A *Search-Classify* Approach for Cluttered Indoor Scene Understanding

Liangliang Nan[1] Ke Xie[1] Andrei Sharf[2]

[1] Shenzhen VisuCA Key Lab/SIAT [2] Ben Gurion University

Figure 1: *A raw scan of a highly cluttered indoor scene is given (left). Applying our* search-classify *method, we segment the scene into meaningful objects (middle: chairs (blue) and tables (purple)), followed by a template deform-to-fit reconstruction (right).*

Abstract

We present an algorithm for recognition and reconstruction of scanned 3D indoor scenes. 3D indoor reconstruction is particularly challenging due to object interferences, occlusions and overlapping which yield incomplete yet very complex scene arrangements. Since it is hard to assemble scanned segments into complete models, traditional methods for object recognition and reconstruction would be inefficient. We present a *search-classify* approach which interleaves segmentation and classification in an iterative manner. Using a robust classifier we traverse the scene and gradually propagate classification information. We reinforce classification by a template fitting step which yields a scene reconstruction. We *deform-to-fit* templates to classified objects to resolve classification ambiguities. The resulting reconstruction is an approximation which captures the general scene arrangement. Our results demonstrate successful classification and reconstruction of cluttered indoor scenes, captured in just few minutes.

Keywords: point cloud classification, scene understanding, reconstruction

Links: DL PDF

1 Introduction

Processing of 3D digital environments is an increasingly important research problem, motivated by ambitious applications that aim to build digital copies of cities (e.g., Microsoft Virtual Earth and Google Earth). Advances in laser scanning technology and recent proliferation of GIS services have been driving a strong trend towards processing and modeling of large-scale outdoor scenes based on aerial photography and street-level laser scanners.

Surprisingly, scanned 3D interiors pose a much more difficult problem. In contrast to building exteriors which are relatively piecewise flat, interior scenes are more complicated in their 3D structures. Rooms are densely populated with objects in arbitrary arrangements, for example, chairs pulled underneath tables, semi-open drawers, and etc. Additionally, proper acquisition of interior spaces is challenging in many ways. For example, when scanning densely populated indoor scenes, significant parts remain occluded in all views, yielding a partial representation. The prevalence of thin structures such as doors, walls and table legs poses another significant challenge since their acquisition requires high sampling rate relative to the scale of the scene. Hence, traditional modeling techniques would perform relatively poor on interior scenes, due to typical clutter, missing parts and noise (see Figure 1).

3D scans of large scale environments are relatively new and were made possible due to recent progress in scanning technology. Several algorithms have been proposed for modeling [Sinha et al. 2008; Schnabel et al. 2009; Nan et al. 2010; Livny et al. 2010], and object segmentation in scanned outdoor scenes [Frome et al. 2004b; Anguelov et al. 2005; Golovinskiy et al. 2009]. However, only very recently, researchers have been addressing the complex challenges present in cluttered scanned indoor scenes [Kim et al. 2012; Shao et al. 2012].

We develop a fully automatic algorithm that is capable of understanding and modeling raw scans of cluttered indoor scenes (see Figures 1, 2). In our method, we define a set of shape features that are used for supervised learning of a classifier. We argue that object classification cannot be directly applied to the scene, since object segmentation is unavailable. Moreover, the segmentation of the scene into objects is as challenging as the classification since spatial relationships between points and patches are neither complete nor reliable. For example, it is practically impossible to segment and attribute legs of chairs drawn close together. Thus, classification and segmentation together, constitute a typical *chicken-egg* problem.

Our key idea is to interleave the computations of segmentation and classification of the scene into meaningful parts. We denote this approach *search-classify*, since we search for meaningful segments using a classifier that estimates the probability of a segment to be

ACM Reference Format
Nan, L., Xie, K., Sharf, A. 2012. A *Search-Classify* Approach for Cluttered Indoor Scene Understanding. *ACM Trans. Graph. 31* 6, Article 137 (November 2012), 10 pages. DOI = 10.1145/2366145.2366156 http://doi.acm.org/10.1145/2366145.2366156.

Figure 2: *A zoom into the cluttered region of Figure 1 (left), reveal that accurate segmentation and classification are challenging, even for human perception. We initially over segment the scene (mid-left) and* search-classify *meaningful objects in the scene (mid-right), that are reconstructed by templates (right) overcoming the high clutter.*

part of an object. Global fitting techniques such as Hough transform [Vosselman et al. 2004] and RANSAC [Schnabel et al. 2007] aim at fitting a primitive shape to partial or noisy data using a robust process. Nevertheless, ours is a much harder problem since shape is not known apriori and it lies in a cluttered scene with challenging foreground/background separation.

In our algorithm, we traverse the scene and incrementally accumulate patches into meaningful labeled objects. In each step, we query accumulated parts with our classifier and obtain a set of likelihood probabilities for different classes. We proceed by growing regions with highest likelihood probability. We further reinforce classification by template fitting in order to solve ambiguous cases. Given a classified shape, we fit it a deformable template. Using fitting error, we can detect outliers and misclassified parts and re-iterate the search-classify process. An immediate outcome of this step is an approximated scene reconstruction by deformed templates which captures the general objects' arrangements.

In a nutshell, our method consists of the following steps. We develop a set of features and train a classifier from a large set of examples for recognition of several specific object classes. Given a raw scan of an indoor scene we initially over-segment it into piecewise smooth patches. We *search-classify* the scene by iteratively accumulating patches that form regions with high classification likelihood. This process performs until the whole scene is classified into meaningful objects and outliers. Nevertheless, classified objects can overlap , contain outliers and ambiguities. Therefore, we reinforce the classification step with a template fitting step. Thus, we deform templates to fit to the classified point cloud and select the best matching template. The fitting process reinforces or undermines classification leading to a consistent scene understanding and reconstruction.

Our paper makes the following novel contributions:

- Presenting a *search-classify* approach for interleaving segmentation and classification in cluttered scanned indoor scenes. In contrast to traditional techniques, we do not decouple segmentation and classification apart. Instead, we detect meaningful objects in the scene by growing maximum likelihood regions based on a learned classifier.

- Our scene understanding and fitting contributes an approximate reconstruction of the global scene which conveys the general arrangement and interrelations in the scene. I.e., instead of aiming for exact 3D reconstruction which is impractical with such data quality, we loosely match the local geometry with deformable templates of the same class.

- We utilize a template fitting method for reinforcement of scene recognition and ambiguity resolution.

2 Related Work

Our work essentially bridges between scene understanding and reconstruction. We divide our discussion on related work in two main parts, focusing on 3D scene classification and reconstruction of large-scale digital scenes.

Scanned Scene Understanding The problem of object recognition in 2D images has been extensively researched in computer vision for many years [Ullman 1996; Belongie et al. 2002a; Fei-Fei et al. 2007; Lowe 2004; Viola and Jones 2004]. With the current proliferation of scanning devices (e.g. Kinnect©) and acquisition projects (e.g., Google Earth), research has also transitioned to object recognition in 3D scanned data. Since our work focuses on 3D scanned indoor scenes, and due to large dissimilarities between 2D and 3D data characteristics, we will naturally restrict our discussion to the 3D domain.

