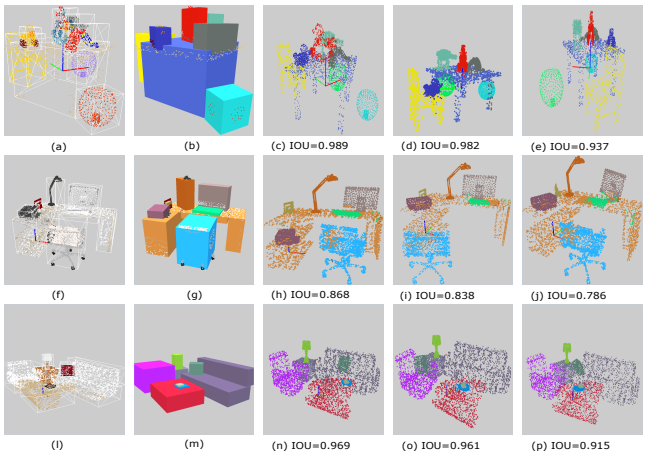

Figure 5: *Three rows in the figure shows segmentation evaluations on three groups of synthetic data(child table, office desk, living room). Each group of data have 13 point sets. The first column are examples of point sets for each group of data. The second column are mannual placed layout for each group of data. The 3^{rd} column shows the segmentation result with maximum IOU scores in the groups. The 4^{th} column shows the segmentation result with median IOU scores in the groups. The 5^{th} column shows the segmentation result with minimum IOU scores in the groups.*

for initialization and do not use the hot intervention mechanism. For numercial estimation, we calculate the intersection over union (IOU) scores for the result segmentation against ground-truth segmentation. We generate three group of synthetic point sets and each group have 13 point sets as inputs. The Figure 5 shows the result of the evalution.

From the evaluation, we want to discuss one interesting observation:

For all three groups, the point set with highest IOU score is not the same as the point set equiped with mannually placed layout.In other words , the point sets from the 3^{rd} column in Figure 5 are not the same point sets from 2^{nd}. We believe this is because that the mannually placed layout is not accurate respecting to point-wise segmentation. At early iterations of the optimization, the alteration in (19) can serve as a soft constraint to help constratining the shape of object, but in the final iterations the alteration will obstruct the further improvement of segmentation for the correspondent point set.

5.2. Evaluation for Joint Registration on Synthetic Data

From the perspective of joint registration, we evaluate the result by transfering the point cloud of objects to each input point set based on result $\{\phi_{mn}\}$ and calculating the average distance from a point to its true correspondent point for each input point set. We use this average distance as fitness error to evaluate the registration quality respect to each input set. Table 2 shows the result of this evaluation.

For this evaluation we want to discusss that:

We find that even the input set with high IOU scores in segmen-

Dataset	Maximum Error	Median Error	Minimum Error
Child Table	0.0715	0.0112	4.91e-005
Office Desk	0.189	0.0618	0.00518
Living Room	0.132	0.0563	0.0301

Table 2: *Error for Registration for the Same Three Groups of Data in Figure 5 The unit of these numbers is 1 meter*

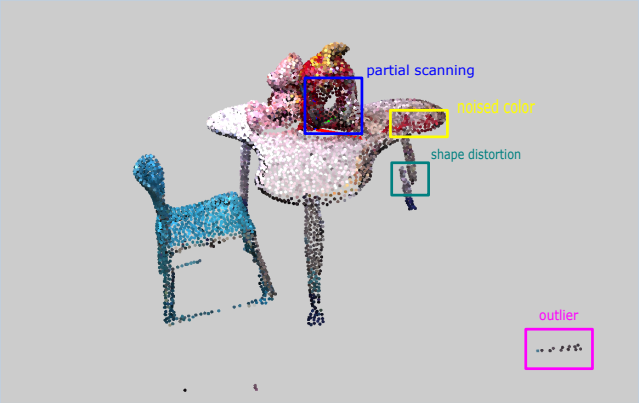

Figure 7: *This figure highlights the common challenges on real data.*

tation can result in high fitness error. We believe this is due to the symmetric and near-symmetric objects in the scene. For symmetric objects, even the registration is correct the distance from a point to its true correspondent point can be high, since the rotation in registration result can be different from the one we use to generate this synthetic data. For near-symmetric objects, the registration often stuck in a local optimal and result in high IOU score but high fitness error.

5.3. Test On Real Data

To capture real data we employ the method of [NZIS13] and use plane fitting to remove walls and floors. We then transfer the mesh into point set with the sampling process from [CCS12]. We test our algorithm on a group of real data, Figure 6 shows the complete result of our test. On real data, there are noised color, shape distortion, partial scanning and outliers as they are highlighted in Figure 7. From Figure 6(e), we can see that all input point sets are partitioned into objects. From the Figure 6(g), we can verifys that the object from each input set are aligned together by the result transformation.