In recent years, object recognition has been explored using scanned depth information combined with texture(RGB-D) [Quigley et al. 2009; Lai and Fox 2010; Lai et al. 2011]. While these methods learn scanned object classifiers from a training set, they assume very simple scenes with one or very few objects. Thus, segmentation into meaningful parts is simple and is either ignored [Quigley et al. 2009] or approached using primitive fitting and background subtraction [Lai and Fox 2010; Lai et al. 2011]. In contrast, for cluttered indoor scenes, segmentation is non-trivial and cannot be solved using existing techniques.

Golovinskiy et al. [2009] introduced a classification algorithm for outdoor environments based on foreground/background separation and supervised learning classification. Since indoor scenes are more complex, cluttered, with large missing parts, segmentation into meaningful objects is a very difficult problem. In fact, we claim that classification and segmentation of scanned indoors are interdependent problems. Hedau et al. [2010] introduced a high-level object classifier for indoor scenes which is based on texture information and 3D box geometry around objects. Nevertheless, assuming a correct segmentation of the scene into boxes, is difficult in clutter scenes. Silberman and Fergus [2011] presented an algorithm for indoor scene segmentation which uses graph-cut to propagate classification labels of features to the full scene. Cluttered indoor scenes consist of poor connectivity due to missing parts and outliers, thus a global segmentation solution such as graph-cut typically reaches local minima. Our segmentation performs by an iterative growing process reinforced by classification. Koppula et al. [2011] addressed semantic labeling of 3D indoor scenes by fusing color, depth and contextual information together. They developed a classifier that performs on an over segmented low-level scene representation. In contrast, we seek for meaningful segments in the scene by a controlled region growing process which interleaves classification and segmentation.

There has been considerable work on defining and computing shape

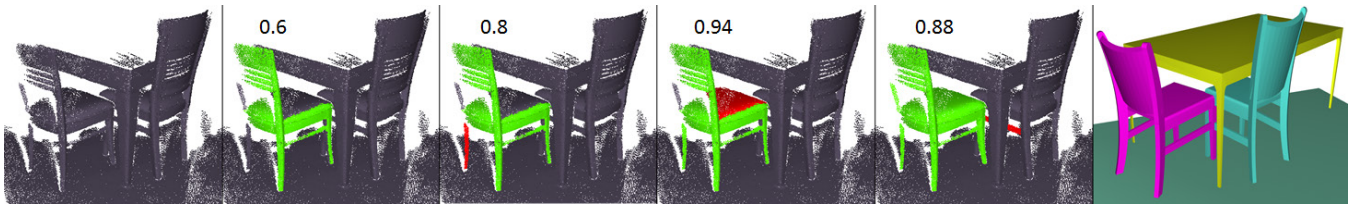

Figure 3: Search-classify *overview. Left-to-right, starting from a raw scan, we randomly select a patch triplet with high classification likelihood value (green). In each iteration we grow by adding one neighbor patch (red). We do not add patches that decrease classification likelihood (chair bar yielding 0.88). Finally, we deform templates to fit points and reinforce classification.*

descriptors for 3D point clouds [Johnson and Hebert 1999; Belongie et al. 2002b; Frome et al. 2004a]. These works focus on defining discriminative and invariant shape descriptors for single object classification. Recently, spin images descriptors were used by Matei et al. [2006] to categorize 3D point cloud objects.

Several papers use statistical models to classify different point descriptors. Markov Random Fields were used in this context by computing the classification of a point from its local descriptor and neighbors [Anguelov et al. 2005; Munoz et al. 2009]. Galleguillos et al. [2008] and Xiong et al. [2010] used conditional random fields for patch classification based on contextual relations. In contrast, we perform *search-classify* to detect and label full objects. Recently, Shotton et. al [2011] presented an efficient body-parts recognition approach using low level per-pixel classifiers trained on randomized decision forests. While this method works well for body parts, our problem consists of larger class variations (e.g. chairs, sofas, tables, cabinets, monitors, etc.) and higher clutter resulting in self occlusions and missing parts. Fisher et. al [2010; 2011] showed an efficient graph representation for synthetic indoor scenes that facilitates the semantic relationships between objects and can be applied to scene similarity and querying computations.

Data-driven Reconstruction Much of the prior work on large-scale scene reconstruction focus on outdoor urban environments, providing automatic and interactive reconstruction solutions from 3D point clouds [Nan et al. 2010; Zheng et al. 2010; Shen et al. 2011], collections of photos [Werner and Zisserman 2002; Dick et al. 2004; Goesele et al. 2007; Sinha et al. 2008; Xiao et al. 2008; Furukawa et al. 2009] or multi-view video [Pollefeys et al. 2008]. Few works approached the reconstruction challenge using prior fitting. Gal et al. [2007] fitted local basic shapes to scans via partial matching. Schnabel et al. [2009] presented a hole-filling algorithm that is guided by primitive detection in the input. While these works locally fit basic shapes to the point cloud, ours is a more challenging problem. We do not know apriori what is the object's shape nor its position in the scene. Thus we need to segment and classify the point cloud prior to a deformable template fitting step. Recently, Li et al. [2011] introduced a method that simultaneously recovers a set of locally fitted primitives and their accurate global mutual relations in man-made objects. They balanced between local primitive fitting and global relations through an iterative optimization process.

Our template fitting algorithm is inspired by recent works of Xu et al. [2010; 2011]. In [Xu et al. 2010], co-segmented shapes are deformed by scaling corresponding parts in the source and target models. In [Xu et al. 2011], a model is deformed to fit a silhouette target while using high-level structural controllers. We continue this trend and deform a template to fit the imperfect point cloud in a constrained manner using non-rigid ICP and deformation energy minimization.

3 Overview

The key-idea underlying our *search-classify* approach is a controlled region growing process which searches for meaningful objects in the scene by accumulating surface patches with high classification likelihood. We reinforce recognition with a template fitting step where templates are deformed-to-fit classified objects. These two algorithmic components perform in a feedback loop, where initial classification is refined by template fitting which in turn is reevaluated by classification (see Figure 4).

Scene understanding includes two intricate subproblems: objects separation and objects classification. This constitutes a *chicken-egg* type of problem since object classification requires its separation from background and other objects. Conversely, to segment an object in a noisy imperfect scene, prior knowledge of object's type and shape is needed. We solve this problem using a *search-classify* region-growing process which traverses the scene.

In the preprocessing stage, we compute a classifier by learning shape features from a training set. Give a raw scan of an indoor scene, we initially over-segment the scene into smooth patches and compute an adjacency graph between parts. Our scene understanding problem reduces to finding disjoint sets in this graph with maximum classification likelihood. These sets define objects in the scene that have good classification recall of one of the categories. We start from a set of random seeds defined by patch triplets. Region growing performs from the initial seeds, by traversal of their adjacency graph and accumulating segments into significant objects. For each set of segments (representing a potential object), we attempt to accumulate adjacent segments by querying our classifier with the new set for likelihood probability value. We grow a set if its likelihood value is non-decreasing (see Figure 3).