5.4. Limitations and Future Work

The biggest problem holding us back is the time performance of current implementation of our tool. Due to i.i.d. assumption most calculation of our algorithm can be parallelized. We plan to implement a new version on GPU cluster so that we can explore more potentials of our algorithm, for example,Try to integrate semantic feature vectors (generated by neural networks) into it and try it on a scene of larger scale like [CP16]. As advocated in the recent work of [EKT*16], it may be a good idea to do the joint registration and co-segmentation with hierarchical GMM representation when applyed to scenes on a larger scale.

References

[CB07] CHEN H., BHANU B.: 3d free-form object recognition in range images using local surface patches. *Pattern Recogn. Lett. 28*, 10 (July 2007), 1252–1262. URL: http://dx.doi.org/10.1016/j.patrec.2007.02.009, doi:10.1016/j.patrec.2007.02.009. 3

[CCS12] CORSINI M., CIGNONI P., SCOPIGNO R.: Efficient and flexible sampling with blue noise properties of triangular meshes. *IEEE Transactions on Visualization and Computer Graphics 18*, 6 (June 2012), 914–924. URL: http://dx.doi.org/10.1109/TVCG.2012.34, doi:10.1109/TVCG.2012.34. 7, 8

[CLW*14] CHEN K., LAI Y.-K., WU Y.-X., MARTIN R., HU S.-M.: Automatic semantic modeling of indoor scenes from low-quality rgb-d data using contextual information. *ACM Trans. Graph. 33*, 6 (Nov. 2014), 208:1–208:12. URL: http://doi.acm.org/10.1145/2661229.2661239, doi:10.1145/2661229.2661239. 1, 2

[CP16] CAMPBELL D., PETERSSON L.: Gogma: Globally-optimal gaussian mixture alignment. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2016), pp. 5685–5694. doi:10.1109/CVPR.2016.613. 2, 7

[DSS12] DEMA M. A., SARI-SARRAF H.: 3d scene generation by learning from examples. In *Multimedia (ISM), 2012 IEEE International Symposium on* (Dec 2012), pp. 58–64. doi:10.1109/ISM.2012.19. 1

[EKBHP14] EVANGELIDIS G. D., KOUNADES-BASTIAN D., HORAUD R., PSARAKIS E. Z.: *A Generative Model for the Joint Registration of Multiple Point Sets*. Springer International Publishing, Cham, 2014, pp. 109–122. URL: http://dx.doi.org/10.1007/978-3-319-10584-0_8, doi:10.1007/978-3-319-10584-0_8. 2, 3, 4

[EKT*16] ECKART B., KIM K., TROCCOLI A., KELLY A., KAUTZ J.: Accelerated generative models for 3d point cloud data. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2016), pp. 5497–5505. doi:10.1109/CVPR.2016.593. 7

[FRS*12] FISHER M., RITCHIE D., SAVVA M., FUNKHOUSER T., HANRAHAN P.: Example-based synthesis of 3d object arrangements. *ACM Trans. Graph. 31*, 6 (Nov. 2012), 135:1–135:11. URL: http://doi.acm.org/10.1145/2366145.2366154, doi:10.1145/2366145.2366154. 1

[FSL*15] FISHER M., SAVVA M., LI Y., HANRAHAN P., NIESSNER M.: Activity-centric scene synthesis for functional 3d scene modeling. *ACM Trans. Graph. 34*, 6 (Oct. 2015), 179:1–179:13. URL: http://doi.acm.org/10.1145/2816795.2818057, doi:10.1145/2816795.2818057. 1

[IKH*11] IZADI S., KIM D., HILLIGES O., MOLYNEAUX D., NEWCOMBE R., KOHLI P., SHOTTON J., HODGES S., FREEMAN D., DAVISON A., FITZGIBBON A.: Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, 2011), UIST '11, ACM, pp. 559–568. URL: http://doi.acm.org/10.1145/2047196.2047270, doi:10.1145/2047196.2047270. 2

[JGSC15] JIA Z., GALLAGHER A. C., SAXENA A., CHEN T.: 3d reasoning from blocks to stability. *IEEE Transactions on Pattern Analysis and Machine Intelligence 37*, 5 (May 2015), 905–918. doi:10.1109/TPAMI.2014.2359435. 2

[JV11] JIAN B., VEMURI B. C.: Robust point set registration using gaussian mixture models. *IEEE Transactions on Pattern Analysis and*