The above segmentation process is not perfect since objects may still overlap due to ambiguities in cluttered regions. Our feedback loop utilizes a deformable template fitting step to reinforce or undermine segmentation and classification results. We fit a deformable template to the classified point cloud aiming at minimizing their one-sided Hausdorff distance (points to template). Thus, incorrectly segmented parts (outliers) will have a low fitting score to template (see red loop in Figure 4). Template deformation is computed using local scale controls which deforms the template via local scaling to the nearest points. This is a part-aware non-homogeneous scaling deformation we call "deform-to-fit".

4 Algorithm Details

Our algorithm consists of the following fundamental tasks:

- Feature set definition
- Supervised learning of classifier
- Scene understanding using *search-classify* region growing

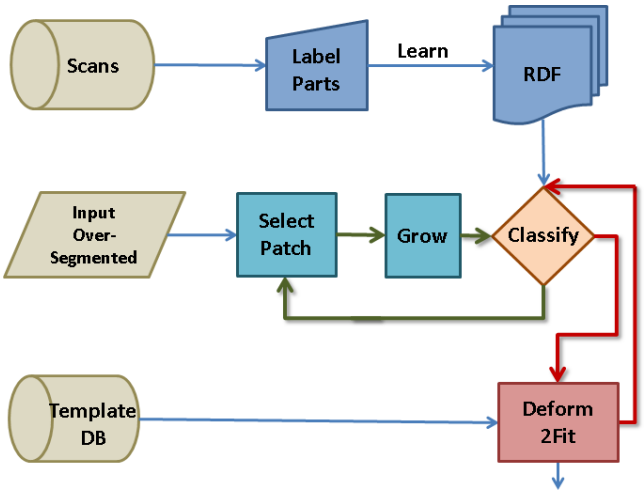

Figure 4: *Block-diagram overview of our method.*

- Deformable template fitting

In the following, we describe the technical details of our algorithm.

Point cloud features In order to separate and classify objects in raw scans, we first need to define a set of descriptive features that provide good separability and discernibility characteristics with regards to scanned objects in the scene. Local descriptors such as curvature, spin images [Johnson and Hebert 1999] or SIFT [Lowe 2004] are inefficient for our data, resulting in redundant features due to the high clutter and noise levels.

Our aim is at defining features with the following characteristics: global, good separation, generic enough for indoor scenes and fast computation time. Since the shape descriptor will be utilized in our *search-classify* controlled region growing process, we are not concerned with descriptor robustness to missing data and noise.

As observed by Fu et al. [2008], there is a strong correlation between functional parts in man-made objects, their geometry and their general upward orientation. Similarly, we observe that man-made objects consist of a natural segmentation along their upward orientation which relates to their functional parts (e.g. table legs, top, etc.). Therefore, we make a simplifying assumption of having a floor plane which defines the upward orientation of objects. Floor information can be easily extracted from camera gyroscope or by analyzing major planes in the scene. We segment each object into horizontal slabs (see Figure 5) by analyzing point distribution along the upward axis. We project all points onto the corresponding upward axis, and detect jumps in the point distribution gradient (i.e. zero crossings in the 2nd derivative). In our experiments, it was sufficient to segment the objects into at most three meaningful horizontal slabs, hence, we detect the three strongest zero crossings passing a threshold.

We utilize the horizontal segmentation into slabs to compute discriminative features as follows:

- oriented bounding box height-size ratio: (height B_h, width B_w and depth B_d) $B_h/\sqrt{B_w \cdot B_d}$
- object's top layer aspect ratio: B_h^t/B_w^t, B_h^t/B_d^t
- object's mid layer aspect ratio: B_h^m/B_w^m, B_h^m/B_d^m
- object's bottom layer aspect ratio: B_h^b/B_w^b, B_h^b/B_d^b

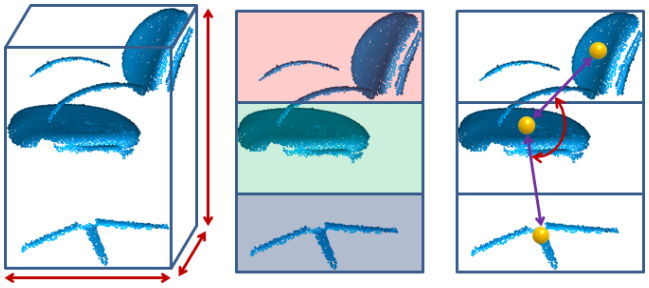

Figure 5: *Features used for point cloud classification. Left-to-right, bounding box measures, detected horizontal slabs by change in point distribution along upward position, angle between slabs COMs.*

- bottom-top size ratio: $\sqrt{B_w^b \cdot B_d^b}/\sqrt{B_w^t \cdot B_d^t}$
- mid-top size ratio: $\sqrt{B_w^m \cdot B_d^m}/\sqrt{B_w^t \cdot B_d^t}$
- change in center of mass (COM) along horizontal slabs, computed as angles between slabs COM's: $\angle_1(B_{com}^t, B_{com}^m)$, $\angle_2(B_{com}^b, B_{com}^m)$

For each dimension of the feature vector, we normalize it to [0, 1] and define an out-of-range value of 200 for missing features in the vector.

A Randomized Decision Forest Classifier Randomized decision forests (RDF) [Breiman 2001] have proven to be efficient multi-class classifiers for many tasks [Shotton et al. 2008; Shotton et al. 2011]. A key feature of RDF classifiers is that they can handle missing data in the classification query in a straightforward manner. Specifically, if one of the features is missing, we simply provide with an *out-of-range* value for that feature in the feature vector. Thus, while traversing the trained decision forest, it will simply ignore these *out-of-range* values yielding a correct classification result.

Handling missing data in the classification process is an important characteristic for our method, since large parts of the object are missing in the scanned data as well as not known apriori in the classification process.

Here, we use supervised learning to train an RDF classifier with various indoor objects. We define an RDF as an ensemble of T decision trees, each consisting of interior split and leaf nodes. Each split node consists of a feature θ and a threshold τ. To classify an object defined as a set of points x in the point cloud, we compute our feature vector and starting at the root, we repeatedly branch left or right according to the comparison of each feature θ to threshold τ. At the leaf node of tree t, a learned distribution $P_t(c|S, x)$ over different labels c is stored. The distributions are averaged together for all trees in the forest to give the final classification:

$$P(c|x) = \frac{1}{T}\sum_{t=1}^{T} P_t(c|x)$$

The decision tree is computed by selecting a set of features and thresholds that maximize separation of the given training set (see Figure 6). Each tree is trained on a random subset of labeled scanned objects. In our implementation we define 50 decision trees and for each tree, we randomly use 30% of the data for its training. Our training set consists of manually segmented and labeled scans of roughly 1000 different objects (e.g. 20 beds, 110 cabinets, 510

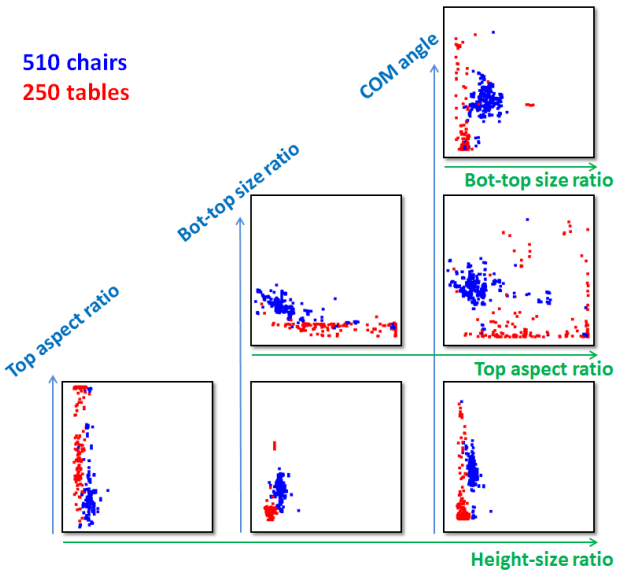

Figure 6: *Visualization of joint distribution of several feature pairs using our RDF classifier, showing clear separation between chairs (blue) and tables (red).*

chairs, 40 monitors, 250 tables, etc.). We choose to train our classifier on sets consisting of both synthetic and scanned objects (which are possibly imperfect), thus to provide with better, close to reality object distributions.

Search-Classify: Segmentation and Classification via Region Growing Given a raw scan of an indoor scene S, we would like to understand it by detecting and labeling meaningful objects. This task involves solving two problems: segmentation of objects from their background and from other objects and their classification. These problems are heavily interdependent: knowing the object type and shape apriori allows a specified search of it and its segmentation; having a segmentation into meaningful objects allows their classification. Nevertheless, a scanned indoor scene consists of a large number of objects with unknown shapes and no clear segmentation.

We initially oversegment the scan S into a set of piecewise smooth patches $P = \bigcup_i (p_i)$ with smooth varying normals by considering k-closest points for each point. We use a normal variation threshold of $n_i \cdot n_j > 0.8$ and with $k = 6$ under a distance threshold $d < 1cm$. The resulting patches are not accurate nor they need to be as we use over segmentation to reduce data complexity. We compute an adjacency graph $G(E, V)$ with nodes $v_i \in V$ corresponding to patches $p_i \in P$ and edges E connecting close patches using Euclidean distance d_{euc} from patch centers with threshold $d_{euc} < 15cm$ (see Figure 7).

The *search-classify* procedure starts by selecting m random patch triplets $(p_i, p_j, p_k) \in P$. Similar to randomized fitting methods (e.g. RANSAC), we analogously test for classification likelihood of each patch triplet. Thus, a patch triplet (p_i, p_j, p_k) (also denoted as object seed $O_l = \bigcup(p_i, p_j, p_k)$) with good classification likelihood to one of the trained object has high probability to be part of this object class in the point cloud. We remove triplets with very low classification likelihood as they do not form an object or contain too little shape information.

We define the region growing process by traversing the graph

Algorithm 1 Graph Traversal

```
while ||P|| > 0 do
  RandomSelect{(p_i, p_j, p_k) ∈ P | C(p_i, p_j, p_k) ≥ 0.55}
  O_l ← {p_i, p_j, p_k}
  while !stop do
    N(e_ij) ← [e_ij | p_i ∈ O_l , p_j ∉ O_l]
    e*_ij ← max_{e_ij ∈ N(e)} C(O_l ∪ p_j)
    if C(O_l ∪ p_j) ≥ C(O_l) · 0.95 then
      O_l ← O_l ∪ p_j
    else
      stop
    end if
  end while
end while
```

$G(V, E)$ from each object seed O_l while searching for high classification likelihood values. In each traversal step, we loop over all edges $\{e_{ij} | p_i \in O_l \ , p_j \notin O_l\}$. For each edge e_{ij}, we test classification likelihood value for the object defined by the union of patches: $O_l \bigcup p_j$. We then select the edge yielding the highest classification likelihood value and grow the current object seed by accumulating this patch: $O_l = O_l \bigcup p_j$.

In the *search-classify* process, we test for classification likelihood using very partial information. In the initial step, the object is defined by only three patches which we grow by accumulating more data. Therefore, classification likelihood value $C(O_l)$ in the beginning is low but then should increase if data is accumulated correctly. Hence, we facilitate our algorithm using an adaptive likelihood threshold. In the beginning we use a low threshold allowing the growth of many candidate seeds. As more data is accumulated, we increase likelihood threshold, assuming object seeds to grow and converge to the correct object (see Figure 7, middle). Typically, we start with a likelihood threshold of 0.55 and increase it every iteration to the average seeds' classification likelihoods.

Since object data is missing, we need to define classification queries for partial objects, i.e. classification queries with missing feature values. Detecting missing features is a complex problem since it requires apriori knowledge of the object's (complete) shape. Without some knowledge of the complete object's shape, it is practically impossible to decide what exactly is missing. Instead, we take a generic approach and compute for each object seed O_l a horizontal segmentation. Depending on the number of horizontal slabs, we fill the feature vectors with values, while for missing slabs, we fill out-of-range values which are easy to handle by the decision forest.

The region growing process stops if accumulation of any neighboring patch to O_l results in a decrease of the current likelihood value or if classification value is below the current threshold. Specifically, we request that $C(O_l \bigcup p_j) \geq 0.95 \cdot C(O_l)$ using a 0.95 factor to enable hysteresis of our system.

Initially, we select $20 - 50$ random triplet patches depending on the scene complexity and filter triplets with a likelihood value above 0.55. We then grow objects seeds until growing process is stopped. Note that it is possible of two objects O_l and O_m to share common patches. We resolve this ambiguity problem using the deform-to-fit step to follow. Trivially, if two objects merge completely in the growing process, we remove one to avoid redundancy (see Algorithm 1).