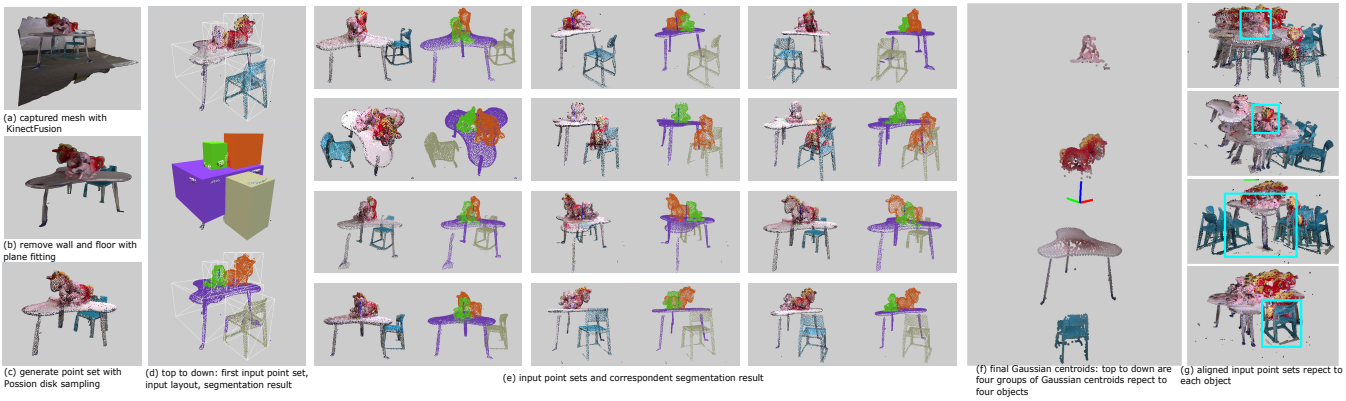

Figure 6: *This figure shows our test on a real data. (a)(b)(c) shows how we capture real data with method in [NZIS13] and convert to input point set with sampling method [CCS12]. (d) shows the first input point set, the input layout on it and its segmentation result. (e) shows pairs of other input point sets and corresponding segmentation result. (f) shows the final Gaussian centroids. (g) verifys the registration result by aligning input point sets respecting to each object. The light blue rectangle highlights the object that is aligned together.*

Machine Intelligence 33, 8 (Aug 2011), 1633–1645. doi:10.1109/TPAMI.2010.223. 2

[LPR*16] LI Y., PALURI M., REHG J. M., DOLLAR P., UNDEFINED, UNDEFINED, UNDEFINED, UNDEFINED: Unsupervised learning of edges. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 00* (2016), 1619–1627. doi:doi.ieeecomputersociety.org/10.1109/CVPR.2016.179. 2

[LZW*15] LIU Z., ZHANG Y., WU W., LIU K., SUN Z.: Model-driven indoor scenes modeling from a single image. In *Graphics Interface Conference* (2015). 2

[MS10] MYRONENKO A., SONG X.: Point set registration: Coherent point drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence 32*, 12 (Dec 2010), 2262–2275. doi:10.1109/TPAMI.2010.46. 2

[MSL*11] MERRELL P., SCHKUFZA E., LI Z., AGRAWALA M., KOLTUN V.: Interactive furniture layout using interior design guidelines. *ACM Trans. Graph. 30*, 4 (July 2011), 87:1–87:10. URL: http://doi.acm.org/10.1145/2010324.1964982, doi:10.1145/2010324.1964982. 1

[NXS12] NAN L., XIE K., SHARF A.: A search-classify approach for cluttered indoor scene understanding. *ACM Trans. Graph. 31*, 6 (Nov. 2012), 137:1–137:10. URL: http://doi.acm.org/10.1145/2366145.2366156, doi:10.1145/2366145.2366156. 1

[NZIS13] NIEβNER M., ZOLLHÖFER M., IZADI S., STAMMINGER M.: Real-time 3d reconstruction at scale using voxel hashing. *ACM Trans. Graph. 32*, 6 (Nov. 2013), 169:1–169:11. URL: http://doi.acm.org/10.1145/2508363.2508374, doi:10.1145/2508363.2508374. 7, 8

[RKB04] ROTHER C., KOLMOGOROV V., BLAKE A.: "grabcut": Interactive foreground extraction using iterated graph cuts. In *ACM SIGGRAPH 2004 Papers* (New York, NY, USA, 2004), SIGGRAPH '04, ACM, pp. 309–314. URL: http://doi.acm.org/10.1145/1186562.1015720, doi:10.1145/1186562.1015720. 2

[TS10] TOMBARI F., STEFANO L. D.: Object recognition in 3d scenes with occlusions and clutter by hough voting. In *2010 Fourth Pacific-Rim Symposium on Image and Video Technology* (Nov 2010), pp. 349–355. doi:10.1109/PSIVT.2010.65. 3

[TSS16] TANIAI T., SINHA S. N., SATO Y.: Joint recovery of dense correspondence and cosegmentation in two images. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2016). 2

[XCF*13] XU K., CHEN K., FU H., SUN W.-L., HU S.-M.: Sketch2scene: Sketch-based co-retrieval and co-placement of 3d models. *ACM Trans. Graph. 32*, 4 (July 2013), 123:1–123:15. URL: http://doi.acm.org/10.1145/2461912.2461968, doi:10.1145/2461912.2461968. 1

[XHS*15] XU K., HUANG H., SHI Y., LI H., LONG P., CAICHEN J., SUN W., CHEN B.: Autoscanning for coupled scene reconstruction and proactive object analysis. *ACM Trans. Graph. 34*, 6 (Oct. 2015), 177:1–177:14. URL: http://doi.acm.org/10.1145/2816795.2818075, doi:10.1145/2816795.2818075. 2