Template Fitting via Deformation Our deform-to-fit step is inspired by the part correspondence algorithm of Xu et al. [2010]. They compute shape correspondence using a set of local scale-deformations between objects parts. Similarly, we use a deformable

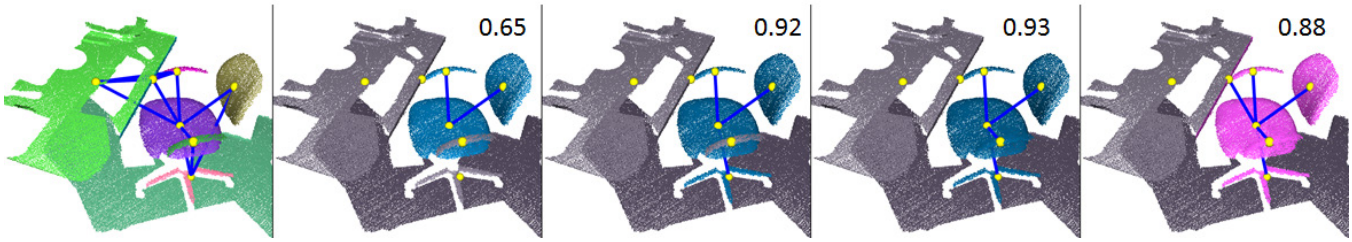

Figure 7: *Visualization of graph traversal and classification. Left-to-right, from a graph defined on initial patches, we select an initial object seed with above threshold classification confidence (mid-left). We traverse the graph in directions where classification confidence increases (number value, also blue color intensity). In rightmost figure, we show a neighboring patch (table-side) causing a steep decrease in classification confidence, hence we do not accumulate.*

template which allows non-rigid fitting onto the target point cloud via a set of localized scale deformations.

The input to our template fitting is a segmented point cloud object and a polygonal template with predefined scalable parts. Both template and point cloud belong to the same object class. Initially, we rigidly align the template to the scanned data by aligning their upward positions. Next, we perform part-based scaling deformations to fit the template parts to the point cloud, while minimizing a one-sided Euclidean distance from points to template. We alternate between establishing correspondence and deformation in a non-rigid ICP manner. In each step we compute closest distances between points to template and deform the template for only a fraction to minimize this distance.

Since this step can yield large deformations in the template parts, we perform a structure-preserving deformation optimization after each step similar to Xu et al. [2011]. Thus, after each deformation step, we optimize the local deformation scales using pre-analyzed part information for each template model. The optimization iteratively restores scale relations such as symmetries and proximities between parts of the template while refitting it to the point cloud.

Given a segmented and classified point cloud object, we fit several templates of the same class using the above deform-to-fit method and select the best matching template in terms of the one-sided distance. This step yields a refinement of the segmentation. We detect outliers in the *search-classify* process as patches with a large Euclidean distance to the fitted template (see Figure 8). We remove outlier patches from the object and return them to the scene classification process.

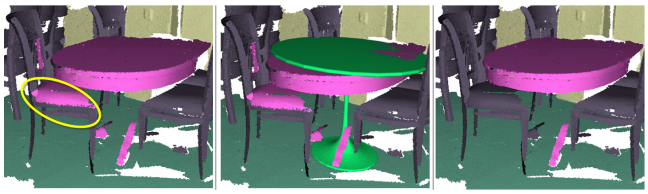

Figure 8: *Misclassification causes a chair seat (pink) to be attributed to a table (left). Fitting (middle), although not perfect, detects chair seat as outlier and is removed (right).*

5 Results and Discussion

In this work, we have experimented with a large amount of scanned indoor scenes and modeled objects for training and evaluation purposes. In our preprocessing step we have trained our classifier on both clean 3D digital models and manually segmented scans. The majority of objects that appear in our indoor scenes are chairs, tables, vases, LCD screens, keyboards and silverware. We used the ALGLIB library [Bochkanov 2012] to compute an RDF classifier. We demonstrate the good separation between chairs and tables achieved by our feature vector classification in Figure 6. Figures 1 and 2 demonstrate the effective power of our method in highly cluttered regions as in the table and chair legs. The algorithm correctly labels objects incrementally, resolving ambiguities and guiding the template fitting to a plausible reconstruction.

To scan large-scale scenes we used a commercial hand-held active-light scanner (MantisVision Inc.). This is a mid-range scanner acquiring dense 3D points with high precision (within 1mm positional accuracy). Compared to other commercial state-of-the-art scanners, ours provides highly detailed captures however, a large amount of data is still missing due to occlusions, restricted accessibility, and scene complexity (see Figures 1, 14). We ran all our experiments on a MacBookPro (4GB RAM, I7 2.6 Ghz CPU). Training for all our objects was below 5 minutes and query time with the RDF was a fraction of a second. Deform-to-fit process of fitting one object class takes 10 seconds on average.

In Figure 9, we analyze the performance of our classifier in terms of precision/recall, ROC and scalability. Both precision/recall and ROC diagrams show that our classifier is highly accurate with good performance values. For scalability, we have created synthetic scenes with varied object density (Figure 10(left)). We have virtually scanned the scene and applied our method resulting in classification of the scene into meaningful objects (Figure 10(right). Accuracy has stayed high with a moderate decline as clutter grows and small fluctuations due to scan noise.

We compare our method with the method of Lai et al. [2011] which focuses on detection of specific objects on top of tables (e.g, bowls, caps etc.). Figure 11 left and middle scenes show accurate segmentation results for both methods. In the right scene, our method was less accurate due to insufficient scan resolution. Nevertheless, since our method is not restricted to objects on tables, it has classified all objects in scene.

Our template database consists of 8 chairs, 8 tables, 3 monitors of different styles (also 1 vase and 1 cabinet). This is a sufficient number to templates as they can deform non-rigidly in a structure preserving manner allowing large variability. In Figure 12 we show the best fitting templates to classified objects in two scenes. Their fitting score measures the average one-sided Euclidean distance. Commonly, likelihood values are high due to robust classification, and reconstruction error is low due to deformable fitting. Nevertheless, when scanned data is too sparse even for our perception (top row - small seat), both classification and fitting values are marginal.

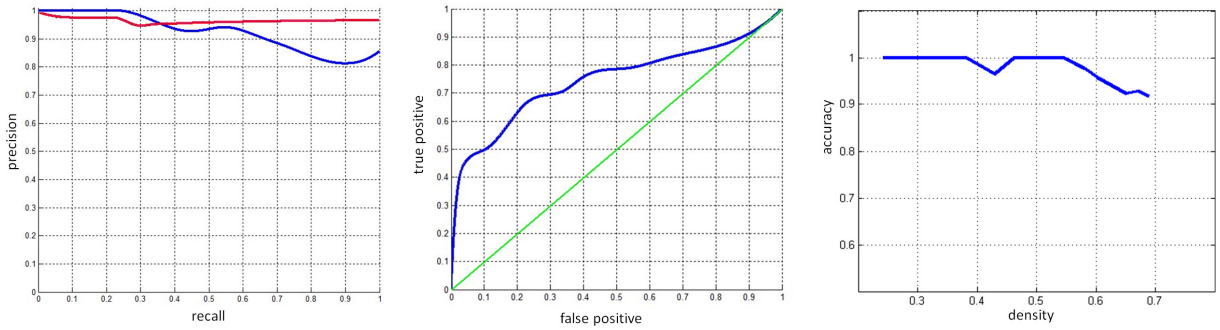

Figure 9: *Performance evaluation of our method. Left-to-right, precision/recall diagram for chairs (red) and tables (blue), ROC curve at various threshold settings and classifier scalability measured as accuracy vs. scene density.*

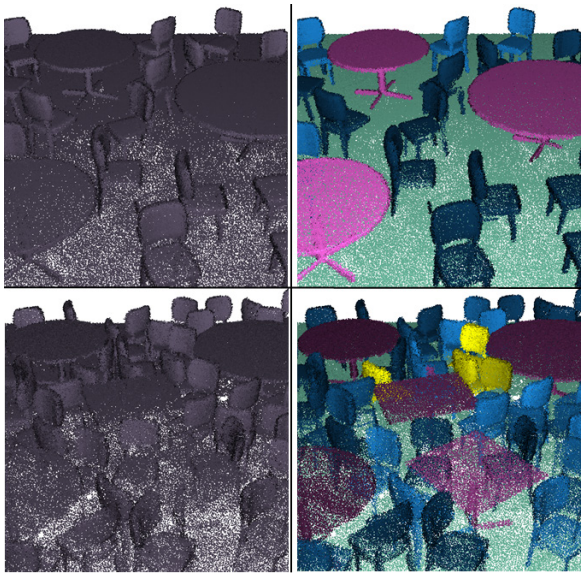

Figure 10: *Two scanned indoor scenes with low (top) and high (bottom) clutter levels. In our classification result (right column), color hue denotes object class (purple-table, blue-chair).*

Figure 13 shows the limitation of our method to natural position of objects in the scene. Our classification features as well as template fitting use the upward direction with respect to the ground floor, in their computation. Once this assumption is not valid, both classification (middle chair is classified as table) and fitting (rightmost chairs are aligned with floor) fail.

Finally, we present multiple results of our method on various scenes in Figure 14. Left-to-right is the raw input scan, over segmentation into piecewise smooth patches (colored independently), our segmentation and classification result, where color hue represents the object class and color intensity represents likelihood value within the class, and finally our deform-to-fit template reconstruction. Figure 14(a), shows a scene consisting of additional objects such as LCD screen and cupboards. Although LCD screen data is very sparse, it is correctly classified and reconstructed. Figures 14(b-d) are especially challenging due to their clutter and large missing parts. For example, in Figure 14(d) chair seats are completely missing as chairs are pulled under the table and can not be accessed by scanner. Our classification algorithm combined with deformable fitting successfully segments and reconstructs the scene resolving existing ambiguities. Scanned data is always partial and noisy, nevertheless, it contains sufficient information when viewed in a larger context than the local geometry in a point neighborhood. Figure 14(e) shows a perfect reconstruction as templates deform and match the scanned data precisely. Figures 14(f-g) show challenging examples of scenes with very large missing parts some containing only a seat to represent a chair. In most cases, data was sufficient to yield a correct reconstruction, however, in Figure 14(g) the yellow seat represented by a simple plane was incorrectly classified as table due to geometrical ambiguity.

Conclusions and Future Work We have presented an automatic algorithm for cluttered indoor scene understanding and modeling. Our method is based on initial random selection and iterative region growing with increasing classification likelihood. By selecting sufficiently many candidates, our algorithm can segment and classify the whole scene consistently. Scene reconstruction performs by deforming template models to fit the classified points.

In future work, we plan to extend this model to incorporate contextual information between different object. This way, we may be able to solve even very challenging ambiguities such as in Figure 14(g) using such information as chairs typically surround a table.

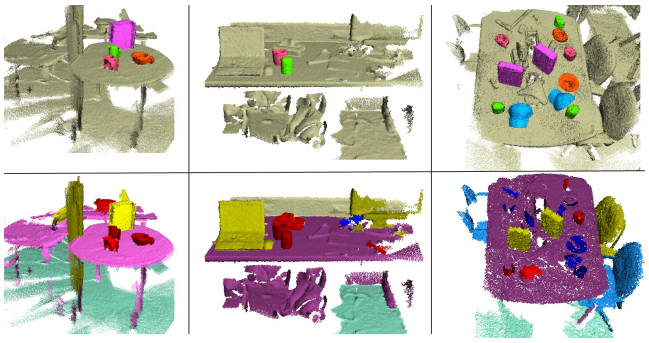

Figure 11: *Comparison of of Lai et al.[2011] (top) with our method (bottom) on three different scenes.*

Acknowledgements

We thank the reviewers for their valuable comments. This work was supported in part by NSFC (61272327, 61003190, 61232011), National 863 Program (2012AA011802, 2011AA010503), Guangdong Science and Technology Program (2011B050200007), Shenzhen Science and Technology Foundation (JC201005270340A, CXB201104220029A, JC201005270329A), Israel Science Foundation (ISF) and European IRG FP7.

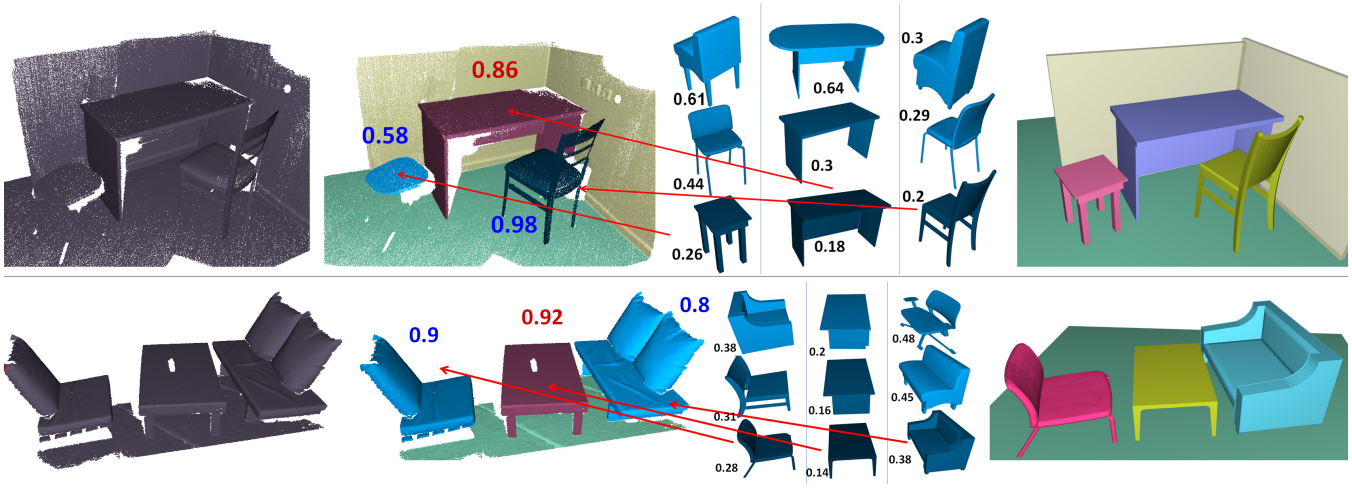

Figure 12: *Two examples of final classification likelihoods and template fitting values. Mid-left figure shows classified scene with likelihood values per object. Mid-right figure shows three best fitting templates to each object with average matching distance.*

Figure 13: *Limitation to upward orientation. Since classification and fitting assume a natural upward orientation, objects not obeying this assumption will yield incorrect results, here a chair classified and fitted by a table.*

References

ANGUELOV, D., TASKAR, B., CHATALBASHEV, V., KOLLER, D., GUPTA, D., HEITZ, G., AND NG, A. 2005. Discriminative learning of markov random fields for segmentation of 3d scan data. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2 - Volume 02*, CVPR '05, 169–176.

BELONGIE, S., MALIK, J., AND PUZICHA, J. 2002. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell. 24*, 509–522.

BELONGIE, S., MALIK, J., AND PUZICHA, J. 2002. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell. 24* (April), 509–522.

BOCHKANOV, S., 2012. Alglib library. http://www.alglib.net/.

BREIMAN, L. 2001. Random forests. *Mach. Learn. 45*, 5–32.

DICK, A. R., TORR, P. H. S., AND CIPOLLA, R. 2004. Modelling and interpretation of architecture from several images. *Int. J. Comput. Vision 60*, 2, 111–134.

FEI-FEI, L., FERGUS, R., AND PERONA, P. 2007. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Comput. Vis. Image Underst. 106*, 59–70.

FISHER, M., AND HANRAHAN, P. 2010. Context-based search for 3d models. In *ACM SIGGRAPH Asia 2010 papers*, 182:1–182:10.

FISHER, M., SAVVA, M., AND HANRAHAN, P. 2011. Characterizing structural relationships in scenes using graph kernels. *ACM Trans. Graph.*, 34:1–34:12.

FROME, A., HUBER, D., KOLLURI, R., AND BÜLOW, T. 2004. Recognizing objects in range data using regional point descriptors. In *ECCV*, 224–237.

FROME, A., HUBER, D., KOLLURI, R., BULOW, T., AND MALIK, J. 2004. Recognizing objects in range data using regional point descriptors. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

FU, H., COHEN-OR, D., DROR, G., AND SHEFFER, A. 2008. Upright orientation of man-made objects. In *ACM SIGGRAPH 2008*, 42:1–42:7.

FURUKAWA, Y., CURLESS, B., SEITZ, S. M., AND SZELISKI, R. 2009. Reconstructing building interiors from images.

GAL, R., SHAMIR, A., HASSNER, T., PAULY, M., AND COHEN-OR, D. 2007. Surface reconstruction using local shape priors. In *Proc. of Eurographics Symp. on Geometry Processing*, 253–262.

GALLEGUILLOS, C., RABINOVICH, A., AND BELONGIE, S. 2008. Object categorization using co-occurrence, location and appearance. *IEEE Conference on Computer Vision and Pattern Recognition (2008)*, 1–8.

GOESELE, M., SNAVELY, N., CURLESS, B., HOPPE, H., AND SEITZ, S. 2007. Multi-view stereo for community photo collections. In *Proc. of Int. Conf. on Comp. Vis.*, 1–8.

GOLOVINSKIY, A., KIM, V. G., AND FUNKHOUSER, T. 2009. Shape-based recognition of 3D point clouds in urban environ-

ments. *International Conference on Computer Vision (ICCV)* (Sept.).

HEDAU, V., HOIEM, D., AND FORSYTH, D. 2010. Thinking inside the box: using appearance models and context based on room geometry. In *Proc. Euro. Conf. on Comp. Vis.*, 224–237.

JOHNSON, A. E., AND HEBERT, M. 1999. Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Trans. Pattern Anal. Mach. Intell. 21* (May), 433–449.

KIM, Y. M., MITRA, N. J., YAN, D., AND GUIBAS, L. 2012. Acquiring 3d indoor environments with variability and repetition. In *ACM SIGGRAPH*, "to appear".

KOPPULA, H. S., ANAND, A., JOACHIMS, T., AND SAXENA, A. 2011. Semantic labeling of 3d point clouds for indoor scenes. In *NIPS*, 244–252.

LAI, K., AND FOX, D. 2010. Object recognition in 3d point clouds using web data and domain adaptation. *International Journal of Robotics Research 29*, 1019–1037.

LAI, K., BO, L., REN, X., AND FOX, D. 2011. A large-scale hierarchical multi-view rgb-d object dataset. *2011 IEEE International Conference on Robotics and Automation*, 1817–1824.

LI, Y., WU, X., CHRYSATHOU, Y., SHARF, A., COHEN-OR, D., AND MITRA, N. J. 2011. Globfit: consistently fitting primitives by discovering global relations. In *ACM SIGGRAPH*, 52:1–52:12.

LIVNY, Y., YAN, F., OLSON, M., CHEN, B., ZHANG, H., AND EL-SANA, J. 2010. Automatic reconstruction of tree skeletal structures from point clouds. *ACM Trans. Graph. 29*, 151:1–151:8.

LOWE, D. G. 2004. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision 60*, 91–110.

MATEI, B., SHAN, Y., SAWHNEY, H. S., TAN, Y., KUMAR, R., HUBER, D., AND HEBERT, M. 2006. Rapid object indexing using locality sensitive hashing and joint 3d-signature space estimation. *IEEE Trans. Pattern Anal. Mach. Intell. 28*, 1111–1126.

MUNOZ, D., BAGNELL, J. A., VANDAPEL, N., AND HEBERT, M. 2009. Contextual classification with functional max-margin markov networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

NAN, L., SHARF, A., ZHANG, H., COHEN-OR, D., AND CHEN, B. 2010. Smartboxes for interactive urban reconstruction. *Proc. of ACM SIGGRAPH 29*, 4, 1–10.

POLLEFEYS, M., NISTÉR, D., FRAHM, J. M., AKBARZADEH, A., MORDOHAI, P., CLIPP, B., ENGELS, C., GALLUP, D., KIM, S. J., MERRELL, P., SALMI, C., SINHA, S., TALTON, B., WANG, L., YANG, Q., STEWÉNIUS, H., YANG, R., WELCH, G., AND TOWLES, H. 2008. Detailed real-time urban 3D reconstruction from video. *Int. J. Comput. Vision 78*, 2-3, 143–167.

QUIGLEY, M., BATRA, S., GOULD, S., KLINGBEIL, E., LE, Q., WELLMAN, A., AND NG, A. Y. 2009. High-accuracy 3d sensing for mobile manipulation: improving object detection and door opening. In *Proceedings of the 2009 IEEE international conference on Robotics and Automation*, 3604–3610.

SCHNABEL, R., WAHL, R., AND KLEIN, R. 2007. Efficient ransac for point-cloud shape detection. *Computer Graphics Forum 26*, 2, 214–226.

SCHNABEL, R., DEGENER, P., AND KLEIN, R. 2009. Completion and reconstruction with primitive shapes. *Computer Graphics Forum (Proc. of Eurographics) 28*, 2, 503–512.

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. In *ACM SIGGRAPH*, "to appear".

SHEN, C.-H., HUANG, S.-S., FU, H., AND HU, S.-M. 2011. Adaptive partitioning of urban facades. In *Proceedings of the 2011 SIGGRAPH Asia Conference*, 184:1–184:10.

SHOTTON, J., JOHNSON, M., AND CIPOLLA, R. 2008. Semantic texton forests for image categorization and segmentation. In *Int. Conf. Computer Vision and Pattern Recognition*.

SHOTTON, J., FITZGIBBON, A., COOK, M., SHARP, T., FINOCCHIO, M., MOORE, R., KIPMAN, A., AND BLAKE, A. 2011. Real-Time human pose recognition in parts from a single depth image. In *CVPR*.

SILBERMAN, N., AND FERGUS, R. 2011. Indoor scene segmentation using a structured light sensor. In *Proc. of Int. Conf. on Comp. Vis.*

SINHA, S. N., STEEDLY, D., SZELISKI, R., AGRAWALA, M., AND POLLEFEYS, M. 2008. Interactive 3D architectural modeling from unordered photo collections. *ACM Trans. on Graphics 27*, 5, 1–10.

ULLMAN, S. 1996. *High-Level Vision: Object Recognition and Visual Cognition*. The MIT Press.

VIOLA, P., AND JONES, M. J. 2004. Robust real-time face detection. *Int. J. Comput. Vision 57*, 137–154.

VOSSELMAN, G., GORTE, B. G. H., SITHOLE, G., AND RABBANI, T. 2004. Recognising structure in laser scanner point clouds. *Information Sciences*, 1–6.

WERNER, T., AND ZISSERMAN, A. 2002. New techniques for automated architecture reconstruction from photographs. In *Proceedings of the 7th European Conference on Computer Vision, Copenhagen, Denmark*, vol. 2, 541–555.

XIAO, J., FANG, T., TAN, P., ZHAO, P., OFEK, E., AND QUAN, L. 2008. Image-based façade modeling. *ACM Trans. on Graphics 27*, 5, 1–10.

XIONG, X., AND HUBER, D. 2010. Using context to create semantic 3d models of indoor environments. In *Proceedings of the British Machine Vision Conference*, 45.1–45.11.

XU, K., LI, H., ZHANG, H., COHEN-OR, D., XIONG, Y., AND CHENG, Z.-Q. 2010. Style-content separation by anisotropic part scales. In *ACM SIGGRAPH Asia 2010 papers*, 184:1–184:10.

XU, K., ZHENG, H., ZHANG, H., COHEN-OR, D., , LIU, L., AND XIONG, Y. 2011. Photo-inspired model-driven 3d object modeling. *ACM Transactions on Graphics, (Proc. of SIGGRAPH 2011) 30*, 4, to appear.

ZHENG, Q., SHARF, A., WAN, G., LI, Y., MITRA, N. J., COHEN-OR, D., AND CHEN, B. 2010. Non-local scan consolidation for 3d urban scenes. *Proc. of ACM SIGGRAPH 29*, 1–9.

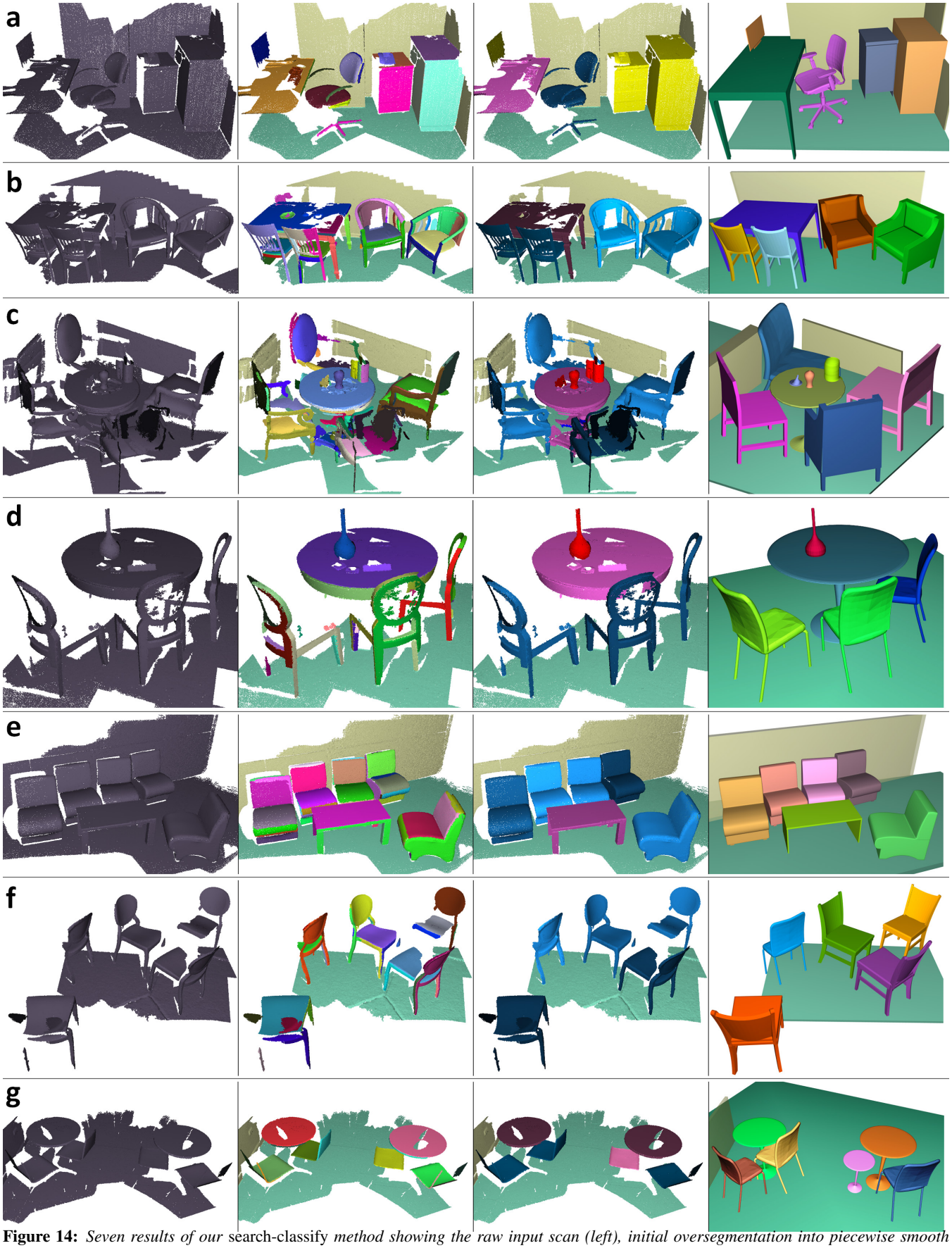

Figure 14: *Seven results of our* search-classify *method showing the raw input scan (left), initial oversegmentation into piecewise smooth patches (mid-left), classification result (mid-right) color hue representing the object class and color intensity representing final likelihood value, and template based reconstruction (right).*